Subsampled Ensemble Can Improve Generalization Tail Exponentially

Huajie Qian
DAMO Academy
Alibaba Group
Bellevue, WA 98004
h.qian@alibaba-inc.com

Donghao Ying
IEOR Department
UC Berkeley
Berkeley, CA 94720
donghaoy@berkeley.edu

Henry Lam
IEOR Department
Columbia University
New York, NY 10027
henry.lam@columbia.edu

Wotao Yin
DAMO Academy
Alibaba Group
Bellevue, WA 98004
wotao.yin@alibaba-inc.com

Abstract

Ensemble learning is a popular technique to improve the accuracy of machine learning models. It traditionally hinges on the rationale that aggregating multiple weak models can lead to better models with lower variance and hence higher stability, especially for discontinuous base learners. In this paper, we provide a new perspective on ensembling. By selecting the most frequently generated model from the base learner when repeatedly applied to subsamples, we can attain exponentially decaying tails for the excess risk, even if the base learner suffers from slow (i.e., polynomial) decay rates. This tail enhancement power of ensembling applies to base learners that have reasonable predictive power to begin with and is stronger than variance reduction in the sense of exhibiting rate improvement. We demonstrate how our ensemble methods can substantially improve out-of-sample performances in a range of numerical examples involving heavy-tailed data or intrinsically slow rates.

1 Introduction

Ensemble learning [17, 65] is a class of methods designed to improve the accuracy of machine learning models by combining multiple models, known as "base learners", through aggregation techniques such as averaging or majority voting. In the existing literature, ensemble methods—most notably bagging [8] and boosting [24]—are primarily justified based on their ability to reduce bias and variance or improve model stability. This justification has been shown to be particularly relevant for certain U-statistics [13] and models with hard-thresholding rules, such as decision trees [9, 19].

In contrast to this traditional understanding, we present a novel perspective on ensemble learning, demonstrating its capability to achieve a significantly stronger effect than variance reduction: By suitably selecting the best base learners trained on random subsamples, ensembling leads to exponentially decaying excess risk tails. Specifically, for general stochastic optimization problems that suffer from a slow (polynomial) decay in excess risk tails, ensembling can reduce these tails to an exponential decay rate, a substantial improvement beyond the constant-factor gains typically exhibited by variance reduction.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

To detail our contribution, we consider the generic stochastic optimization problem

$$\min_{\theta\in\Theta} L(\theta) := \mathbb{E}\left[l(\theta, z)\right], \tag{1}$$

where $\theta \in \Theta$ is the decision variable, $z \in \mathcal{Z}$ is a random variable governed by some probability distribution, and $l(\cdot,\cdot)$ is the cost function. A dataset of n i.i.d. samples $\{z_1, \ldots, z_n\}$ is drawn from the underlying distribution of z. In the context of machine learning, θ corresponds to the model parameters, $\{z_1, \ldots, z_n\}$ represents the training data, l denotes the loss function, and L is the population-level expected loss. More generally, (1) also encompasses data-driven decision-making problems, i.e., the integration of data on z into a downstream optimization task with overall cost function l and prescriptive decision θ. These problems are increasingly prevalent in various industrial applications [42, 6, 30]. For example, in supply chain network design, θ may represent the decision to open processing facilities, z the uncertain supply and demand, and l the total cost of processing and transportation.

Given the data, a learning algorithm can be used to map the data to an element in Θ, yielding a trained model or decision. This encompasses a variety of methods, including machine learning training algorithms and data-driven approaches such as sample average approximation (SAA) [61] and distributionally robust optimization (DRO) [55] in stochastic optimization. The theoretical framework and methodology proposed in this paper work for all learning algorithms that meet the formal performance criterion in our theorems.

Main Results at a High Level. Let $\hat{\theta}$ be the output of a learning algorithm. We characterize its generalization performance through the tail probability bound on the excess risk $L(\hat{\theta}) - \min_{\theta\in\Theta} L(\theta)$, i.e., $\mathbb{P}(L(\hat{\theta}) > \min_{\theta\in\Theta} L(\theta) + \delta)$ for some fixed $\delta > 0$, where the probability is over both the data and training randomness. A *polynomially decaying generalization tail* refers to:

$$\mathbb{P}\Big(L(\hat{\theta}) > \min_{\theta\in\Theta} L(\theta) + \delta\Big) \geq C_1 n^{-\alpha}, \tag{2}$$

for some $\alpha > 0$ and C_1 that depends on δ. Such bounds are common under heavy-tailed data distributions [43, 40, 41] due to slow concentration, which frequently arises in machine learning applications such as large language models [39, 63, 16], finance [50, 31], and physics [22, 53]. This can be illustrated with a simple linear program (LP):

Example 1.1 (LP with a polynomial tail)**.** *Consider the stochastic LP* $\min_{\theta\in[0,1]} \mathbb{E}[z\theta]$*, i.e.,* $l(\theta, z) = z\theta$ *and* $\Theta = [0, 1]$ *in (1), and its SAA solution* $\hat{\theta} \in \operatorname{argmin}_{\theta\in[0,1]} \sum_{i=1}^n z_i/n \cdot \theta$*. Assume* z *has a non-zero density everywhere and is symmetric with respect to its mean* $\mathbb{E}[z] = 1$ *(hence* $L(\theta) = \theta$*). Then we have* $\mathbb{P}(\hat{\theta} = 1) \geq \mathbb{P}(\sum_{i=1}^n z_i/n < 0) \geq \mathbb{P}(\sum_{i=1}^{n-1} z_i \leq n-1$ *and* $z_n < 1-n) = \mathbb{P}(\sum_{i=1}^{n-1} z_i \leq n-1)\mathbb{P}(z_n < 1-n)$*, where the last equality uses the independence of* z_i*'s. By the symmetry of* z*, we have* $\mathbb{P}(\sum_{i=1}^{n-1} z_i \leq n-1) = 1/2$*, thus for* $\delta \in (0, 1)$ *the tail*

$$\mathbb{P}\Big(L(\hat{\theta}) > \min_{\theta\in[0,1]} L(\theta) + \delta\Big) \geq \mathbb{P}(\hat{\theta} = 1) \geq \frac{1}{2}\mathbb{P}(z < 1-n). \tag{3}$$

If z *has a polynomial tail, e.g.,* $\mathbb{P}(z < 1-n) = \Omega(n^{-\alpha})$ *for some* $\alpha > 0$ *where* $\Omega(\cdot)$ *contains some multiplicative constant, the generalization tail bound (2) applies. Appendix E provides another example from linear regression.*

As the key contribution of our work, we propose ensemble methods that significantly improve these bounds, achieving an *exponentially decaying generalization tail*:

$$\mathbb{P}\Big(L(\hat{\theta}) > \min_{\theta\in\Theta} L(\theta) + \delta\Big) \leq C_2\gamma^{n/k}, \tag{4}$$

where k is the subsample size and $\gamma < 1$ depends on k, δ with $\gamma \to 0$ as $k \to \infty$. By appropriately choosing k at a slower rate in n, the decay becomes exponential. This exponential acceleration is fundamentally different from the well-known variance reduction benefit of ensembling in two perspectives. First, variance reduction refers to the smaller variability in predictions from models trained on independent data sets, thus has a more direct impact on the expected regret than the tail decay rate. Second, variance reduction typically yields a constant-factor improvement (e.g., [12] report a reduction factor of 3), whereas we obtain an order-of-magnitude improvement.

Main Intuition. Consider first the discrete space Θ. Our ensemble method employs a majority-vote mechanism at the model level: The learning algorithm is repeatedly run on subsamples to generate multiple models, and the model appearing most frequently is selected as the output. This resembles the majority vote in ensemble methods for classification but the voting is on models instead of classes. This process effectively estimates the mode of the sampling distribution of the learned model by subsampling, and thus is less susceptible to extreme data and training randomness that incurs the slow tail decay in (2). This mode estimation can be formalized via a surrogate optimization problem over the same decision space Θ as (1) that maximizes the probability of being output by the learning algorithm. The probability objective, as the expected value of a random indicator function, is uniformly bounded even if the original objective is heavy-tailed and hence admits exponential decay in the tail. Consequently, base learners with high-quality mode models receive an exponential enhancement in their tail behavior. To illustrate the main idea using Example 1.1, although $\mathbb{P}(\hat{\theta} = 1)$ can be substantial in a heavy-tailed setting, it holds that $\mathbb{P}(\hat{\theta} = 0) > \mathbb{P}(\hat{\theta} = 1)$, and thus the mode of $\hat{\theta}$, i.e., 0, recovers the optimal solution.

For general problems with possibly continuous decision spaces, we replace the majority vote with a voting mechanism based on the likelihood of being ϵ-optimal among all models when evaluated on random subsamples. This avoids the degeneracy of using a majority vote for continuous problems while retaining similar (in fact, even stronger) theoretical guarantees. For both discrete and continuous problems, our method fundamentally improves the tail behavior from (2) to (4).

Organization. The rest of the paper is organized as follows. Section 2 presents our methods and their finite-sample bounds. Section 3 presents experimental results, Section 4 discusses related work, and Section 5 concludes the paper. A review of additional related work, technical proofs, and additional experiments can be found in the appendix.

2 Methodology and Theoretical Guarantees

We consider the generic learning algorithm in the form of

$$\mathcal{A}(z_1, \ldots, z_n; \omega) : \mathcal{Z}^n \times \mathbf{\Omega} \to \Theta$$

that takes in the training data $(z_1, \ldots, z_n) \in \mathcal{Z}^n$ and outputs a model possibly under some algorithmic randomness $\omega \in \mathbf{\Omega}$ that is independent of the data. Examples of ω include gradient sampling in stochastic algorithms and feature/data subsampling in random forests. For convenience, we omit ω to write $\mathcal{A}(z_1, \ldots, z_n)$ when no confusion arises.

2.1 A Basic Procedure

We first introduce a procedure called MoVE that applies to discrete solution or model space Θ. MoVE, which is formally described in Algorithm 1, repeatedly draws a total of B subsamples from the data without replacement, learns a model via $\mathcal{A}$ on each subsample, and finally conducts a majority vote to output the most frequently subsampled model. Tie-breaking can be done arbitrarily.

Algorithm 1 **M**aj**o**rity **V**ote **E**nsembling (MoVE)

1: **Input:** A base learning algorithm $\mathcal{A}$, n i.i.d. observations $\mathbf{z}_{1:n} = (z_1, \ldots, z_n)$, subsample size $k < n$, and ensemble size B.
2: **for** $b = 1$ to B **do**
3: Randomly sample $\mathbf{z}_k^b = (z_1^b, \ldots, z_k^b)$ uniformly from $\mathbf{z}_{1:n}$ without replacement, and obtain $\hat{\theta}_k^b = \mathcal{A}(z_1^b, \ldots, z_k^b)$.
4: **end for**
5: **Output:** $\hat{\theta}_n \in \arg\max_{\theta \in \Theta} \sum_{b=1}^{B} \mathbb{1}(\theta = \hat{\theta}_k^b)$.

To understand MoVE from the lens of mode estimation, we consider an optimization associated with the base learner $\mathcal{A}$:

$$\max_{\theta \in \Theta} \; p_k(\theta) := \mathbb{P}\left(\theta = \mathcal{A}(z_1, \ldots, z_k)\right), \tag{5}$$

which maximizes the probability of a model being output by the base learner on k i.i.d. observations. Here the probability $\mathbb{P}$ is with respect to both the training data and the algorithmic randomness. If

$B = \infty$, MoVE essentially maximizes an empirical approximation of (5), i.e.

$$\max_{\theta \in \Theta} \mathbb{P}_* \left(\theta = \mathcal{A}(z_1^*, \ldots, z_k^*) \right), \tag{6}$$

where $(z_1^*, \ldots, z_k^*)$ is a uniform random subsample from $(z_1, \ldots, z_n)$, and $\mathbb{P}_*$ denotes the probability with respect to the algorithmic randomness and the subsampling randomness conditioned on $(z_1, \ldots, z_n)$. With a finite $B < \infty$, extra Monte Carlo noises are introduced, leading to the following maximization problem

$$\max_{\theta \in \Theta} \frac{1}{B} \sum_{b=1}^{B} \mathbb{1}(\theta = \mathcal{A}(z_1^b, \ldots, z_k^b)), \tag{7}$$

which gives exactly the output of MoVE. In other words, MoVE is a *bootstrap approximation* to the solution of (5). The following result materializes the intuition explained in Section 1 on the conversion of the original potentially heavy-tailed problem (1) into a probability maximization (7) that possesses exponential bounds.

Theorem 2.1 (Informal bound for Algorithm 1)**.** *Consider discrete decision space Θ. Let $\Theta^\delta := \{\theta \in \Theta : L(\theta) \leq \min_{\theta' \in \Theta} L(\theta') + \delta\}$ be the set of δ-optimal models and*

$$\eta_{k,\delta} := \max_{\theta \in \Theta} p_k(\theta) - \max_{\theta \in \Theta / \Theta^\delta} p_k(\theta),$$

where $p_k(\theta)$ is defined in (5) and $\max_{\theta \in \Theta \setminus \Theta^\delta} p_k(\theta)$ evaluates to 0 if $\Theta \setminus \Theta^\delta$ is empty. Then, for every $k \leq n$ and $\delta \geq 0$ such that $\eta_{k,\delta} > 0$, the solution output by MoVE *satisfies that*

$$\mathbb{P}(L(\hat{\theta}_n) > \min_{\theta \in \Theta} L(\theta) + \delta) \leq |\Theta| \left[4 \exp(-\Omega(n/k)) + \exp(-\Omega(B)) \right], \tag{8}$$

where $|\Theta|$ denotes the cardinality of Θ, and $\Omega(\cdot)$ contains multiplicative coefficients that depend on $\max_{\theta \in \Theta} p_k(\theta)$ and $\eta_{k,\delta}$. If $\eta_{k,\delta} > 4/5$, (8) is further bounded by

$$|\Theta| \left(3 \min \left\{ e^{-2/5}, C_1 \max \left\{ 1 - \max_{\theta \in \Theta} p_k(\theta), \max_{\theta \in \Theta / \Theta^\delta} p_k(\theta) \right\} \right\}^{\frac{n}{C_2 k}} + e^{-\frac{B}{C_3}} \right), \tag{9}$$

where $C_1, C_2, C_3 > 0$ are universal constants.

The formal statement is deferred to Theorem C.7 in Appendix C.2. Theorem 2.1 states that the excess risk tail of MoVE decays exponentially in the ratio n/k and ensemble size B. The bound (8) consists of two terms: The term $\exp(-\Omega(n/k))$ arises from the bootstrap approximation of (5) with (6), whereas the term $\exp(-\Omega(B))$ quantifies the Monte Carlo error in approximating (6) with a finite B. The multiplier $|\Theta|$ in the bound is avoidable, e.g., via a space reduction as in our next algorithm.

The quantity $\eta_{k,\delta}$ plays two roles. First, it quantifies how suboptimality in the surrogate problem (5) propagates to the original problem (1) in that every $\eta_{k,\delta}$-optimal solution for (5) is δ-optimal for (1). Second, $\eta_{k,\delta} > 0$ simply means that the mode solution is δ-optimal and hence $\eta_{k,\delta}$ directly quantifies the concentration of the base learner on near-optimal solutions. Therefore, a large $\eta_{k,\delta}$ signals the situation where the base learner already generalizes well. In this case, (8) reduces to (9). (9) suggests that our approach does not hurt the performance of an already high-performing base learner as its generalization power is inherited through the $\max\left\{1 - \max_{\theta \in \Theta} p_k(\theta), \max_{\theta \in \Theta / \Theta^\delta} p_k(\theta)\right\}$ term in the bound. See Appendix B for a more detailed discussion.

Theorem 2.1 also hints at the choice of hyperparameters k and B. As long as $\eta_{k,\delta} > 0$, our bound decays exponentially fast, and in this regime the bound (8) suggests that a smaller k (consequently a larger ratio n/k) leads to thinner tails. However, like other subsampling-based ensemble methods (e.g., subagging [12]), reducing the subsample size k also enlarges the model bias. In experiments, we set $k = \max(10, n/200)$ for a balance between tail and bias performance. Regarding the choice of B, we observe from (8) that using a $B = \mathcal{O}(n/k)$ is sufficient to control the Monte Carlo error to a similar magnitude as the statistical error.

Applying Theorem 2.1 to Example 1.1 gives an exponential tail as opposed to the slow decay in (3).

Corollary 2.2 (Enhanced tail for Example 1.1)**.** *Consider the stochastic LP in Example 1.1 and denote $q_k := \mathbb{P}(\sum_{i=1}^k z_i > 0)$. We have $q_k > 1/2$ by the symmetry of z. If* MoVE *is applied to Example 1.1 with $\mathcal{A}$ being the SAA, we have $\max_{\theta \in \Theta} p_k(\theta) = q_k$, $\max_{\theta \in \Theta / \Theta^\delta} p_k(\theta) = 1 - q_k$ whenever $\delta < 1$,*

and $|\Theta| = 2$. *Consequently,* $\eta_{k,\delta} = 2q_k - 1 > 0$ *for every* $k > 0$ *and* $\delta < 1$, *ensuring the tail upper bound (8) holds. If* $q_k > 0.9$, *we also have the tail upper bound* $6\min\left\{e^{-2/5}, C_1(1-q_k)\right\}^{\frac{n}{C_2 k}} + 2e^{-\frac{B}{C_3}}$ *from (9).*

The proof of Corollary 2.2 can be found in Appendix C.3.

2.2 A More General Procedure

We next present a more general procedure called ROVE that applies to continuous space where the simple majority vote in MoVE can lead to degeneracy, i.e., all learned models appear exactly once in the pool. Moreover, this general procedure relaxes the dependence on $|\Theta|$ in the bound (8).

Algorithm 2 Retrieval and ϵ-**O**ptimality **V**ote **E**nsembling (ROVE / ROVEs)

Input: A base learning algorithm $\mathcal{A}$, n i.i.d. observations $\mathbf{z}_{1:n} = (z_1, \ldots, z_n)$, subsample size $k_1, k_2 < n$ (if no split) or $n/2$ (if split), ensemble sizes B_1 and B_2.

Phase I: Model Candidate Retrieval
for $b = 1$ to B_1 **do**
Randomly sample $\mathbf{z}^b_{k_1} = (z^b_1, \ldots, z^b_{k_1})$ uniformly from $\mathbf{z}_{1:n}$ (if no split) or $\mathbf{z}_{1:\lfloor \frac{n}{2} \rfloor}$ (if split) without replacement, and obtain $\hat{\theta}^b_{k_1} = \mathcal{A}(z^b_1, \ldots, z^b_{k_1})$.
end for
Let $\mathcal{S} := \{\hat{\theta}^b_{k_1} : b = 1, \ldots, B_1\}$ be the set of all retrieved models.

Phase II: ϵ-Optimality Vote
Choose $\epsilon \geq 0$ using the data $\mathbf{z}_{1:n}$ (if no split) or $\mathbf{z}_{1:\lfloor \frac{n}{2} \rfloor}$ (if split).
for $b = 1$ to B_2 **do**
Randomly sample $\mathbf{z}^b_{k_2} = (z^b_1, \ldots, z^b_{k_2})$ uniformly from $\mathbf{z}_{1:n}$ (if no split) or $\mathbf{z}_{\lfloor \frac{n}{2} \rfloor + 1:n}$ (if split) without replacement, and calculate

$$\widehat{\Theta}^{\epsilon,b}_{k_2} := \left\{ \theta \in \mathcal{S} : \frac{1}{k_2} \sum_{i=1}^{k_2} l(\theta, z^b_i) \leq \min_{\theta' \in \mathcal{S}} \frac{1}{k_2} \sum_{i=1}^{k_2} l(\theta', z^b_i) + \epsilon \right\}.$$

end for
Output: $\hat{\theta}_n \in \arg\max_{\theta \in \mathcal{S}} \sum_{b=1}^{B_2} \mathbb{1}(\theta \in \widehat{\Theta}^{\epsilon,b}_{k_2})$.

ROVE, displayed in Algorithm 2, proceeds initially the same as MoVE in repeatedly subsampling data and training the model using $\mathcal{A}$. However, in the aggregation step, instead of using a simple majority vote, ROVE outputs, among all the trained models, the one that has the highest likelihood of being ϵ-optimal. This ϵ-optimality avoids the degeneracy of the majority vote. Moreover, since we have restricted our output to the collection of retrieved models, the corresponding likelihood maximization is readily doable by direct enumeration. In addition, it helps reduce competition for votes among best models, as each subsample can now vote for multiple candidates, ensuring a high vote count for each of the top models even when there are many of them. This makes ROVE more effective than MoVE in the case of multiple (near) optima as our experiments will show. We have the following theoretical guarantees for Algorithm 2.

Theorem 2.3 (Informal bound for Algorithm 2)**.** *Let* $\mathcal{E}_{k,\delta} := \mathbb{P}(L(\mathcal{A}(z_1, \ldots, z_k)) > \min_{\theta \in \Theta} L(\theta) + \delta)$ *be the excess risk tail of* $\mathcal{A}$. *Consider Algorithm 2 with data splitting, i.e.,* ROVEs. *Let* $T_k(\cdot) := \mathbb{P}(\sup_{\theta \in \Theta} |(1/k) \sum_{i=1}^{k} l(\theta, z_i) - L(\theta)| > \cdot)$ *be the tail function of the maximum deviation of the empirical objective estimate. Then, for every* $\delta > 0$, *under mild conditions on* ϵ *and* $T_{k_2}(\cdot)$, *it holds that*

$$\begin{aligned} \mathbb{P}\Big(L(\hat{\theta}_n) > \min_{\theta \in \Theta} L(\theta) + 2\delta\Big) \leq B_1 \Big[3\exp(-\Omega(n/k_2)) + \exp(-\Omega(B_2))\Big] \\ + \exp(-\Omega(n/k_1)) + \exp(-\Omega(B_1)), \end{aligned} \tag{10}$$

where $\Omega(\cdot)$ *contains multiplicative coefficients depending on* $T_{k_2}(\cdot)$, ϵ, δ *and* $\mathcal{E}_{k_1,\delta}$.

Consider Algorithm 2 without data splitting, i.e., ROVE, *and discrete space* Θ. *Assume* $\lim_{k \to \infty} T_k(\delta) = 0$ *for all* $\delta > 0$. *Then, for every fixed* $\delta > 0$, *under mild conditions, it holds that* $\lim_{n \to \infty} \mathbb{P}(L(\hat{\theta}_n) > \min_{\theta \in \Theta} L(\theta) + 2\delta) \to 0$.

The formal statement can be found in Appendix C.4. Theorem 2.3 provides an exponential excess risk tail, regardless of discrete or continuous space. The terms in the square bracket of (10) are inherited from the bound (9) for MoVE with the majority vote replaced by ϵ-optimality vote. In particular, the multiplier $|\Theta|$ in (9) is now replaced by B_1, the number of retrieved models from Phase I. The last two terms in (10) bound the performance sacrifice due to the restriction to the retrieved models.

ROVE may be carried out with the data split between the two phases, where it is referred to as ROVEs. Data splitting makes the procedure theoretically more tractable by avoiding inter-dependency between the phases but sacrifices some statistical power by halving the data size. Empirically we find it more effective not to split data.

The optimality threshold ϵ is allowed to be chosen in a data-driven way and the main goal guiding this choice is to distinguish models of different qualities. In other words, ϵ should be chosen to create enough variability in the likelihood of being ϵ-optimal across models. In our experiments, we find it a good strategy to choose an ϵ that leads to a maximum likelihood around $1/2$.

Technical Novelty. Our main theoretical results, Theorems 2.1 and 2.3, are derived using several novel techniques. First, we develop a sharper concentration result for U-statistics with binary kernels, improving upon standard Bernstein-type inequalities (e.g., [3, 57]). This refinement ensures the correct order of the bound, particularly (9), which captures the convergence of both the bootstrap approximation and the base learner, offering insights into the robustness of our methods for fast-converging base learners. Second, we perform a sensitivity analysis on the regret for the original problem (1) relative to the surrogate optimization (5), translating the superior generalization in the surrogate problem into accelerated convergence for the original. Finally, to establish asymptotic consistency for Algorithm 2 without data splitting, we develop a uniform law of large numbers (LLN) for the class of events of being ϵ-optimal, using direct analysis of the second moment of the maximum deviation. Uniform LLNs are particularly challenging here because, unlike fixed function classes in standard settings, this function class dynamically changes with the subsample size k_2 as $n \to \infty$.

3 Numerical Experiments

In this section, we numerically test Algorithm 1 (MoVE), Algorithm 2 with (ROVEs) and without (ROVE) data splitting in training machine learning models and solving stochastic programs. Due to space constraints, additional experimental results are provided in Appendix F. In particular, a comprehensive hyperparameter profiling of our algorithms is performed in Appendix F.3 to find empirically well-performing configurations for general use. Unless specified otherwise, all our experiments use the recommended configuration summarized at the end of Appendix F.3. All experiments are conducted on a personal computer, and Gurobi Optimizer is required for certain experiments on stochastic programs. The code is available at: https://github.com/mickeyhqian/VoteEnsemble.

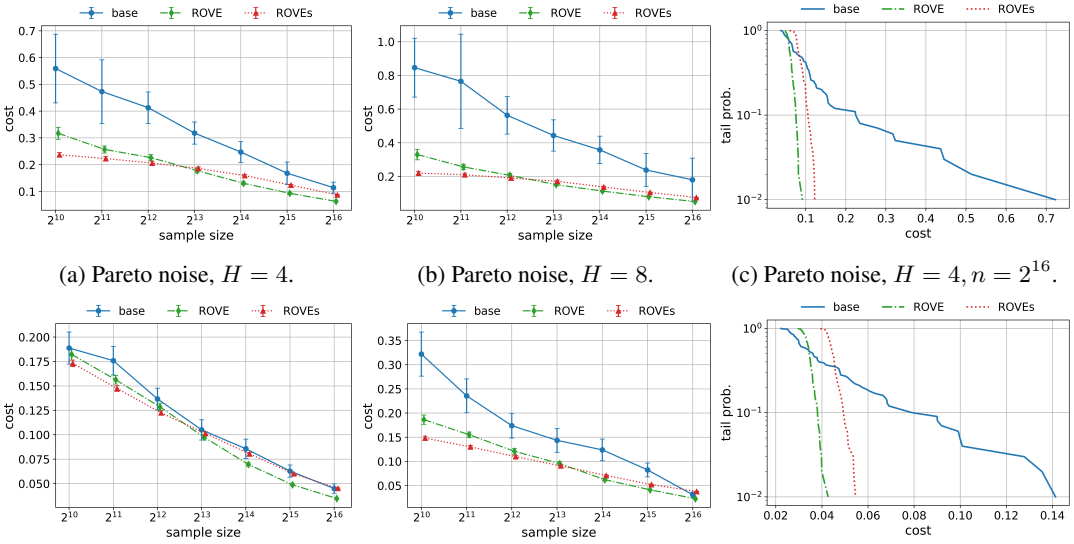
3.1 Neural Networks and Trees for Regression

Setup. We consider regression problems with multilayer perceptrons (MLPs) and decision trees. Note that classification models are prevalently trained using the cross-entropy loss that is inherently less prone to heavy-tailed noises thanks to the presence of the logarithm. For neural networks, the base learner splits the data into training (70%) and validation (30%), and uses Adam to minimize the mean squared error (MSE), with early stopping triggered when the validation improvement falls below 3% between epochs. The architecture of the MLPs is provided in Appendix F.1. For trees, the base learner is a single regression decision tree with the MSE as the splitting criterion. Besides the base learner, we also compare with three popular tree ensembles, Random Forests (RF) [9, 28], Gradient Boosted Decision Trees (GBDT) [26, 27], and XGBoost (XGB) [15]. RF are constructed with the same number of trees as our methods for a fair comparison, whereas GBDT and XGB are run with early stopping to avoid overfitting. MoVE is not included in this comparison as it's applicable to discrete problems only.

Synthetic Data. Input-output pairs (X, Y) are generated as $Y = (1/50) \cdot \sum_{j=1}^{50} \log(X_j + 1) + \varepsilon$, where each X_j is drawn independently from $\mathrm{Unif}(0, 2 + 198(j-1)/49)$, and the noise ε is independent of X with zero mean. We consider both standard Gaussian noise and Pareto noise $\varepsilon = \varepsilon_1 - \varepsilon_2$, where each $\varepsilon_i \sim \mathrm{Pareto}(2.1)$. The out-of-sample performance is estimated on a

common test set of one million samples. Each algorithm is repeatedly applied to 200 independently generated datasets to assess the average and tail performance.

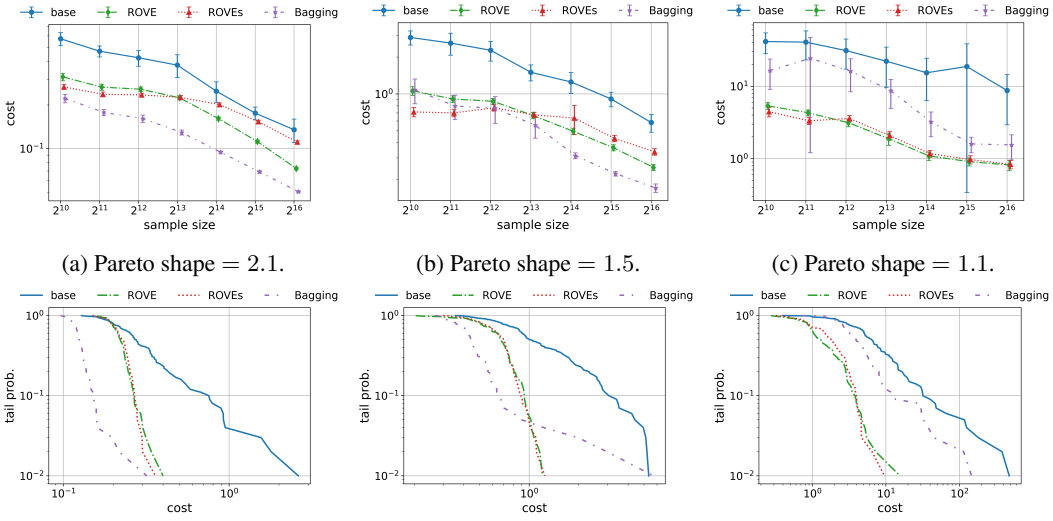
Real Data. We use three datasets from the UCI Machine Learning Repository [7]: *Bike Sharing* [21], *Superconductivity* [33], and *Gas Turbine Emission* [1]. The data is standardized (zero mean, unit variance). To evaluate the tail probabilities of out-of-sample costs, we permute each dataset 100 times, and each time use the first half for training and the second for testing. Results for three other datasets can be found in Figure 13 in Appendix F.



(a) Pareto noise, $H = 4$. (b) Pareto noise, $H = 8$. (c) Pareto noise, $H = 4, n = 2^{16}$.

(d) Gaussian noise, $H = 4$. (e) Gaussian noise, $H = 8$. (f) Gaussian noise, $H = 4, n = 2^{16}$.

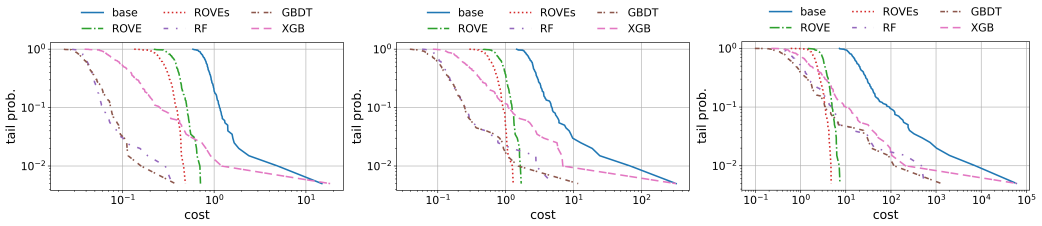
Figure 1: Results of neural networks. (a)(b)(d)(e): Expected out-of-sample costs (MSE) with 95% confidence intervals under different noise distributions and varying numbers of hidden layers (H). (c) and (f): Tail probabilities of out-of-sample costs.



(a) Pareto shape $= 2.1$. (b) Pareto shape $= 1.5$. (c) Pareto shape $= 1.1$.

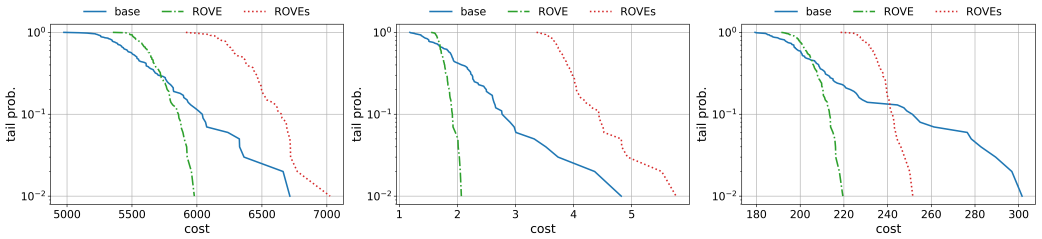
(d) Pareto shape $= 2.1, n = 2^{13}$. (e) Pareto shape $= 1.5, n = 2^{13}$. (f) Pareto shape $= 1.1, n = 2^{13}$.

Figure 2: Comparison with bagging in terms of expected out-of-sample costs (MSE) with 95% confidence intervals (a-c) or tail probabilities (d-f) under varying degrees of tail heaviness. Hyperparameters: $k_1 = \max(30, n/2), k_2 = \max(30, n/1000), B_1 = 50, B_2 = 1000$.



(a) Pareto shape = 2.5, $n = 2^{11}$. (b) Pareto shape = 2.0, $n = 2^{11}$. (c) Pareto shape = 1.5, $n = 2^{11}$.

Figure 3: Results of decision trees in terms of tail probabilities of out-of-sample costs (MSE). Hyperparameters: $k_1 = \max(30, n/10), k_2 = \max(30, n/200), B_1 = 50, B_2 = 200$.



(a) *Bike Sharing*. (b) *Gas Turbine Emission*. (c) *Superconductivity*.

Figure 4: Results of neural networks with 4 hidden layers on three real datasets, in terms of tail probabilities of out-of-sample costs (MSE).

Results. As shown in Figure 1, in heavy-tailed noise settings (Figures 1a–1c), both ROVE and ROVEs significantly outperform the base algorithm in terms of both expected out-of-sample MSE and tail performance under all sample sizes n. Notably, the performance improvement becomes more pronounced with deeper networks ($H = 8$), indicating that the benefits of ROVE and ROVEs are more apparent in models with higher expressiveness and lower bias.

In light-tailed settings (Figures 1d–1f), ROVE and ROVEs show comparable expected out-of-sample performance to the base when $H = 4$, but outperform the base as H increases. Additionally, ROVE and ROVEs outperform the base in tail probabilities even when $H = 4$. This indicates that ROVE and ROVEs provide better generalization as the model complexity grows even for light-tailed problems. Similar results for MLPs with 2 and 6 hidden layers can be found in Appendix F.4, where results on least squares regression and Ridge regression are also provided.

Figure 2 shows a comparison with bagging that resembles our method most closely among existing ensemble methods as both involve repeated training on randomly drawn subsamples. We implement bagging, or subagging [12] to be precise, on the MLP with $H = 4$ hidden layers by averaging the predictions of the repeatedly trained MLPs. The same subsample size and ensemble size are used for our methods and bagging to ensure a fair comparison. Whether bagging or our method wins depends on the tail heaviness: ROVE and ROVEs exhibit relatively inferior test performance when the noise has a shape of 2.1, but outperform bagging as the tail of the noise gets heavier towards a shape of 1.1.

Figure 3 demonstrates a similar pattern for tree base learners: ROVE and ROVEs outperform the base learner in all cases, and also outperform RF, GBDT, and XGB especially in high-end tails when the noise gets heavy-tailed with a Pareto shape of 1.5. For not so heavy-tailed cases, RF, GBDT, and XGB may perform better.

On real datasets (Figure 4), ROVE exhibits much lighter tails compared to the base on three datasets, and similar tail behavior on the other three. ROVEs, however, underperforms the base in these real-world scenarios, potentially due to the data split that compromises its statistical power.

3.2 Stochastic Programs

Setup. We consider four discrete stochastic programs: resource allocation, supply chain network design, maximum weight matching, and stochastic linear programming, and continuous mean-

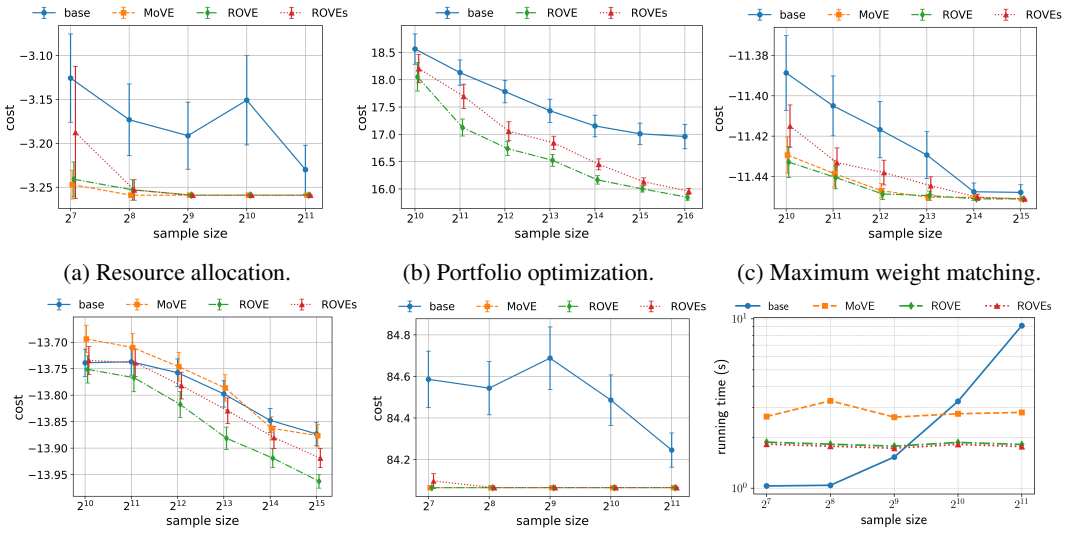
variance portfolio optimization. All problems are designed to possess heavy-tailed uncertainties. The base learner for all the problems is the SAA. Details of the problems are deferred to Appendix F.2 and results with DRO being the base learner are provided in Appendix F.4 Figure 19.

Results. Figure 5 shows that our ensemble methods significantly outperform the base algorithm in all cases except for the linear program case (Figure 5d). Notably, in the linear program case, ROVE and ROVEs still outperform the base, demonstrating their robustness, while MoVE performs slightly worse than the base under small sample sizes. Comparing ROVE and ROVEs, ROVE consistently exhibits superior performance than ROVEs in all cases.

When there is a unique optimal solution, MoVE and ROVE perform similarly, both generally better than ROVEs, as seen in Figures 5a-5c. However, in cases with multiple optima (Figure 5d), the performance of MoVE deteriorates while ROVE and ROVEs stay strong. This is in accordance with our discussion on the advantage of ϵ-optimality vote in Section 2.2. Additional results in Appendix F.4 shall further explain that optima multiplicity weakens the base learner for MoVE in the sense of decreasing the $\eta_{k,\delta}$ and hence inflating the tail bound in Theorem 2.1.

The running time comparison in Figure 5f shows that, despite requiring multiple runs on subsamples, our methods do not necessarily incur a higher computation cost compared to running base learner on the full sample, and can even be advantageous under large sample sizes. This is because, in problems like DRO [5, 55] and two-stage stochastic program, sample-based optimization often has a problem size that grows at least linearly with the sample size and induces a superlinearly growing computation cost. Subsampled optimizations, as our approach, are smaller and more manageable. In general, the computation efficiency of our method is ensured by the fact that no more than $\mathcal{O}(n/k)$ subsamples are needed as suggested by the theory and that training on subsamples can be easily parallelized.

Recommended Method. Among the three proposed ensemble methods, ROVE is the preferred choice over MoVE and ROVEs for general use as it's applicable to both discrete and continuous problems and consistently delivers superior performance across all scenarios.



(a) Resource allocation. (b) Portfolio optimization. (c) Maximum weight matching.

(d) Linear program (multiple optima). (e) Network design. (f) Network design running time.

Figure 5: Results for stochastic programs. (a)-(e): Expected out-of-sample costs with 95% confidence intervals. (f): Running time comparison in the network design problem.

4 Related Work

This work is closely connected to various topics in optimization and machine learning, and we only review the most relevant ones. See Appendix A for additional reviews.

Ensemble Learning. Ensemble learning [17, 65, 60] improves model performance by combining multiple weak learners into strong ones. Popular ensemble methods include bagging [8], boosting [24]

and stacking [62, 20]. Bagging enhances stability by training models on different bootstrap samples and combining their predictions through majority voting or averaging, effectively reducing variance, especially for unstable learners like decision trees that underpin random forests [9]. Subagging [12] is a variant of bagging that constructs the ensemble from subsamples in place of bootstrap samples. Boosting is a sequential process where each subsequent model corrects its predecessors' errors, reducing both bias and variance [37, 29]. Prominent boosting methods include AdaBoost [23], Stochastic Gradient Boosting (SGB) [26, 27], and Extreme Gradient Boosting (XGB) [25] which differ in their approaches to weighting training data and hypotheses. Instead of using simple aggregation like weighted averaging or majority voting, stacking trains a model to combine base predictions to further improve performance. A key procedural difference of our approach from these methods is that we perform majority voting at the model rather than prediction level to select a single best model from the ensemble. That is, our approach outputs models in the same space as the base learner, whereas existing ensemble methods yield aggregated models outside the base space. This also means a constant inference cost for our output model with respect to the ensemble size, as opposed to linearly growing costs seen in existing ensemble methods. Methodologically, our approach operates by accelerating excess risk tail convergence in lieu of bias/variance reduction, and hence is particularly effective in settings with heavy-tailed noise.

Optimization and Learning with Heavy Tails. Optimization with heavy-tailed noises has garnered significant attention due to its relevance in traditional fields such as portfolio management [50] and scheduling [38], as well as emerging domains like large language models [10, 2]. Tail bounds of most existing algorithms are guaranteed to decay exponentially under sub-Gaussian or uniformly bounded costs but deteriorate to a slow polynomial decay under heavy-tailedness [43, 40, 41, 56]. For SAA or ERM, faster rates are possible under the small-ball [52, 51, 59] or Bernstein's condition [18] on the function class, while our approach is free from such conditions. Considerable effort has been made to mitigate the adverse effects of heavy-tailedness with robust procedures among which the geometric median [54], or more generally, median-of-means (MOM) [47, 49] approach is most similar to ours. The basic idea there is to estimate a true mean by dividing the data into disjoint subsamples, computing an estimate on each, and then taking the median. [45, 48, 46, 44] use MOM in estimating the expected cost and establish exponential tail bounds for the mean squared loss and convex function classes. [36, 35] apply MOM directly on the solution level for continuous problems and require strong convexity from the cost to establish generalization bounds. Besides MOM, another approach estimates the expected cost via truncation [14] and allows heavy tails for linear regression [4, 64] or problems with uniformly bounded function classes [11], but is computationally intractable due to the truncation and thus more of theoretical interest. In contrast, our ensemble approach is a meta algorithm that provides exponential tails as long as the base learning algorithm possesses reasonble predictive performance as characterized in our Theorem 2.1. Relatedly, various techniques such as gradient clipping [16, 32] and MOM [58] have been adopted in stochastic gradient descent (SGD) algorithms to handle heavy-tailed noises, but their focus is the faster convergence of SGD rather than generalization.

5 Conclusion and Limitations

This paper introduces a novel ensemble technique that significantly improves generalization by estimating the mode of the sampling distribution of the base learner via subsampling. In particular, our approach converts polynomially decaying generalization tails into exponential decay, thus providing order-of-magnitude improvements as opposed to constant factor improvements exhibited by variance reduction. Extensive numerical experiments in both machine learning and stochastic programming validate its effectiveness, especially for scenarios with heavy-tailed data and slow convergence rates. This work underscores the powerful potential of our new ensemble approach across a broad range of machine learning applications.

Regarding limitation, our method may increase model bias like other subsampling-based techniques such as subagging [12], making it best suited for applications with relatively low bias, e.g., when the base learner is sufficiently expressive. Moreover, the tail guarantee of our method requires the mode of the sampling distribution of the base learner to be a reasonably good model.

References

[1] Gas Turbine CO and NOx Emission Data Set. UCI Machine Learning Repository, 2019. DOI: https://doi.org/10.24432/C5WC95.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[3] Miguel A Arcones. A bernstein-type inequality for u-statistics and u-processes. *Statistics & probability letters*, 22(3):239–247, 1995.

[4] Jean-Yves Audibert and Olivier Catoni. Robust linear least squares regression. *The Annals of Statistics*, 39(5):2766–2794, 2011.

[5] Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

[6] Dimitris Bertsimas, Shimrit Shtern, and Bradley Sturt. A data-driven approach to multistage stochastic linear optimization. *Management Science*, 69(1):51–74, 2023.

[7] Catherine L Blake. Uci repository of machine learning databases. *http://www. ics. uci. edu/˜ mlearn/MLRepository. html*, 1998.

[8] Leo Breiman. Bagging predictors. *Machine learning*, 24:123–140, 1996.

[9] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[10] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[11] Christian Brownlees, Edouard Joly, and Gábor Lugosi. Empirical risk minimization for heavy-tailed losses. *The Annals of Statistics*, 43(6):2507–2536, 2015.

[12] Peter Bühlmann and Bin Yu. Analyzing bagging. *The Annals of Statistics*, 30(4):927–961, 2002.

[13] Andreas Buja and Werner Stuetzle. Observations on bagging. *Statistica Sinica*, pages 323–351, 2006.

[14] Olivier Catoni. Challenging the empirical mean and empirical variance: A deviation study. *Annales de l'IHP Probabilités et statistiques*, 48(4):1148–1185, 2012.

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

[16] Ashok Cutkosky and Harsh Mehta. High-probability bounds for non-convex stochastic optimization with heavy tails. *Advances in Neural Information Processing Systems*, 34:4883–4895, 2021.

[17] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[18] Vu C Dinh, Lam S Ho, Binh Nguyen, and Duy Nguyen. Fast learning rates with heavy-tailed losses. *Advances in neural information processing systems*, 29, 2016.

[19] Harris Drucker and Corinna Cortes. Boosting decision trees. *Advances in neural information processing systems*, 8, 1995.

[20] Saso Džeroski and Bernard Ženko. Is combining classifiers with stacking better than selecting the best one? *Machine learning*, 54:255–273, 2004.

[21] Hadi Fanaee-T. Bike Sharing. UCI Machine Learning Repository, 2013. DOI: https://doi.org/10.24432/C5W894.

[22] Jean-Yves Fortin and Maxime Clusel. Applications of extreme value statistics in physics. *Journal of Physics A: Mathematical and Theoretical*, 48(18):183001, 2015.

[23] Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969, 2003.

[24] Yoav Freund, Robert E Schapire, et al. Experiments with a new boosting algorithm. In *icml*, volume 96, pages 148–156. Citeseer, 1996.

[25] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors). *The Annals of Statistics*, 28(2):337–407, 2000.

[26] Jerome H Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

[27] Jerome H Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

[28] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

[29] Indrayudh Ghosal and Giles Hooker. Boosting random forests to reduce bias; one-step boosted forest and its variance estimate. *Journal of Computational and Graphical Statistics*, 30(2):493–502, 2020.

[30] Shubhechyya Ghosal, Chin Pang Ho, and Wolfram Wiesemann. A unifying framework for the capacitated vehicle routing problem under risk and ambiguity. *Operations Research*, 72(2):425–443, 2024.

[31] Manfred Gilli and Evis Këllezi. An application of extreme value theory for measuring financial risk. *Computational Economics*, 27:207–228, 2006.

[32] Eduard Gorbunov, Marina Danilova, and Alexander Gasnikov. Stochastic optimization with heavy-tailed noise via accelerated gradient clipping. *Advances in Neural Information Processing Systems*, 33:15042–15053, 2020.

[33] Kam Hamidieh. Superconductivty Data. UCI Machine Learning Repository, 2018. DOI: https://doi.org/10.24432/C53P47.

[34] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[35] Daniel Hsu and Sivan Sabato. Heavy-tailed regression with a generalized median-of-means. In *International Conference on Machine Learning*, pages 37–45. PMLR, 2014.

[36] Daniel Hsu and Sivan Sabato. Loss minimization and parameter estimation with heavy tails. *Journal of Machine Learning Research*, 17(18):1–40, 2016.

[37] Bulat Ibragimov and Gleb Gusev. Minimal variance sampling in stochastic gradient boosting. *Advances in Neural Information Processing Systems*, 32, 2019.

[38] Sungjin Im, Benjamin Moseley, and Kirk Pruhs. Stochastic scheduling of heavy-tailed jobs. In *32nd International Symposium on Theoretical Aspects of Computer Science (STACS 2015)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2015.

[39] Hamid Jalalzai, Pierre Colombo, Chloé Clavel, Eric Gaussier, Giovanna Varni, Emmanuel Vignon, and Anne Sabourin. Heavy-tailed representations, text polarity classification & data augmentation. *Advances in Neural Information Processing Systems*, 33:4295–4307, 2020.

[40] Jie Jiang, Zhiping Chen, and Xinmin Yang. Rates of convergence of sample average approximation under heavy tailed distributions. *To preprint on Optimization Online*, 2020.

[41] Jie Jiang and Shengjie Li. On complexity of multistage stochastic programs under heavy tailed distributions. *Operations Research Letters*, 49(2):265–269, 2021.

[42] Sachin S Kamble, Angappa Gunasekaran, and Shradha A Gawankar. Achieving sustainable performance in a data-driven agriculture supply chain: A review for research and applications. *International Journal of Production Economics*, 219:179–194, 2020.

[43] Vlasta Kaňková and Michal Houda. Thin and heavy tails in stochastic programming. *Kybernetika*, 51(3):433–456, 2015.

[44] Joon Kwon, Guillaume Lecué, and Matthieu Lerasle. A mom-based ensemble method for robustness, subsampling and hyperparameter tuning. *Electronic Journal of Statistics*, 15(1):1202–1227, 2021.

[45] Guillaume Lecué and Matthieu Lerasle. Learning from mom's principles: Le cam's approach. *Stochastic Processes and their applications*, 129(11):4385–4410, 2019.

[46] Guillaume Lecué and Matthieu Lerasle. Robust machine learning by median-of-means: Theory and practice. *The Annals of Statistics*, 48(2):906–931, 2020.

[47] Gábor Lugosi and Shahar Mendelson. Mean estimation and regression under heavy-tailed distributions: A survey. *Foundations of Computational Mathematics*, 19(5):1145–1190, 2019.

[48] Gabor Lugosi and Shahar Mendelson. Risk minimization by median-of-means tournaments. *Journal of the European Mathematical Society*, 22(3):925–965, 2019.

[49] Gábor Lugosi and Shahar Mendelson. Sub-Gaussian estimators of the mean of a random vector. *The Annals of Statistics*, 47(2):783–794, 2019.

[50] Georg Mainik, Georgi Mitov, and Ludger Rüschendorf. Portfolio optimization for heavy-tailed assets: Extreme risk index vs. markowitz. *Journal of Empirical Finance*, 32:115–134, 2015.

[51] Shahar Mendelson. Learning without concentration. *Journal of the ACM (JACM)*, 62(3):1–25, 2015.

[52] Shahar Mendelson. Learning without concentration for general loss functions. *Probability Theory and Related Fields*, 171(1):459–502, 2018.

[53] Anna PM Michel and Alan D Chave. Analysis of laser-induced breakdown spectroscopy spectra: the case for extreme value statistics. *Spectrochimica Acta Part B: Atomic Spectroscopy*, 62(12):1370–1378, 2007.

[54] Stanislav Minsker. Geometric median and robust estimation in banach spaces. *Bernoulli*, 21(4):2308–2335, 2015.

[55] Peyman Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166, 2018.

[56] Roberto I Oliveira and Philip Thompson. Sample average approximation with heavier tails i: non-asymptotic bounds with weak assumptions and stochastic constraints. *Mathematical Programming*, 199(1):1–48, 2023.

[57] Thomas Peel, Sandrine Anthoine, and Liva Ralaivola. Empirical bernstein inequalities for u-statistics. *Advances in Neural Information Processing Systems*, 23, 2010.

[58] Nikita Puchkin, Eduard Gorbunov, Nickolay Kutuzov, and Alexander Gasnikov. Breaking the heavy-tailed noise barrier in stochastic optimization problems. In *International Conference on Artificial Intelligence and Statistics*, pages 856–864. PMLR, 2024.

[59] Abhishek Roy, Krishnakumar Balasubramanian, and Murat A Erdogdu. On empirical risk minimization with dependent and heavy-tailed data. *Advances in Neural Information Processing Systems*, 34:8913–8926, 2021.

[60] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 8(4):e1249, 2018.

[61] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

[62] David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[63] Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank Reddi, Sanjiv Kumar, and Suvrit Sra. Why are adaptive methods good for attention models? *Advances in Neural Information Processing Systems*, 33:15383–15393, 2020.

[64] Lijun Zhang and Zhi-Hua Zhou. ℓ_1-regression with heavy-tailed distributions. *Advances in Neural Information Processing Systems*, 31, 2018.

[65] Zhi-Hua Zhou. *Ensemble methods: foundations and algorithms*. CRC press, 2012.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The major claim made in our paper is that we proposed a new ensemble learning method that attains an exponentially decaying tail for excess risk. This claim is theoretically proved in Section 2. Moreover, we have conducted extensive numerical experiments in Section 3 to support our theoretical results.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discussed the limitations of our work in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We clearly stated the assumptions/conditions required for each theoretical results. Proofs of the results are documented in Appendix C.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All our figures are reproducible, and the codes for the experiments are disclosed.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provided the codes for reproducing our experiments. No data is needed for our paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The problem instances used in the experiments can be found in our disclosed code. The experiment methodologies are clearly stated in the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We repeated each our experiments for more than 50 times, reporting both the average performance and the standard deviation.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: As stated at the beginning of Section 3, our experiments are performed on a personal computer, and Gurobi Optimizer is required for reproducing some of our experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our paper is a theoretical work studying ensemble learning and stochastic optimization, and it does not have any ethical issue.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: Our paper is a theoretical work studying ensemble learning and stochastic optimization, and it does not have societal impact concerns.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper is a theoretical work studying ensemble learning and stochastic optimization, and it does not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The Gurobi academic license is used for our numerical experiments.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: Our paper is a theoretical work studying ensemble learning and stochastic optimization, and we do not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our paper is a theoretical work studying ensemble learning and stochastic optimization, and it does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper is a theoretical work studying ensemble learning and stochastic optimization, and it does not incur such risks.

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: LLMs are only used to polish the writings of some sentences in our paper.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Supplemental Materials

The appendices are organized as follows. In Appendix A, we review additional related work. Appendix B presents additional technical discussion for Theorem 2.1. Next, in Appendix C, we document the proofs of the main theoretical results in our paper. Specifically, we introduce some preliminary definitions and lemmas in Appendix C.1. Then, the formal statement and the proof of Theorem 2.1 can be found in Appendix C.2. The proof of Corollary 2.2 is in Appendix C.3. The formal statement and the proof of Theorem 2.3 can be found in Appendix C.4. To improve clarity, we defer the proofs for all technical lemmas to Appendix D. In Appendix E, we provide another motivating example that supplements Example 1.1. Finally, we provide additional numerical experiments in Appendix F.

A Additional Related Work

Bagging for Stochastic Optimization. Bagging has been adopted in stochastic optimization for various purposes. The most relevant line of works [70, 90, 95, 71] study mixed integer reformulations for stochastic optimization with bagging approximated objectives such as random forests and ensembles of neural networks with the ReLU activation. These works focus on computational tractability instead of generalization performance. [66] empirically evaluates several statistical techniques including bagging against the plain SAA and finds bagging advantageous for portfolio optimization problems. [72] investigates a batch mean approach for continuous optimization that creates subsamples by dividing the data set into non-overlapping batches instead of resampling and aggregates SAA solutions on the subsamples via averaging, which is empirically demonstrated to reduce solution errors for constrained and high-dimensional problems. Another related batch of works [87, 88, 78, 77, 81] concern the use of bagging for constructing confidence bounds for generalization errors of data-driven solutions, but they do not attempt to improve generalization. Related to bagging, bootstrap has been utilized to quantify algorithmic uncertainties for randomized algorithms such as randomized least-squares algorithms [89], randomized Newton methods [74], and stochastic gradient descent [82, 97], which is orthogonal to our focus on generalization performance.

Machine Learning for Optimization. Learning to optimize (L2O) studies the use of machine learning in accelerating existing or discovering novel optimization algorithms. Much effort has been in training models via supervised or reinforcement learning to make critical algorithmic decisions such as cut selection (e.g., [80, 94]), search strategies (e.g., [85, 84, 91]), scaling [69], and primal heuristics [93] in mixed-integer optimization, or even directly generate high-quality solutions (e.g., neural combinatorial optimization pioneered by [67]). See [75, 76, 68, 96] for comprehensive surveys on L2O. This line of research is orthogonal to our goal, and L2O techniques can work as part of or directly serve as the base learning algorithm within our framework.

B Implications of Theorem 2.1 for Strong Base Learners

We provide a brief discussion of Theorem 2.1 applied to fast convergent base learners. Based on Theorem 2.1, the way $\max_{\theta\in\Theta} p_k(\theta)$ and $\max_{\theta\in\Theta/\Theta^\delta} p_k(\theta)$ enter into (9) reflects how the generalization performance of the base learning algorithm is inherited by our framework. To explain, large $\max_{\theta\in\Theta} p_k(\theta)$ and small $\max_{\theta\in\Theta/\Theta^\delta} p_k(\theta)$ correspond to better generalization of the base learning algorithm. This can be exploited by the bound (9) with the presence of $\max\left\{1-\max_{\theta\in\Theta} p_k(\theta),\ \max_{\theta\in\Theta/\Theta^\delta} p_k(\theta)\right\}$, which is captured with our sharper concentration of U-statistics with binary kernels. In particular, for base learning algorithms with fast generalization convergence, say $1-\max_{\theta\in\Theta} p_k(\theta) = \mathcal{O}(e^{-k})$ and $\max_{\theta\in\Theta/\Theta^\delta} p_k(\theta) = \mathcal{O}(e^{-k})$ for simplicity, we have $C_1 \max\left\{1-\max_{\theta\in\Theta} p_k(\theta),\ \max_{\theta\in\Theta/\Theta^\delta} p_k(\theta)\right\} = \mathcal{O}(e^{-k})$ and hence the first term in (9) becomes $\mathcal{O}(e^{-n})$ which matches the error of the base learning algorithm applied directly to the full data set.

C Proofs for Main Theoretical Results

C.1 Preliminaries

An important tool in the development of our theories is the U-statistic that naturally arises in subsampling without replacement. We first present the definition of U-statistic below and its concentration properties in Lemma C.2. The proof of Lemma C.2 can be found in Appendix D.1.

Definition C.1. Given the i.i.d. data set $\{z_1, \ldots, z_n\} \subset \mathcal{Z}$ and a (not necessarily symmetric) kernel of order $k \leq n$ is a function $\kappa : \mathcal{Z}^k \to \mathbb{R}$ such that $\mathbb{E}\left[|\kappa(z_1, \ldots, z_k)|\right] < \infty$, the U-statistic associated with the kernel κ is

$$U(z_1, \ldots, z_n) = \frac{1}{n(n-1)\cdots(n-k+1)} \sum_{1 \leq i_1, i_2, \cdots, i_k \leq n \text{ s.t. } i_s \neq i_t \ \forall 1 \leq s < t \leq k} \kappa(z_{i_1}, \ldots, z_{i_k}).$$

Lemma C.2 (MGF dominance of U-statistics from [34])**.** *For any integer $0 < k \leq n$ and any kernel $\kappa(z_1, \ldots, z_k)$, let $U(z_1, \ldots, z_n)$ be the corresponding U-statistic defined in Definition C.1, and*

$$\bar{\kappa}(z_1, \ldots, z_n) = \frac{1}{\lfloor n/k \rfloor} \sum_{i=1}^{\lfloor n/k \rfloor} \kappa(z_{k(i-1)+1}, \ldots, z_{ki}) \tag{11}$$

be the average of the kernel across the first $\lfloor n/k \rfloor k$ data. Then, for every $t \in \mathbb{R}$, it holds that

$$\mathbb{E}\left[\exp(tU)\right] \leq \mathbb{E}\left[\exp(t\bar{\kappa})\right].$$

Next, we present our sharper concentration bound for U-statistics with binary kernels. The proof of Lemma C.3 can be found in Appendix D.2.

Lemma C.3 (Concentration bound for U-statistics with binary kernels)**.** *Let $\kappa(z_1, \ldots, z_k; \omega)$ be a $\{0,1\}$-valued kernel of order $k \leq n$ that possibly depends on additional randomness ω that is independent of the data $\{z_1, \ldots, z_n\}$, $\kappa^*(z_1, \ldots, z_k) := \mathbb{E}\left[\kappa(z_1, \ldots, z_k; \omega) | z_1, \ldots, z_k\right]$, and $U(z_1, \ldots, z_n)$ be the U-statistic associated with κ^*. Then, it holds that*

$$\mathbb{P}\left(U - \mathbb{E}\left[\kappa\right] \geq \epsilon\right) \leq \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(\mathbb{E}\left[\kappa\right] + \epsilon \| \mathbb{E}\left[\kappa\right]\right)\right),$$
$$\mathbb{P}\left(U - \mathbb{E}\left[\kappa\right] \leq -\epsilon\right) \leq \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(\mathbb{E}\left[\kappa\right] - \epsilon \| \mathbb{E}\left[\kappa\right]\right)\right),$$

where $D_{\mathrm{KL}}(p\|q) := p \ln \frac{p}{q} + (1-p) \ln \frac{1-p}{1-q}$ is the KL-divergence between two Bernoulli random variables with parameters p and q, respectively.

Below, Lemma C.4 gives lower bounds for KL divergences which help analyze the bounds in Lemma C.3. The proof of Lemma C.4 is deferred to Appendix D.3.

Lemma C.4. *Let $D_{\mathrm{KL}}(p\|q) := p \ln \frac{p}{q} + (1-p) \ln \frac{1-p}{1-q}$ be the KL-divergence between two Bernoulli random variables with parameters p and q, respectively. Then, it holds that*

$$D_{\mathrm{KL}}(p\|q) \geq p \ln \frac{p}{q} + q - p. \tag{12}$$

If $p \in [\gamma, 1-\gamma]$ for some $\gamma \in (0, \frac{1}{2}]$, it also holds that

$$D_{\mathrm{KL}}(p\|q) \geq -\ln\left(2(q(1-q))^{\gamma}\right). \tag{13}$$

To incorporate all the proposed algorithms in a unified theoretical framework, we consider a set-valued mapping

$$\mathbb{A}(z_1, \ldots, z_k; \omega) : \mathcal{Z}^k \times \mathbf{\Omega} \to 2^{\Theta}, \tag{14}$$

where $\omega \in \mathbf{\Omega}$ denotes algorithmic randomness that is independent of the data $\{z_1, \ldots, z_k\} \in \mathcal{Z}^k$. Each of our proposed algorithms attempts to solve the probability-maximization problem

$$\max_{\theta \in \Theta} \hat{p}_k(\theta) := \mathbb{P}_*\left(\theta \in \mathbb{A}(z_1^*, \ldots, z_k^*; \omega)\right), \tag{15}$$

for a certain choice of $\mathbb{A}$, where $\{z_1^*, \ldots, z_k^*\}$ is subsampled from the i.i.d. data $\{z_1, \ldots, z_n\}$ uniformly without replacement, and $\mathbb{P}_*$ denotes the probability with respect to the algorithmic

randomness ω and the subsampling randomness conditioned on the data. Note that this problem is an empirical approximation of the problem

$$\max_{\theta\in\Theta} p_k(\theta) := \mathbb{P}\left(\theta \in \mathbb{A}(z_1,\ldots,z_k;\omega)\right). \tag{16}$$

The problem actually solved with a finite number of subsamples is

$$\max_{\theta\in\Theta} \bar{p}_k(\theta) := \frac{1}{B}\sum_{b=1}^{B} \mathbb{1}(\theta \in \mathbb{A}(z_1^b,\ldots,z_k^b;\omega_b)). \tag{17}$$

Specifically, Algorithm 1 uses

$$\mathbb{A}(z_1^*,\ldots,z_k^*;\omega) = \{\mathcal{A}(z_1^*,\ldots,z_k^*;\omega)\}, \tag{18}$$

where $\mathcal{A}$ denotes the base learning algorithm, and Algorithm 2 uses

$$\mathbb{A}(z_1^*,\ldots,z_{k_2}^*;\omega) = \left\{\theta\in\mathcal{S} : \frac{1}{k_2}\sum_{i=1}^{k_2} l(\theta,z_i^*) \le \min_{\theta'\in\mathcal{S}} \frac{1}{k_2}\sum_{i=1}^{k_2} l(\theta',z_i^*) + \epsilon\right\}, \tag{19}$$

conditioned on the solution set $\mathcal{S}$ retrieved in Phase I. Note that no algorithmic randomness is involved in (19) once the set $\mathcal{S}$ is given. Now, we introduce the following definitions.

Definition C.5. For any $\delta\in[0,1]$, let

$$\mathcal{P}_k^\delta := \{\theta\in\Theta : p_k(\theta) \ge \max_{\theta'\in\Theta} p_k(\theta') - \delta\} \tag{20}$$

be the set of δ-optimal solutions of problem (16). Let

$$\theta_k^{\max} \in \arg\max_{\theta\in\Theta} p_k(\theta)$$

be a solution with maximum probability that is chosen in a unique manner if there are multiple such solutions. Let

$$\widehat{\mathcal{P}}_k^\delta := \{\theta\in\Theta : \hat{p}_k(\theta) \ge \hat{p}_k(\theta_k^{\max}) - \delta\} \tag{21}$$

be the set of δ-optimal solutions relative to $\theta_k^{\max}$ for problem (15).

Definition C.6. Let

$$\Theta^\delta := \left\{\theta\in\Theta : L(\theta) \le \min_{\theta'\in\Theta} L(\theta') + \delta\right\} \tag{22}$$

be the set of δ-optimal solutions of problem (1). In particular, Θ^0 represents the set of optimal solutions. Let

$$\widehat{\Theta}_k^\delta := \left\{\theta\in\Theta : \frac{1}{k}\sum_{i=1}^{k} l(\theta,z_i) \le \min_{\theta'\in\Theta} \frac{1}{k}\sum_{i=1}^{k} l(\theta',z_i) + \delta\right\} \tag{23}$$

be the set of δ-optimal solutions of the SAA with i.i.d. data $(z_1,\ldots,z_k)$.

C.2 Proof of Theorem 2.1

Theorem C.7 (Formal finite-sample bound for Algorithm 1)**.** *Consider discrete decision space Θ. Let $p_k^{\max} := \max_{\theta\in\Theta} p_k(\theta)$, where $p_k(\theta)$ is defined in (5), and*

$$\eta_{k,\delta} := p_k^{\max} - \max_{\theta\in\Theta/\Theta^\delta} p_k(\theta), \tag{24}$$

where $\max_{\theta\in\Theta\setminus\Theta^\delta} p_k(\theta)$ evaluates to 0 if $\Theta\setminus\Theta^\delta$ is empty. Then, for every $k\le n$ and $\delta\ge 0$ such that $\eta_{k,\delta}>0$, the solution output by MoVE *satisfies that*

$$\begin{aligned}
&\mathbb{P}\left(L(\hat{\theta}_n) > \min_{\theta\in\Theta} L(\theta) + \delta\right)\\
\le&|\Theta|\left[\exp\left(-\frac{n}{2k}\cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{3\eta_{k,\delta}}{4}\middle\|p_k^{\max}-\eta_{k,\delta}\right)\right) + 2\exp\left(-\frac{n}{2k}\cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\eta_{k,\delta}}{4}\middle\|p_k^{\max}\right)\right)\right.\\
&\quad+\exp\left(-\frac{B}{24}\cdot\frac{\eta_{k,\delta}^2}{\min\{p_k^{\max},1-p_k^{\max}\}+3\eta_{k,\delta}/4}\right)\\
&\quad\left.+\mathbb{1}\left(p_k^{\max}+\frac{\eta_{k,\delta}}{4}\le 1\right)\cdot\exp\left(-\frac{n}{2k}\cdot D_{\mathrm{KL}}\left(p_k^{\max}+\frac{\eta_{k,\delta}}{4}\middle\|p_k^{\max}\right) - \frac{B}{24}\cdot\frac{\eta_{k,\delta}^2}{1-p_k^{\max}+\eta_{k,\delta}/4}\right)\right].
\end{aligned} \tag{25}$$

In particular, if $\eta_{k,\delta} > 4/5$, (25) is further bounded by

$$|\Theta| \left(3 \min \left\{ e^{-2/5}, C_1 \max \left\{ 1 - p_k^{\max}, \max_{\theta \in \Theta / \Theta^\delta} p_k(\theta) \right\} \right\}^{\frac{n}{C_2 k}} + e^{-B/C_3} \right), \tag{26}$$

where $C_1, C_2, C_3 > 0$ are universal constants, $|\Theta|$ denotes the cardinality of Θ, and $D_{\mathrm{KL}}(p\|q) := p \ln \frac{p}{q} + (1-p) \ln \frac{1-p}{1-q}$ is the Kullback–Leibler divergence between two Bernoulli distributions with means p and q.

We consider Algorithm 3, a generalization of Algorithm 1 applied to the set-valued learning algorithm $\mathbb{A}$ in (14). This framework recovers Algorithm 1 as a special case under condition (18). Again, we omit the algorithmic randomness ω in $\mathbb{A}$ for convenience. For Algorithm 3, we derive the following finite-sample guarantee.

Algorithm 3 Majority Vote Ensembling for Set-Valued Learning Algorithms

1: Input: A set-valued learning algorithm $\mathbb{A}$, n i.i.d. observations $\mathbf{z}_{1:n} = (z_1, \ldots, z_n)$, positive integers $k < n$, and ensemble size B.
2: **for** $b = 1$ to B **do**
3: Randomly sample $\mathbf{z}_k^b = (z_1^b, \ldots, z_k^b)$ uniformly from $\mathbf{z}_{1:n}$ without replacement, and obtain $\Theta_k^b = \mathbb{A}(z_1^b, \ldots, z_k^b)$
4: **end for**
5: Output $\hat{\theta}_n \in \arg\max_{\theta \in \Theta} \sum_{b=1}^B \mathbb{1}(\theta \in \Theta_k^b)$.

Theorem C.8 (Finite-sample bound for Algorithm 3)**.** *Consider discrete decision space Θ. Let $p_k^{\max} := \max_{\theta \in \Theta} p_k(\theta)$, where $p_k(\theta)$ is defined in (16). For any $\delta > 0$, denote*

$$\bar{\eta}_{k,\delta} := p_k^{\max} - \max_{\theta \in \Theta \setminus \Theta^\delta} p_k(\theta), \tag{27}$$

where $\max_{\theta \in \Theta \setminus \Theta^\delta} p_k(\theta)$ evaluates to 0 if $\Theta \setminus \Theta^\delta$ is empty. Then, for every $k \le n$ and $\delta \ge 0$ such that $\bar{\eta}_{k,\delta} > 0$, the solution output by Algorithm 3 satisfies that

$$\begin{aligned}
&\mathbb{P}\left(L(\hat{\theta}_n) > \min_{\theta \in \Theta} L(\theta) + \delta \right) \\
&\le |\Theta| \Bigg[\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{3\eta}{4} \middle\| p_k^{\max} - \eta \right) \right) + 2\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\eta}{4} \middle\| p_k^{\max} \right) \right) \\
&\qquad + \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\{p_k^{\max}, 1 - p_k^{\max}\} + 3\eta/4} \right) \\
&\qquad + \mathbb{1}\left(p_k^{\max} + \frac{\eta}{4} \le 1 \right) \cdot \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} + \frac{\eta}{4} \middle\| p_k^{\max} \right) - \frac{B}{24} \cdot \frac{\eta^2}{1 - p_k^{\max} + \eta/4} \right) \Bigg]
\end{aligned} \tag{28}$$

for every $\eta \in (0, \bar{\eta}_{k,\delta}]$. In particular, if $\bar{\eta}_{k,\delta} > 4/5$, (28) is further bounded by

$$|\Theta| \left(3 \min \left\{ e^{-2/5}, C_1 \max \left\{ 1 - p_k^{\max}, \max_{\theta \in \Theta \setminus \Theta^\delta} p_k(\theta) \right\} \right\}^{\frac{n}{C_2 k}} + \exp\left(-\frac{B}{C_3} \right) \right), \tag{29}$$

where $C_1, C_2, C_3 > 0$ are universal constants, and $D_{\mathrm{KL}}(p\|q) := p \ln \frac{p}{q} + (1-p) \ln \frac{1-p}{1-q}$ is the Kullback–Leibler divergence between two Bernoulli distributions with means p and q.

Proof of Theorem C.8. We first prove excess risk tail bounds for the problem (16), split into two lemmas, Lemmas C.9 and C.10 below. The proofs for these two lemmas can be found in Appendix D.4 and Appendix D.5, respectively.

Lemma C.9. *Consider discrete decision space Θ. Recall from Definition C.5 that $p_k^{\max} = p_k(\theta_k^{\max})$ holds for $\theta_k^{\max}$. For every $0 \le \epsilon \le \delta \le p_k^{\max}$, it holds that*

$$\begin{aligned}
\mathbb{P}\left(\widehat{\mathcal{P}}_k^\epsilon \not\subseteq \mathcal{P}_k^\delta \right) \le |\Theta| \Bigg[&\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\delta + \epsilon}{2} \middle\| p_k^{\max} - \delta \right) \right) \\
&+ \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\delta - \epsilon}{2} \middle\| p_k^{\max} \right) \right) \Bigg].
\end{aligned}$$

Lemma C.10. *Consider discrete decision space Θ. For every $\epsilon \in [0,1]$ it holds for the solution output by Algorithm 3 that*

$$\mathbb{P}_*\left(\hat{\theta}_n \notin \widehat{\mathcal{P}}_k^{\epsilon}\right) \leq |\Theta| \cdot \exp\left(-\frac{B}{6} \cdot \frac{\epsilon^2}{\min\left\{\hat{p}_k(\theta_k^{\max}), 1-\hat{p}_k(\theta_k^{\max})\right\}+\epsilon}\right),$$

where $|\cdot|$ denotes the cardinality of a set and $\mathbb{P}_*$ denotes the probability with respect to both the resampling randomness conditioned on the observations and the algorithmic randomness.

We are now ready for the proof of Theorem C.8. We first note that, if $\bar{\eta}_{k,\delta} > 0$, it follows from Definition C.5 that

$$\mathcal{P}_k^{\eta} \subseteq \Theta^{\delta} \text{ for any } \eta \in (0,\ \bar{\eta}_{k,\delta}).$$

Therefore, for any $\eta \in (0,\ \bar{\eta}_{k,\delta})$, we can write that

$$\begin{aligned}\mathbb{P}\left(\hat{\theta}_n \notin \Theta^{\delta}\right) \leq \mathbb{P}\left(\hat{\theta}_n \notin \mathcal{P}_k^{\eta}\right) \leq &\quad \mathbb{P}\left(\left\{\hat{\theta}_n \notin \widehat{\mathcal{P}}_k^{\eta/2}\right\} \cup \left\{\widehat{\mathcal{P}}_k^{\eta/2} \not\subseteq \mathcal{P}_k^{\eta}\right\}\right) \\ \leq &\quad \mathbb{P}\left(\hat{\theta}_n \notin \widehat{\mathcal{P}}_k^{\eta/2}\right) + \mathbb{P}\left(\widehat{\mathcal{P}}_k^{\eta/2} \not\subseteq \mathcal{P}_k^{\eta}\right).\end{aligned} \tag{30}$$

We first evaluate the second probability on the right-hand side of (30). Lemma C.9 gives that

$$\begin{aligned}\mathbb{P}\left(\widehat{\mathcal{P}}_k^{\eta/2} \not\subseteq \mathcal{P}_k^{\eta}\right) \leq |\Theta| \Bigg[&\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{3\eta}{4} \middle\| p_k^{\max} - \eta\right)\right) \\ &+ \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\eta}{4} \middle\| p_k^{\max}\right)\right)\Bigg].\end{aligned} \tag{31}$$

Next, by applying Lemma C.10 with $\epsilon = \eta/2$, we can bound the first probability on the right-hand side of (30) as

$$\mathbb{P}\left(\hat{\theta}_n \notin \widehat{\mathcal{P}}_k^{\eta/2}\right) \leq |\Theta| \cdot \mathbb{E}\left[\exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\left\{\hat{p}_k(\theta_k^{\max}), 1-\hat{p}_k(\theta_k^{\max})\right\}+\eta/2}\right)\right]. \tag{32}$$

Conditioned on the value of $\hat{p}_k(\theta_k^{\max})$, we can further upper-bound the right-hand side of (32) as follows

$$\begin{aligned}&\mathbb{E}\left[\exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\left\{\hat{p}_k(\theta_k^{\max}), 1-\hat{p}_k(\theta_k^{\max})\right\}+\eta/2}\right)\right] \\ \leq &\quad \mathbb{P}\left(\hat{p}_k(\theta_k^{\max}) \leq p_k^{\max} - \frac{\eta}{4}\right) \cdot \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{p_k^{\max}+\eta/4}\right) + \\ &\quad \mathbb{P}\left(|\hat{p}_k(\theta_k^{\max}) - p_k^{\max}| < \frac{\eta}{4}\right) \cdot \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\left\{p_k^{\max}, 1-p_k^{\max}\right\}+3\eta/4}\right) + \\ &\quad \mathbb{P}\left(\hat{p}_k(\theta_k^{\max}) \geq p_k^{\max} + \frac{\eta}{4}\right) \cdot \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{1-p_k^{\max}+\eta/4}\right) \\ \leq &\quad \mathbb{P}\left(\hat{p}_k(\theta_k^{\max}) \leq p_k^{\max} - \frac{\eta}{4}\right) + \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\left\{p_k^{\max}, 1-p_k^{\max}\right\}+3\eta/4}\right) + \\ &\quad \mathbb{P}\left(\hat{p}_k(\theta_k^{\max}) \geq p_k^{\max} + \frac{\eta}{4}\right) \cdot \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{1-p_k^{\max}+\eta/4}\right) \\ \overset{(i)}{\leq} &\quad \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\eta}{4} \middle\| p_k^{\max}\right)\right) + \\ &\quad \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\left\{p_k^{\max}, 1-p_k^{\max}\right\}+3\eta/4}\right) + \\ &\quad \mathbb{1}\left(p_k^{\max} + \frac{\eta}{4} \leq 1\right) \cdot \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} + \frac{\eta}{4} \middle\| p_k^{\max}\right)\right) \cdot \exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{1-p_k^{\max}+\eta/4}\right),\end{aligned}$$

where inequality (i) results from applying Lemma C.3 with $\hat{p}_k(\theta_k^{\max})$, the U-statistic estimate for $p_k^{\max}$. Together, the above equations imply that

$$\begin{aligned}
& \mathbb{P}\left(\hat{\theta}_n \notin \Theta^\delta\right) \\
\leq\ & |\Theta| \left[\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{3\eta}{4} \middle\| p_k^{\max} - \eta\right)\right)\right. \\
& +2\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\eta}{4} \middle\| p_k^{\max}\right)\right) \\
& +\exp\left(-\frac{B}{24} \cdot \frac{\eta^2}{\min\left\{p_k^{\max}, 1-p_k^{\max}\right\} + 3\eta/4}\right) \\
& \left.+\mathbb{1}\left(p_k^{\max} + \frac{\eta}{4} \leq 1\right) \cdot \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} + \frac{\eta}{4} \middle\| p_k^{\max}\right) - \frac{B}{24} \cdot \frac{\eta^2}{1 - p_k^{\max} + \eta/4}\right)\right].
\end{aligned}$$

Since the above probability bound is left-continuous in η and η can be arbitrarily chosen from $(0,\ \bar{\eta}_{k,\delta})$, the validity of the case $\eta = \bar{\eta}_{k,\delta}$ follows from pushing η to the limit $\bar{\eta}_{k,\delta}$. This gives (28).

To simplify the bound in the case $\bar{\eta}_{k,\delta} > 4/5$. Consider the bound (28) with $\eta = \bar{\eta}_{k,\delta}$. Since $p_k^{\max} \geq \bar{\eta}_{k,\delta}$ by the definition of $\bar{\eta}_{k,\delta}$, it must hold that $p_k^{\max} + \bar{\eta}_{k,\delta}/4 > 4/5 + 1/5 = 1$, therefore the last term in the finite-sample bound (28) vanishes. To simplify the first two terms in the finite-sample bound, we note that

$$\begin{aligned}
p_k^{\max} - \frac{3\bar{\eta}_{k,\delta}}{4} &\leq 1 - \frac{3}{4} \cdot \frac{4}{5} = \frac{2}{5}, \\
p_k^{\max} - \frac{3\bar{\eta}_{k,\delta}}{4} &\geq \bar{\eta}_{k,\delta} - \frac{3\bar{\eta}_{k,\delta}}{4} \geq \frac{1}{5}, \\
p_k^{\max} - \frac{\bar{\eta}_{k,\delta}}{4} &\leq 1 - \frac{1}{4} \cdot \frac{4}{5} = \frac{4}{5}, \\
p_k^{\max} - \frac{\bar{\eta}_{k,\delta}}{4} &\geq \bar{\eta}_{k,\delta} - \frac{\bar{\eta}_{k,\delta}}{4} \geq \frac{3}{5},
\end{aligned}$$

and that $p_k^{\max} - \bar{\eta}_{k,\delta} \leq 1 - \bar{\eta}_{k,\delta} \leq 1/5$, therefore by the bound (13) from Lemma C.4, we can bound the first two terms as

$$\begin{aligned}
& \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{3\bar{\eta}_{k,\delta}}{4} \middle\| p_k^{\max} - \bar{\eta}_{k,\delta}\right)\right) \\
\leq\ & \exp\left(\frac{n}{2k} \ln\left(2((p_k^{\max} - \bar{\eta}_{k,\delta})(1 - p_k^{\max} + \bar{\eta}_{k,\delta}))^{1/5}\right)\right) \\
=\ & \left(2((p_k^{\max} - \bar{\eta}_{k,\delta})(1 - p_k^{\max} + \bar{\eta}_{k,\delta}))^{1/5}\right)^{n/(2k)} \\
\leq\ & \left(2(p_k^{\max} - \bar{\eta}_{k,\delta})^{1/5}\right)^{n/(2k)} \\
=\ & \left(2^5(p_k^{\max} - \bar{\eta}_{k,\delta})\right)^{n/(10k)},
\end{aligned}$$

and similarly

$$\begin{aligned}
\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{\bar{\eta}_{k,\delta}}{4} \middle\| p_k^{\max}\right)\right) &\leq \exp\left(\frac{n}{2k} \ln\left(2(p_k^{\max}(1 - p_k^{\max}))^{1/5}\right)\right) \\
&= \left(2(p_k^{\max}(1 - p_k^{\max}))^{1/5}\right)^{n/(2k)} \\
&\leq \left(2(1 - p_k^{\max})^{1/5}\right)^{n/(2k)} \\
&= \left(2^5(1 - p_k^{\max})\right)^{n/(10k)}.
\end{aligned}$$

On the other hand, by Lemma C.4 both $D_{\mathrm{KL}}\left(p_k^{\max} - 3\bar{\eta}_{k,\delta}/4 \| p_k^{\max} - \bar{\eta}_{k,\delta}\right)$ and $D_{\mathrm{KL}}\left(p_k^{\max} - \bar{\eta}_{k,\delta}/4 \| p_k^{\max}\right)$ are bounded below by $\bar{\eta}_{k,\delta}^2/8$, therefore

$$\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k^{\max} - \frac{3\bar{\eta}_{k,\delta}}{4} \middle\| p_k^{\max} - \bar{\eta}_{k,\delta}\right)\right) \leq \exp\left(-\frac{n}{2k} \cdot \frac{\bar{\eta}_{k,\delta}^2}{8}\right) \leq \exp\left(-\frac{n}{25k}\right),$$

and the same holds for $\exp\left(-n/(2k)\cdot D_{\mathrm{KL}}\left(p_k^{\max}-\bar{\eta}_{k,\delta}/4\|p_k^{\max}\right)\right)$. For the third term in the bound (28) we have

$$\frac{\bar{\eta}_{k,\delta}^2}{\min\left\{p_k^{\max},1-p_k^{\max}\right\}+3\bar{\eta}_{k,\delta}/4}\geq\frac{(4/5)^2}{\min\left\{1,1/5\right\}+3/4}\geq\frac{16}{25},$$

and hence

$$\exp\left(-\frac{B}{24}\cdot\frac{\bar{\eta}_{k,\delta}^2}{\min\left\{p_k^{\max},1-p_k^{\max}\right\}+3\bar{\eta}_{k,\delta}/4}\right)\leq\exp\left(-\frac{B}{75/2}\right).$$

The first desired bound then follows by setting C_1, C_2, C_3 to be the appropriate constants. This completes the proof of Theorem C.8. □

Proof of Theorem C.7. Algorithm 1 is a special case of Algorithm 3 with the learning algorithm (18) that outputs a singleton, therefore the results of Theorem C.8 automatically apply. In particular, $\eta_{k,\delta}=\bar{\eta}_{k,\delta}$ in the context of Theorem C.7, and (25) follows from setting η to be $\eta_{k,\delta}$ in (28). □

C.3 Proof of Corollary 2.2

Proof of Corollary 2.2. By the continuity and symmetry of z we have $q_k=\mathbb{P}(\sum_{i=1}^k z_i>k)+\mathbb{P}(\sum_{i=1}^k z_i\in(0,k))=1/2+\mathbb{P}(\sum_{i=1}^k z_i\in(0,k))$. Since z has a non-zero density everywhere, $\mathbb{P}(\sum_{i=1}^k z_i\in(0,k))>0$, thus $q_k>1/2$ for every $k>0$. We note that the SAA of the linear program outputs either 0 or 1, therefore the space $[0,1]$ can be effectively viewed as the binary set $\{0,1\}$ and Theorem 2.1 is applicable with $|\Theta|=2$. To apply Theorem 2.1, it can be easily seen that $\max_{\theta\in\Theta}p_k(\theta)=\mathbb{P}(\hat{\theta}=0)=q_k$ and that $\max_{\theta\in\Theta/\Theta^\delta}p_k(\theta)=\mathbb{P}(\hat{\theta}=1)=1-q_k$ for $\delta<1$. This gives $\eta_{k,\delta}=2q_k-1>0$. Therefore the bound (8) holds for every $k>0$ and $\delta<1$. If $q_k>0.9$, we have $\eta_{k,\delta}>4/5$, and hence the bound (10) holds. The particular form of the bound (10) is then obtained by plugging in the values for $\max_{\theta\in\Theta}p_k(\theta)$, $\max_{\theta\in\Theta/\Theta^\delta}p_k(\theta)$ and $|\Theta|$. □

C.4 Proof of Theorem 2.3

Theorem C.11 (Formal finite-sample bound for Algorithm 2)**.** *Let $\mathcal{E}_{k,\delta}:=\mathbb{P}(L(\mathcal{A}(z_1,\ldots,z_k))>\min_{\theta\in\Theta}L(\theta)+\delta)$ be the excess risk tail of $\mathcal{A}$. Consider Algorithm 2 with data splitting, i.e.,* ROVEs*. Let $T_k(\cdot):=\mathbb{P}(\sup_{\theta\in\Theta}|(1/k)\sum_{i=1}^k l(\theta,z_i)-L(\theta)|>\cdot)$ be the tail function of the maximum deviation of the empirical objective estimate. For every $\delta>0$, if ϵ is chosen such that $\mathbb{P}\left(\epsilon\in[\underline{\epsilon},\bar{\epsilon}]\right)=1$ for some $0<\underline{\epsilon}\leq\bar{\epsilon}<\delta$ and $T_{k_2}\left((\delta-\bar{\epsilon})/2\right)+T_{k_2}\left(\underline{\epsilon}/2\right)<1/5$, then*

$$\begin{aligned}\mathbb{P}\Big(L(\hat{\theta}_n)>\min_{\theta\in\Theta}L(\theta)+2\delta\Big)\leq&B_1\left[3\min\left\{e^{-2/5},C_1T_{k_2}\left(\frac{\min\left\{\underline{\epsilon},\delta-\bar{\epsilon}\right\}}{2}\right)\right\}^{\frac{n}{2C_2k_2}}+e^{-B_2/C_3}\right]\\&+\min\left\{e^{-(1-\mathcal{E}_{k_1,\delta})/C_4},C_5\mathcal{E}_{k_1,\delta}\right\}^{\frac{n}{2C_6k_1}}+e^{-B_1(1-\mathcal{E}_{k_1,\delta})/C_7},\end{aligned}\tag{33}$$

where C_1, C_2, C_3 are the same as those in Theorem C.7, and C_4, C_5, C_6, C_7 are universal constants.

Consider Algorithm 2 without data splitting, i.e., ROVE*, and discrete space Θ. Assume $\lim_{k\to\infty}T_k(\delta)=0$ for all $\delta>0$. Then, for every fixed $\delta>0$, we have $\lim_{n\to\infty}\mathbb{P}(L(\hat{\theta}_n)>\min_{\theta\in\Theta}L(\theta)+2\delta)\to 0$, if $\limsup_{k\to\infty}\mathcal{E}_{k,\delta}<1$, $\mathbb{P}\left(\epsilon>\delta/2\right)\to 0$, k_1 and $k_2\to\infty$, n/k_1 and $n/k_2\to\infty$, and $B_1, B_2\to\infty$ as $n\to\infty$.*

We first present two lemmas to be used in the main proof. The following Lemma C.12 characterizes the exponentially improving quality of the solution set retrieved in Phase I, where its proof can be found in Appendix D.6.

Lemma C.12 (Quality of retrieved solutions in Algorithm 2)**.** *For every k and $\delta\geq 0$, the set of retrieved solutions $\mathcal{S}$ from Phase I of Algorithm 2 with $k_1=k$ and without data splitting satisfies that*

$$\mathbb{P}\left(\mathcal{S}\cap\Theta^\delta=\emptyset\right)\leq\min\left\{e^{-(1-\mathcal{E}_{k,\delta})/C_4},C_5\mathcal{E}_{k,\delta}\right\}^{\frac{n}{C_6k}}+\exp\left(-\frac{B_1}{C_7}(1-\mathcal{E}_{k,\delta})\right),\tag{34}$$

where $C_4, C_5, C_6, C_7>0$ are universal constants. The same bound with n replaced by $n/2$ holds true for Algorithm 2 with data splitting.

Then, Lemma C.13 gives bounds for the excess risk sensitivity $\bar{\eta}_{k,\delta}$ in the case of the set-valued learning algorithm (19). The proof of Lemma C.13 can be found in Appendix D.7.

Lemma C.13 (Bounds of $\bar{\eta}_{k,\delta}$ for the set-valued learning algorithm (19))**.** *Consider discrete decision space Θ. If the set-valued learning algorithm*

$$\mathbb{A}(z_1, \ldots, z_k; \omega) := \left\{ \theta \in \Theta : \frac{1}{k} \sum_{i=1}^{k} l(\theta, z_i) \leq \min_{\theta' \in \Theta} \frac{1}{k} \sum_{i=1}^{k} l(\theta', z_i) + \epsilon \right\}$$

is used with $\epsilon \geq 0$, it holds that

$$p_k^{\max} = \max_{\theta \in \Theta} p_k(\theta) \geq 1 - T_k\left(\frac{\epsilon}{2}\right), \tag{35}$$

$$\max_{\theta \in \Theta \setminus \Theta^{\delta}} p_k(\theta) \leq T_k\left(\frac{\delta - \epsilon}{2}\right), \tag{36}$$

and hence

$$\bar{\eta}_{k,\delta} \geq 1 - T_k\left(\frac{\epsilon}{2}\right) - T_k\left(\frac{\delta - \epsilon}{2}\right), \tag{37}$$

where T_k is the tail probability defined in Theorem 2.3.

To prove Theorem C.11, we also introduce some notations. For every non-empty subset $\mathcal{W} \subseteq \Theta$, we use the following counterpart of Definition C.6. Let

$$\mathcal{W}^{\delta} := \left\{ \theta \in \mathcal{W} : L(\theta) \leq \min_{\theta' \in \mathcal{W}} L(\theta') + \delta \right\} \tag{38}$$

be the set of δ-optimal solutions in the restricted decision space $\mathcal{W}$, and

$$\widehat{\mathcal{W}}_k^{\delta} := \left\{ \theta \in \mathcal{W} : \frac{1}{k} \sum_{i=1}^{k} l(\theta, z_i) = \min_{\theta' \in \mathcal{W}} \frac{1}{k} \sum_{i=1}^{k} l(\theta', z_i) + \delta \right\} \tag{39}$$

be the set of δ-optimal solutions of the SAA with an i.i.d. data set of size k.

Proof of Theorem C.11 for ROVEs. Given the retrieved solution set $\mathcal{S}$ and the chosen ϵ, the rest of Phase II of Algorithm 2 exactly performs Algorithm 3 on the restricted problem $\min_{\theta \in \mathcal{S}} \mathbb{E}\left[l(\theta, z)\right]$ to obtain $\hat{\theta}_n$ with the data $\boldsymbol{z}_{\lfloor n/2 \rfloor + 1:n}$, the set-valued learning algorithm (19), the chosen ϵ value and $k = k_2, B = B_2$.

To show the upper bound for the unconditional convergence probability $\mathbb{P}\left(\hat{\theta}_n \notin \Theta^{2\delta}\right)$, note that

$$\left\{\mathcal{S} \cap \Theta^{\delta} \neq \emptyset\right\} \cap \left\{ L(\hat{\theta}_n) \leq \min_{\theta \in \mathcal{S}} L(\theta) + \delta \right\} \subseteq \left\{ \hat{\theta}_n \in \Theta^{2\delta} \right\},$$

and hence by union bound we can write

$$\mathbb{P}\left(\hat{\theta}_n \notin \Theta^{2\delta}\right) \leq \mathbb{P}\left(\mathcal{S} \cap \Theta^{\delta} = \emptyset\right) + \mathbb{P}\left(L(\hat{\theta}_n) > \min_{\theta \in \mathcal{S}} L(\theta) + \delta \right). \tag{40}$$

$\mathbb{P}\left(\mathcal{S} \cap \Theta^{\delta} = \emptyset\right)$ has a bound from Lemma C.12. We focus on the second probability.

For a fixed retrieved subset $\mathcal{S} \subseteq \Theta$, define the tail of the maximum deviation on $\mathcal{S}$

$$T_k^{\mathcal{S}}(\cdot) := \mathbb{P}\left(\sup_{\theta \in \mathcal{S}} \left| \frac{1}{k} \sum_{i=1}^{k} l(\theta, z_i) - L(\theta) \right| > \cdot \right).$$

It is straightforward that $T_k^{\mathcal{S}}(\cdot) \leq T_k(\cdot)$ where T_k is the tail of the maximum deviation over the whole space Θ. Since $\mathbb{P}\left(\epsilon \in [\underline{\epsilon}, \bar{\epsilon}]\right) = 1$, we have

$$1 - T_{k_2}^{\mathcal{S}}\left(\frac{\epsilon}{2}\right) - T_{k_2}^{\mathcal{S}}\left(\frac{\delta - \epsilon}{2}\right) \geq 1 - T_{k_2}^{\mathcal{S}}\left(\frac{\underline{\epsilon}}{2}\right) - T_{k_2}^{\mathcal{S}}\left(\frac{\delta - \bar{\epsilon}}{2}\right).$$

If $T_{k_2}((\delta-\overline{\epsilon})/2)+T_{k_2}(\underline{\epsilon}/2)<1/5$, we have $T^{\mathcal{S}}_{k_2}((\delta-\overline{\epsilon})/2)+T^{\mathcal{S}}_{k_2}(\underline{\epsilon}/2)<1/5$ and subsequently $1-T^{\mathcal{S}}_{k_2}((\delta-\epsilon)/2)-T^{\mathcal{S}}_{k_2}(\epsilon/2)>4/5$, and hence $\bar{\eta}_{k_2,\eta}\geq 1-T^{\mathcal{S}}_{k_2}((\delta-\epsilon)/2)-T^{\mathcal{S}}_{k_2}(\epsilon/2)>4/5$ by Lemma C.13 for Phase II of ROVEs conditioned on $\mathcal{S}$ and ϵ, therefore the bound (29) from Theorem C.8 applies. Using the inequalities (35) and (36) to upper bound the $\min\{1-p_k^{\max}, p_k^{\max}-\bar{\eta}_{k,\delta}\}$ term in (29) gives

$$\begin{aligned}
&\mathbb{P}\left(L(\hat{\theta}_n)>\min_{\theta\in\mathcal{S}}L(\theta)+\delta\middle|\mathcal{S},\epsilon\right)\\
\leq\ & |\mathcal{S}|\left(3\min\left\{e^{-2/5}, C_1\max\left\{T^{\mathcal{S}}_{k_2}\left(\frac{\underline{\epsilon}}{2}\right), T^{\mathcal{S}}_{k_2}\left(\frac{\delta-\overline{\epsilon}}{2}\right)\right\}\right\}^{\frac{n}{2C_2k_2}}+\exp\left(-\frac{B_2}{C_3}\right)\right)\\
=\ & |\mathcal{S}|\left(3\min\left\{e^{-2/5}, C_1T^{\mathcal{S}}_{k_2}\left(\frac{\min\{\underline{\epsilon},\delta-\overline{\epsilon}\}}{2}\right)\right\}^{\frac{n}{2C_2k_2}}+\exp\left(-\frac{B_2}{C_3}\right)\right)\\
\leq\ & |\mathcal{S}|\left(3\min\left\{e^{-2/5}, C_1T_{k_2}\left(\frac{\min\{\underline{\epsilon},\delta-\overline{\epsilon}\}}{2}\right)\right\}^{\frac{n}{2C_2k_2}}+\exp\left(-\frac{B_2}{C_3}\right)\right).
\end{aligned}$$

Further relaxing $|\mathcal{S}|$ to B_1 and taking full expectation on both sides give

$$\mathbb{P}\left(L(\hat{\theta}_n)>\min_{\theta\in\mathcal{S}}L(\theta)+\delta\right)\leq B_1\left(3\min\left\{e^{-2/5}, C_1T_{k_2}\left(\frac{\min\{\underline{\epsilon},\delta-\overline{\epsilon}\}}{2}\right)\right\}^{\frac{n}{2C_2k_2}}+\exp\left(-\frac{B_2}{C_3}\right)\right).$$

This leads to the desired bound (33) after the above bound is plugged into (40) and the bound (34) from Lemma C.12 is applied with $k=k_1$. □

Proof of Theorem C.11 for ROVE. For every non-empty subset $\mathcal{W}\subseteq\Theta$ and k_2, we consider the indicator

$$\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}(z_1,\ldots,z_{k_2}):=\mathbb{1}\left(\frac{1}{k_2}\sum_{i=1}^{k_2}l(\theta,z_i)\leq\min_{\theta'\in\mathcal{W}}\frac{1}{k_2}\sum_{i=1}^{k_2}l(\theta',z_i)+\epsilon\right)\quad\text{for }\theta\in\mathcal{W},\epsilon\in[0,\delta/2],$$

which indicates whether a solution $\theta\in\mathcal{W}$ is ϵ-optimal for the SAA formed by $\{z_1,\ldots,z_{k_2}\}$. Here we add ϵ and $\mathcal{W}$ to the superscript to emphasize its dependence on them. The counterparts of the solution probabilities $p_k,\hat{p}_k,\bar{p}_k$ for $\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}$ are

$$\begin{aligned}
p^{\mathcal{W},\epsilon}_{k_2}(\theta)&:=\mathbb{E}\left[\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}(z_1,\ldots,z_{k_2})\right],\\
\hat{p}^{\mathcal{W},\epsilon}_{k_2}(\theta)&:=\mathbb{E}_*\left[\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}(z_1^*,\ldots,z_{k_2}^*)\right],\\
\bar{p}^{\mathcal{W},\epsilon}_{k_2}(\theta)&:=\frac{1}{B_2}\sum_{b=1}^{B_2}\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}(z_1^b,\ldots,z_{k_2}^b).
\end{aligned}$$

We need to show the uniform convergence of these probabilities for $\epsilon\in[0,\delta/2]$. To do so, we define a slighted modified version of $\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}$

$$\mathbb{1}^{\theta,\mathcal{W},\epsilon-}_{k_2}(z_1,\ldots,z_{k_2}):=\mathbb{1}\left(\frac{1}{k_2}\sum_{i=1}^{k_2}l(\theta,z_i)<\min_{\theta'\in\mathcal{W}}\frac{1}{k_2}\sum_{i=1}^{k_2}l(\theta',z_i)+\epsilon\right)\quad\text{for }\theta\in\mathcal{W},\epsilon\in[0,\delta/2],$$

which indicates a strict ϵ-optimal solution, and let $p^{\mathcal{W},\epsilon-}_{k_2},\hat{p}^{\mathcal{W},\epsilon-}_{k_2},\bar{p}^{\mathcal{W},\epsilon-}_{k_2}$ be the corresponding counterparts of solution probabilities. For any integer $m>1$ we construct brackets of size at most $1/m$ to cover the family of indicator functions $\{\mathbb{1}^{\theta,\mathcal{W},\epsilon}_{k_2}:\epsilon\in[0,\delta/2]\}$, i.e., let $m'=\lfloor p^{\mathcal{W},\delta/2}_{k_2}(\theta)m\rfloor$ and

$$\begin{aligned}
\epsilon_0&:=0,\\
\epsilon_i&:=\inf\left\{\epsilon\in[0,\delta/2]:p^{\mathcal{W},\epsilon}_{k_2}(\theta)\geq i/m\right\}\quad\text{for }1\leq i\leq m',\\
\epsilon_{m'+1}&:=\frac{\delta}{2},
\end{aligned}$$

where we assume that $\epsilon_i, i = 0, \ldots, m'+1$ are strictly increasing without loss of generality (otherwise we can delete duplicated values). Then for any $\epsilon \in [\epsilon_i, \epsilon_{i+1})$, we have that

$$\begin{aligned}
\bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - p_{k_2}^{\mathcal{W},\epsilon}(\theta) &\leq \bar{p}_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta) \\
&\leq \bar{p}_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) + p_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta) \\
&\leq \bar{p}_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) + \frac{1}{m}
\end{aligned}$$

and that

$$\begin{aligned}
\bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - p_{k_2}^{\mathcal{W},\epsilon}(\theta) &\geq \bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) \\
&\geq \bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta) + p_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_{i+1}-}(\theta) \\
&\geq \bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - \frac{1}{m}.
\end{aligned}$$

Therefore

$$\sup_{\epsilon\in[0,\delta/2]} \left|\bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - p_{k_2}^{\mathcal{W},\epsilon}(\theta)\right| \leq \max_{0\leq i\leq m'+1} \max\left\{\left|\bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right|, \left|\bar{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i-}(\theta)\right|\right\} + \frac{1}{m}. \tag{41}$$

To show that the random variable in (41) converges to 0 in probability, we note that the U-statistic has the minimum variance among all unbiased estimators, in particular the following simple sample average estimators based on the first $\lfloor n/k_2 \rfloor \cdot k_2$ data

$$\begin{aligned}
\tilde{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) &:= \frac{1}{\lfloor n/k_2 \rfloor} \sum_{i=1}^{\lfloor n/k_2 \rfloor} \mathbb{1}_{k_2}^{\theta,\mathcal{W},\epsilon}(z_{k_2(i-1)+1}, \ldots, z_{k_2 i}), \\
\tilde{p}_{k_2}^{\mathcal{W},\epsilon-}(\theta) &:= \frac{1}{\lfloor n/k_2 \rfloor} \sum_{i=1}^{\lfloor n/k_2 \rfloor} \mathbb{1}_{k_2}^{\theta,\mathcal{W},\epsilon-}(z_{k_2(i-1)+1}, \ldots, z_{k_2 i}).
\end{aligned}$$

Therefore we can write

$$\begin{aligned}
& \mathbb{E}\left[\left(\max_{0\leq i\leq m'+1} \max\left\{\left|\bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right|, \left|\bar{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i-}(\theta)\right|\right\}\right)^2\right] \\
\leq & \sum_{0\leq i\leq m'+1} \left(\mathbb{E}\left[\left(\bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right)^2\right] + \mathbb{E}\left[\left(\bar{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i-}(\theta)\right)^2\right]\right) \\
\leq & \sum_{0\leq i\leq m'+1} \left(\mathbb{E}\left[\left(\bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - \hat{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right)^2\right] + \mathbb{E}\left[\left(\hat{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right)^2\right]\right) + \\
& \sum_{0\leq i\leq m'+1} \left(\mathbb{E}\left[\left(\bar{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - \hat{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta)\right)^2\right] + \mathbb{E}\left[\left(\hat{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i-}(\theta)\right)^2\right]\right) \\
& \text{since } \bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) \text{ and } \bar{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) \text{ are conditionally unbiased for } \hat{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) \text{ and } \hat{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) \\
\leq & \sum_{0\leq i\leq m'+1} \left(\mathbb{E}\left[\mathbb{E}_*\left[\left(\bar{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - \hat{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right)^2\right]\right] + \mathbb{E}\left[\left(\hat{p}_{k_2}^{\mathcal{W},\epsilon_i}(\theta) - p_{k_2}^{\mathcal{W},\epsilon_i}(\theta)\right)^2\right]\right) + \\
& \sum_{0\leq i\leq m'+1} \left(\mathbb{E}\left[\mathbb{E}_*\left[\left(\bar{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - \hat{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta)\right)^2\right]\right] + \mathbb{E}\left[\left(\tilde{p}_{k_2}^{\mathcal{W},\epsilon_i-}(\theta) - p_k^{\mathcal{W},\epsilon_i-}(\theta)\right)^2\right]\right) \\
\leq & (m'+2)\left(\frac{2}{B_2} + \frac{2}{\lfloor n/k_2 \rfloor}\right) \leq (m+2)\left(\frac{2}{B_2} + \frac{4}{n/k_2}\right).
\end{aligned}$$

By Minkowski inequality, the supremum satisfies

$$\mathbb{E}\left[\sup_{\epsilon\in[0,\delta/2]} \left|\bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - p_{k_2}^{\mathcal{W},\epsilon}(\theta)\right|\right] \leq \sqrt{(m+2)\left(\frac{2}{B_2} + \frac{4}{n/k_2}\right)} + \frac{1}{m}.$$

Choosing m such that $m \to \infty$, $m/B_2 \to 0$ and $mk_2/n \to 0$ leads to the convergence $\sup_{\epsilon\in[0,\delta/2]} \left|\bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - p_{k_2}^{\mathcal{W},\epsilon}(\theta)\right| \to 0$ in probability. Since Θ has finite cardinality and has a finite number of subsets, it also holds that

$$\sup_{\mathcal{W}\subseteq\Theta,\theta\in\mathcal{W},\epsilon\in[0,\delta/2]} \left|\bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - p_{k_2}^{\mathcal{W},\epsilon}(\theta)\right| \to 0 \text{ in probability.} \tag{42}$$

Recall the bound (54) from the proof of Lemma C.13 in Appendix D.7. Here we have the similar bound $\max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} p_{k_2}^{\mathcal{W},\epsilon}(\theta) \le \mathbb{P}\left(\widehat{\mathcal{W}}_{k_2}^{\epsilon} \not\subseteq \mathcal{W}^\delta\right)$, and hence

$$\sup_{\epsilon\in[0,\delta/2]} \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} p_k^{\mathcal{W},\epsilon}(\theta) \le \sup_{\epsilon\in[0,\delta/2]} \mathbb{P}\left(\widehat{\mathcal{W}}_{k_2}^{\epsilon} \not\subseteq \mathcal{W}^\delta\right) = \mathbb{P}\left(\widehat{\mathcal{W}}_{k_2}^{\delta/2} \not\subseteq \mathcal{W}^\delta\right).$$

We bound the probability $\mathbb{P}\left(\widehat{\mathcal{W}}_{k_2}^{\delta/2} \not\subseteq \mathcal{W}^\delta\right)$ more carefully. We let

$$\Delta_o := \min\{L(\theta') - L(\theta) : \theta, \theta' \in \Theta, L(\theta') > L(\theta)\} > 0,$$
$$\hat{L}_{k_2}(\theta) := \frac{1}{k_2}\sum_{i=1}^{k_2} l(\theta, z_i),$$

and have

$$\begin{aligned}
&\left\{\widehat{\mathcal{W}}_{k_2}^{\delta/2} \not\subseteq \mathcal{W}^\delta\right\} \\
\subseteq\ & \bigcup_{\theta,\theta'\in\mathcal{W} \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_{k_2}(\theta') \le \hat{L}_{k_2}(\theta) + \frac{\delta}{2}\right\} \\
\subseteq\ & \bigcup_{\theta,\theta'\in\Theta \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_{k_2}(\theta') - L(\theta') + L(\theta') - L(\theta) \le \hat{L}_{k_2}(\theta) - L(\theta) + \frac{\delta}{2}\right\} \\
\subseteq\ & \bigcup_{\theta,\theta'\in\Theta \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_{k_2}(\theta') - L(\theta') + \max\{\Delta, \delta\} \le \hat{L}_{k_2}(\theta) - L(\theta) + \frac{\delta}{2}\right\} \\
& \text{by the definition of } \Delta_o \\
\subseteq\ & \bigcup_{\theta,\theta'\in\Theta} \left\{\hat{L}_{k_2}(\theta') - L(\theta') + \max\left\{\Delta_o - \frac{\delta}{2}, \frac{\delta}{2}\right\} \le \hat{L}_{k_2}(\theta) - L(\theta)\right\} \\
\subseteq\ & \bigcup_{\theta,\theta'\in\Theta} \left\{\hat{L}_{k_2}(\theta') - L(\theta') \le -\max\left\{\frac{\Delta_o}{2} - \frac{\delta}{4}, \frac{\delta}{4}\right\} \text{ or } \hat{L}_{k_2}(\theta) - L(\theta) \ge \max\left\{\frac{\Delta_o}{2} - \frac{\delta}{4}, \frac{\delta}{4}\right\}\right\} \\
\subseteq\ & \bigcup_{\theta\in\Theta} \left\{\left|\hat{L}_{k_2}(\theta) - L(\theta)\right| \ge \max\left\{\frac{\Delta_o}{2} - \frac{\delta}{4}, \frac{\delta}{4}\right\}\right\} \\
\subseteq\ & \bigcup_{\theta\in\Theta} \left\{\left|\hat{L}_{k_2}(\theta) - L(\theta)\right| \ge \frac{\Delta_o}{4}\right\} \\
\subseteq\ & \left\{\sup_{\theta\in\Theta}\left|\hat{L}_{k_2}(\theta) - L(\theta)\right| \ge \frac{\Delta_o}{4}\right\},
\end{aligned}$$

where the last line holds because $\max\{\Delta_o/2 - \delta/4, \delta/4\} \ge \Delta_o/4$. This gives

$$\sup_{\epsilon\in[0,\delta/2]} \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} p_{k_2}^{\mathcal{W},\epsilon}(\theta) \le T_{k_2}\left(\frac{\Delta_o}{4}\right) \to 0 \text{ as } k_2 \to \infty.$$

We also have the trivial bound $\inf_{\epsilon\in[0,\delta/2]} \max_{\theta\in\mathcal{W}} p_{k_2}^{\mathcal{W},\epsilon}(\theta) = \max_{\theta\in\mathcal{W}} p_{k_2}^{\mathcal{W},0}(\theta) \ge 1/|\mathcal{W}|$, where the inequality comes from the fact that $\sum_{\theta\in\mathcal{W}} p_{k_2}^{\mathcal{W},0}(\theta) \ge 1$. Now choose a $\underline{k} < \infty$ such that

$$T_{k_2}\left(\frac{\Delta_o}{4}\right) \le \frac{1}{2|\Theta|} \text{ for all } k_2 \ge \underline{k},$$

and we have for all $k_2 \geq \underline{k}$ and all non-empty $\mathcal{W} \subseteq \Theta$ that

$$\begin{aligned}\inf_{\epsilon\in[0,\delta/2]}\left(\max_{\theta\in\mathcal{W}} p_{k_2}^{\mathcal{W},\epsilon}(\theta) - \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} p_{k_2}^{\mathcal{W},\epsilon}(\theta)\right) &\geq \inf_{\epsilon\in[0,\delta/2]}\max_{\theta\in\mathcal{W}} p_{k_2}^{\mathcal{W},\epsilon}(\theta) - \sup_{\epsilon\in[0,\delta/2]}\max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} p_{k_2}^{\mathcal{W},\epsilon}(\theta) \\ &\geq \frac{1}{|\mathcal{W}|} - \frac{1}{2\,|\Theta|} \geq \frac{1}{2\,|\Theta|}.\end{aligned} \tag{43}$$

Due to the uniform convergence (42), we have

$$\min_{\mathcal{W}\subseteq\Theta}\inf_{\epsilon\in[0,\delta/2]}\left(\max_{\theta\in\mathcal{W}} \bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} \bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta)\right) \to \min_{\mathcal{W}\subseteq\Theta}\inf_{\epsilon\in[0,\delta/2]}\left(\max_{\theta\in\mathcal{W}} p_{k_2}^{\mathcal{W},\epsilon}(\theta) - \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} p_{k_2}^{\mathcal{W},\epsilon}(\theta)\right)$$

in probability, and hence

$$\mathbb{P}\left(\min_{\mathcal{W}\subseteq\Theta}\inf_{\epsilon\in[0,\delta/2]}\left(\max_{\theta\in\mathcal{W}} \bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} \bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta)\right) \leq 0\right) \to 0. \tag{44}$$

Finally, we combine all the pieces to get

$$\begin{aligned}&\left\{\hat{\theta}_n \notin \Theta^{2\delta}\right\} \\ \subseteq\ &\left\{\mathcal{S}\cap\Theta^\delta = \emptyset\right\} \cup \left\{\hat{\theta}_n \notin \mathcal{S}^\delta\right\} \\ \subseteq\ &\left\{\mathcal{S}\cap\Theta^\delta = \emptyset\right\} \cup \left\{\max_{\theta\in\mathcal{S}} \bar{p}_{k_2}^{\mathcal{S},\epsilon}(\theta) - \max_{\theta\in\mathcal{S}\setminus\mathcal{S}^\delta} \bar{p}_{k_2}^{\mathcal{S},\epsilon}(\theta) \leq 0\right\} \\ \subseteq\ &\left\{\mathcal{S}\cap\Theta^\delta = \emptyset\right\} \cup \left\{\epsilon > \frac{\delta}{2}\right\} \cup \left\{\inf_{\epsilon\in[0,\delta/2]}\left(\max_{\theta\in\mathcal{S}} \bar{p}_{k_2}^{\mathcal{S},\epsilon}(\theta) - \max_{\theta\in\mathcal{S}\setminus\mathcal{S}^\delta} \bar{p}_{k_2}^{\mathcal{S},\epsilon}(\theta)\right) \leq 0\right\} \\ \subseteq\ &\left\{\mathcal{S}\cap\Theta^\delta = \emptyset\right\} \cup \left\{\epsilon > \frac{\delta}{2}\right\} \cup \left\{\min_{\mathcal{W}\subseteq\Theta}\inf_{\epsilon\in[0,\delta/2]}\left(\max_{\theta\in\mathcal{W}} \bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta) - \max_{\theta\in\mathcal{W}\setminus\mathcal{W}^\delta} \bar{p}_{k_2}^{\mathcal{W},\epsilon}(\theta)\right) \leq 0\right\}.\end{aligned}$$

By Lemma C.12 we have $\mathbb{P}\left(\mathcal{S}\cap\Theta^\delta = \emptyset\right) \to 0$ under the conditions that $\limsup_{k\to\infty} \mathcal{E}_{k,\delta} < 1$ and $k_1, n/k_1, B_1 \to \infty$. Together with the condition $\mathbb{P}\left(\epsilon \geq \delta/2\right) \to 0$ and (44), we conclude $\mathbb{P}\left(\hat{\theta}_n \notin \Theta^{2\delta}\right) \to 0$ by the union bound. □

D Proofs for Technical Lemmas

D.1 Proof of Lemma C.2

By symmetry, we have that

$$U(z_1,\ldots,z_n) = \frac{1}{n!}\sum_{\text{bijection }\pi:[n]\to[n]} \bar{\kappa}(z_{\pi(1)},\ldots,z_{\pi(n)}),$$

where we denote $[n] := \{1,\ldots,n\}$. Then, by the convexity of the exponential function and Jensen's inequality, we have that

$$\begin{aligned}\mathbb{E}\left[\exp(tU)\right] &= \mathbb{E}\left[\exp\left(t\cdot\frac{1}{n!}\sum_{\text{bijection }\pi:[n]\to[n]} \bar{\kappa}(z_{\pi(1)},\ldots,z_{\pi(n)})\right)\right] \\ &\leq \mathbb{E}\left[\frac{1}{n!}\sum_{\text{bijection }\pi:[n]\to[n]} \exp\left(t\cdot\bar{\kappa}(z_{\pi(1)},\ldots,z_{\pi(n)})\right)\right] \\ &= \mathbb{E}\left[\exp\left(t\cdot\bar{\kappa}(z_1,\ldots,z_n)\right)\right].\end{aligned}$$

This completes the proof. □

D.2 Proof of Lemma C.3

We first consider the direction $U - \mathbb{E}[\kappa] \geq \epsilon$. Let

$$\tilde{\kappa}^* := \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} \kappa^*(z_{k(i-1)+1}, \ldots, z_{ki}),$$

and

$$\tilde{\kappa} := \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} \kappa(z_{k(i-1)+1}, \ldots, z_{ki}; \omega_i),$$

where we use the shorthand notation $\hat{n} := \lfloor \frac{n}{k} \rfloor$, and ω_i's are mutually independent and also independent from $\{z_1, \ldots, z_n\}$. Then, since $\mathbb{E}[\kappa] = \mathbb{E}[\kappa^*]$, for all $t > 0$ it holds that

$$\begin{aligned}
\mathbb{P}(U - \mathbb{E}[\kappa] \geq \epsilon) &= \mathbb{P}(\exp(tU) \geq \exp(t(\mathbb{E}[\kappa] + \epsilon))) \\
&\overset{(i)}{\leq} \exp(-t(\mathbb{E}[\kappa] + \epsilon)) \cdot \mathbb{E}[\exp(tU)] \\
&\overset{(ii)}{\leq} \exp(-t(\mathbb{E}[\kappa] + \epsilon)) \cdot \mathbb{E}[\exp(t\tilde{\kappa}^*)] \\
&\overset{(iii)}{\leq} \exp(-t(\mathbb{E}[\kappa] + \epsilon)) \cdot \mathbb{E}[\exp(t\tilde{\kappa})],
\end{aligned} \tag{45}$$

where we apply the Markov inequality in (i), step (ii) is due to Lemma C.2, and step (iii) uses Jensen's inequality and the convexity of the exponential function. Due to independence, $\tilde{\kappa}$ can be viewed as the sample average of $\hat{n}$ i.i.d. Bernoulli random variables, i.e., $\tilde{\kappa} \sim \frac{1}{\hat{n}} \sum_{i=1}^{\hat{n}} \text{Bernoulli}(\mathbb{E}[\kappa])$. Hence, we have that

$$\begin{aligned}
\mathbb{E}[\exp(t\tilde{\kappa})] &= \mathbb{E}\left[\exp\left(\frac{t}{\hat{n}} \sum_{i=1}^{\hat{n}} \text{Bernoulli}(\mathbb{E}[\kappa])\right)\right] \\
&= \left(\mathbb{E}\left[\exp\left(\frac{t}{\hat{n}} \text{Bernoulli}(\mathbb{E}[\kappa])\right)\right]\right)^{\hat{n}} \\
&= \left[(1 - \mathbb{E}[\kappa]) + \mathbb{E}[\kappa] \cdot \exp\left(\frac{t}{\hat{n}}\right)\right]^{\hat{n}},
\end{aligned} \tag{46}$$

where we use the moment-generating function of Bernoulli random variables in the last line. Substituting (46) into (45), we have that

$$\mathbb{P}(U - \mathbb{E}[\kappa] \geq \epsilon) \leq \exp(-t(\mathbb{E}[\kappa] + \epsilon)) \cdot \left[(1 - \mathbb{E}[\kappa]) + \mathbb{E}[\kappa] \cdot \exp\left(\frac{t}{\hat{n}}\right)\right]^{\hat{n}} =: f(t). \tag{47}$$

Now, we consider minimizing $f(t)$ for $t > 0$. Let $g(t) = \log f(t)$, then it holds that

$$g'(t) = -(\mathbb{E}[\kappa] + \epsilon) + \frac{\mathbb{E}[\kappa] \cdot \exp\left(\frac{t}{\hat{n}}\right)}{(1 - \mathbb{E}[\kappa]) + \mathbb{E}[\kappa] \cdot \exp\left(\frac{t}{\hat{n}}\right)}.$$

By setting $g'(t) = 0$, it is easy to verify that the minimum point of $f(t)$, denoted by $t^\star$, satisfies that

$$\begin{aligned}
&\mathbb{E}[\kappa] \cdot \exp\left(\frac{t}{\hat{n}}\right) \cdot (1 - \mathbb{E}[\kappa] - \epsilon) = (1 - \mathbb{E}[\kappa]) \cdot (\mathbb{E}[\kappa] + \epsilon) \\
\Leftrightarrow \quad &\exp(t) = \left[\frac{(1 - \mathbb{E}[\kappa]) \cdot (\mathbb{E}[\kappa] + \epsilon)}{\mathbb{E}[\kappa] \cdot (1 - \mathbb{E}[\kappa] - \epsilon)}\right]^{\hat{n}}.
\end{aligned} \tag{48}$$

Substituting (48) into (47) gives

$$\begin{aligned}
\mathbb{P}(U - \mathbb{E}[\kappa] \geq \epsilon) &\leq \left(\frac{1 - \mathbb{E}[\kappa]}{1 - \mathbb{E}[\kappa] - \epsilon}\right)^{\hat{n}} \cdot \left[\frac{\mathbb{E}[\kappa] \cdot (1 - \mathbb{E}[\kappa] - \epsilon)}{(1 - \mathbb{E}[\kappa])(\mathbb{E}[\kappa] + \epsilon)}\right]^{\hat{n}(\mathbb{E}[\kappa] + \epsilon)} \\
&= \left[\left(\frac{1 - \mathbb{E}[\kappa]}{1 - \mathbb{E}[\kappa] - \epsilon}\right)^{1 - \mathbb{E}[\kappa] - \epsilon} \cdot \left(\frac{\mathbb{E}[\kappa]}{\mathbb{E}[\kappa] + \epsilon}\right)^{\mathbb{E}[\kappa] + \epsilon}\right]^{\hat{n}} \\
&= \exp(-\hat{n} \cdot D_{\text{KL}}(\mathbb{E}[\kappa] + \epsilon \| \mathbb{E}[\kappa])).
\end{aligned} \tag{49}$$

Since $n/k \leq 2\hat{n}$, the first bound immediately follows from (49).

Since $D_{\mathrm{KL}}(p\|q) = D_{\mathrm{KL}}(1-p\|1-q)$, the bound for the reverse side $U - \mathbb{E}[\kappa] \leq -\epsilon$ then follows by applying the first bound to the flipped binary kernel $1-\kappa$ and $1-U$. This completes the proof of Lemma C.3. □

D.3 Proof of Lemma C.4

To show (12), some basic calculus shows that for any fixed q, the function $g(p) := (1-p)\ln\frac{1-p}{1-q}$ is convex in p, and we have that

$$g(q) = 0, g'(q) = -1.$$

Therefore $g(p) \geq g(q) + g'(q)(p-q) = q - p$, which implies (12) immediately.

The lower bound (13) follows from

$$\begin{aligned} D_{\mathrm{KL}}(p\|q) &\geq -p\ln q - (1-p)\ln(1-q) + \min_{p\in[\gamma,1-\gamma]}\{p\ln p + (1-p)\ln(1-p)\} \\ &\geq -\gamma\ln q - \gamma\ln(1-q) - \ln 2 = -\ln(2(q(1-q))^{\gamma}). \end{aligned}$$

This completes the proof of Lemma C.4. □

D.4 Proof of Lemma C.9

By Definition C.5, we observe the following equivalence

$$\left\{\widehat{\mathcal{P}}_k^{\epsilon} \not\subseteq \mathcal{P}_k^{\delta}\right\} = \bigcup_{\theta\in\Theta\setminus\mathcal{P}_k^{\delta}} \left\{\theta \in \widehat{\mathcal{P}}_k^{\epsilon}\right\} = \bigcup_{\theta\in\Theta\setminus\mathcal{P}_k^{\delta}} \left\{\hat{p}_k(\theta) \geq \hat{p}_k\left(\theta_k^{\max}\right) - \epsilon\right\}.$$

Hence, by the union bound, it holds that

$$\mathbb{P}\left(\widehat{\mathcal{P}}_k^{\epsilon} \not\subseteq \mathcal{P}_k^{\delta}\right) \leq \sum_{\theta\in\Theta\setminus\mathcal{P}_k^{\delta}} \mathbb{P}\left(\hat{p}_k(\theta) \geq \hat{p}_k\left(\theta_k^{\max}\right) - \epsilon\right).$$

We further bound the probability $\mathbb{P}\left(\{\hat{p}_k(\theta) \geq \hat{p}_k\left(\theta_k^{\max}\right) - \epsilon\}\right)$ as follows

$$\begin{aligned} & \mathbb{P}\left(\hat{p}_k(\theta) \geq \hat{p}_k\left(\theta_k^{\max}\right) - \epsilon\right) \\ \leq\ & \mathbb{P}\left(\left\{\hat{p}_k(\theta) \geq p_k(\theta_k^{\max}) - \frac{\delta+\epsilon}{2}\right\} \cap \left\{\hat{p}_k\left(\theta_k^{\max}\right) \leq p_k(\theta_k^{\max}) - \frac{\delta-\epsilon}{2}\right\}\right) \qquad (50) \\ \leq\ & \mathbb{P}\left(\hat{p}_k(\theta) \geq p_k(\theta_k^{\max}) - \frac{\delta+\epsilon}{2}\right) + \mathbb{P}\left(\hat{p}_k\left(\theta_k^{\max}\right) \leq p_k(\theta_k^{\max}) - \frac{\delta-\epsilon}{2}\right). \end{aligned}$$

On one hand, the first probability in (50) is solely determined by and increasing in $p_k(\theta) = \mathbb{E}[\hat{p}_k(\theta)]$. On the other hand, we have $p_k(\theta) < p_k\left(\theta_k^{\max}\right) - \delta$ for every $\theta \in \Theta\setminus\mathcal{P}_k^{\delta}$ by the definition of $\mathcal{P}_k^{\delta}$. Therefore we can slightly abuse the notation to write

$$\begin{aligned} \mathbb{P}\left(\hat{p}_k(\theta) \geq \hat{p}_k\left(\theta_k^{\max}\right) - \epsilon\right) &\leq \mathbb{P}\left(\hat{p}_k(\theta) \geq p_k(\theta_k^{\max}) - \frac{\delta+\epsilon}{2}\Big| p_k(\theta) = p_k(\theta_k^{\max}) - \delta\right) \\ &\quad + \mathbb{P}\left(\hat{p}_k\left(\theta_k^{\max}\right) \leq p_k(\theta_k^{\max}) - \frac{\delta-\epsilon}{2}\right) \\ &\leq \mathbb{P}\left(\hat{p}_k(\theta) - p_k(\theta) \geq \frac{\delta-\epsilon}{2}\Big| p_k(\theta) = p_k(\theta_k^{\max}) - \delta\right) \\ &\quad + \mathbb{P}\left(\hat{p}_k\left(\theta_k^{\max}\right) - p_k(\theta_k^{\max}) \leq -\frac{\delta-\epsilon}{2}\right). \end{aligned}$$

Note that, with $\kappa(z_1,\ldots,z_k;\omega) := \mathbf{1}\left(\theta \in \mathbb{A}(z_1,\ldots,z_k;\omega)\right)$, the probability $\hat{p}_k(\theta)$ can be viewed as a U-statistic with the kernel $\kappa^*(z_1,\ldots,z_k) := \mathbb{E}[\kappa(z_1,\ldots,z_k;\omega)|z_1,\ldots,z_k]$. A similar representa-

tion holds for $\hat{p}_k\left(\theta_k^{\max}\right)$ as well. Therefore, we can apply Lemma C.3 to conclude that

$$\begin{aligned}
\mathbb{P}\left(\widehat{\mathcal{P}}_k^{\epsilon} \not\subseteq \mathcal{P}_k^{\delta}\right) &\leq \sum_{\theta \in \Theta \setminus \mathcal{P}_k^{\delta}} \mathbb{P}\left(\hat{p}_k(\theta) \geq \hat{p}_k\left(\theta_k^{\max}\right) - \epsilon\right) \\
&\leq \left|\Theta \setminus \mathcal{P}_k^{\delta}\right| \left[\mathbb{P}\left(\hat{p}_k(\theta) - p_k(\theta) \geq \frac{\delta - \epsilon}{2} \middle| p_k(\theta) = p_k(\theta_k^{\max}) - \delta\right)\right. \\
&\qquad \left. + \mathbb{P}\left(p_k\left(\theta_k^{\max}\right) - \hat{p}_k\left(\theta_k^{\max}\right) \leq -\frac{\delta - \epsilon}{2}\right)\right] \\
&\leq |\Theta| \left[\exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k\left(\theta_k^{\max}\right) - \delta + \frac{\delta - \epsilon}{2} \middle\| p_k\left(\theta_k^{\max}\right) - \delta\right)\right)\right. \\
&\qquad \left. + \exp\left(-\frac{n}{2k} \cdot D_{\mathrm{KL}}\left(p_k\left(\theta_k^{\max}\right) - \frac{\delta - \epsilon}{2} \middle\| p_k\left(\theta_k^{\max}\right)\right)\right)\right],
\end{aligned}$$

which completes the proof of Lemma C.9. □

D.5 Proof of Lemma C.10

We observe that $\bar{p}_k(\theta)$ is an conditionally unbiased estimator for $\hat{p}_k(\theta)$, i.e., $\mathbb{E}_*\left[\bar{p}_k(\theta)\right] = \hat{p}_k(\theta)$. We can express the difference between $\bar{p}_k(\theta)$ and $\bar{p}_k(\theta_k^{\max})$ as the sample average

$$\bar{p}_k(\theta) - \bar{p}_k(\theta_k^{\max}) = \frac{1}{B}\sum_{b=1}^{B}\left[\mathbb{1}(\theta \in \mathbb{A}(z_1^b, \ldots, z_k^b)) - \mathbb{1}(\theta_k^{\max} \in \mathbb{A}(z_1^b, \ldots, z_k^b))\right],$$

whose expectation is equal to $\hat{p}_k(\theta) - \hat{p}_k(\theta_k^{\max})$. We denote by

$$\mathbb{1}_\theta^* := \mathbb{1}(\theta \in \mathbb{A}(z_1^*, \ldots, z_k^*)) \text{ for } \theta \in \Theta$$

for convenience, where $(z_1^*, \ldots, z_k^*)$ represents a random subsample. Then by Bernstein's inequality, we have every $t \geq 0$ that

$$\mathbb{P}_*\left(\bar{p}_k(\theta) - \bar{p}_k(\hat{\theta}_k^{\max}) - (\hat{p}_k(\theta) - \hat{p}_k(\theta_k^{\max})) \geq t\right) \leq \exp\left(-B \cdot \frac{t^2}{2\mathrm{Var}_*(\mathbb{1}_\theta^* - \mathbb{1}_{\theta_k^{\max}}^*) + 4/3 \cdot t}\right). \tag{51}$$

Since

$$\mathrm{Var}_*(\mathbb{1}_\theta^* - \mathbb{1}_{\theta_k^{\max}}^*) \quad \leq \quad \mathbb{E}_*\left[(\mathbb{1}_\theta^* - \mathbb{1}_{\theta_k^{\max}}^*)^2\right] \leq \hat{p}_k(\theta) + \hat{p}_k(\theta_k^{\max}) \leq 2\hat{p}_k(\theta_k^{\max}),$$

and

$$\begin{aligned}
\mathrm{Var}_*(\mathbb{1}_\theta^* - \mathbb{1}_{\theta_k^{\max}}^*) &\leq \mathrm{Var}_*(1 - \mathbb{1}_\theta^* - 1 + \mathbb{1}_{\theta_k^{\max}}^*) \\
&\leq \mathbb{E}_*\left[(1 - \mathbb{1}_\theta^* - 1 + \mathbb{1}_{\theta_k^{\max}}^*)^2\right] \\
&\leq 1 - \hat{p}_k(\theta) + 1 - \hat{p}_k(\theta_k^{\max}) \leq 2(1 - \hat{p}_k(\theta)),
\end{aligned}$$

we have $\mathrm{Var}_*(\mathbb{1}_\theta^* - \mathbb{1}_{\theta_k^{\max}}^*) \leq 2\min\left\{\hat{p}_k(\theta_k^{\max}), 1 - \hat{p}_k(\theta)\right\}$. Substituting this bound to (51) and taking $t = \hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta)$ lead to

$$\begin{aligned}
&\mathbb{P}_*\left(\bar{p}_k(\theta) - \bar{p}_k(\hat{\theta}_k^{\max}) \geq 0\right) \\
\leq\ &\exp\left(-B \cdot \frac{(\hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta))^2}{4\min\left\{\hat{p}_k(\theta_k^{\max}), 1 - \hat{p}_k(\theta)\right\} + 4/3 \cdot (\hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta))}\right) \\
\leq\ &\exp\left(-B \cdot \frac{(\hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta))^2}{4\min\left\{\hat{p}_k(\theta_k^{\max}), 1 - \hat{p}_k(\theta_k^{\max})\right\} + 16/3 \cdot (\hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta))}\right) \\
\leq\ &\exp\left(-\frac{B}{6} \cdot \frac{(\hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta))^2}{\min\left\{\hat{p}_k(\theta_k^{\max}), 1 - \hat{p}_k(\theta_k^{\max})\right\} + \hat{p}_k(\theta_k^{\max}) - \hat{p}_k(\theta)}\right).
\end{aligned}$$

Therefore, we have that

$$\begin{aligned}
\mathbb{P}_*\left(\hat{\theta}_n \notin \widehat{\mathcal{P}}_k^\epsilon\right) &= \mathbb{P}_*\left(\bigcup_{\theta\in\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon}\left\{\bar{p}_k(\theta) = \max_{\theta'\in\Theta}\bar{p}_k(\theta')\right\}\right) \\
&\leq \sum_{\theta\in\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon}\mathbb{P}_*\left(\bar{p}_k(\theta) = \max_{\theta'\in\Theta}\bar{p}_k(\theta')\right) \\
&\leq \sum_{\theta\in\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon}\mathbb{P}_*\left(\bar{p}_k(\theta) \geq \bar{p}_k(\theta_k^{\max})\right) \\
&\leq \sum_{\theta\in\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon}\exp\left(-\frac{B}{6}\cdot\frac{(\hat{p}_k(\theta_k^{\max})-\hat{p}_k(\theta))^2}{\min\left\{\hat{p}_k(\theta_k^{\max}), 1-\hat{p}_k(\theta_k^{\max})\right\}+\hat{p}_k(\theta_k^{\max})-\hat{p}_k(\theta)}\right).
\end{aligned}$$

Note that the function $x^2/(\min\{\hat{p}_k(\theta_k^{\max}), 1-\hat{p}_k(\theta_k^{\max})\}+x)$ in $x\in[0,1]$ is monotonically increasing and that $\hat{p}_k(\theta_k^{\max})-\hat{p}_k(\theta)>\epsilon$ for all $\theta\in\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon$. Therefore, we can further bound the probability as

$$\mathbb{P}_*\left(\hat{\theta}_n \notin \widehat{\mathcal{P}}_k^\epsilon\right) \quad\leq\quad \left|\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon\right|\cdot\exp\left(-\frac{B}{6}\cdot\frac{\epsilon^2}{\min\left\{\hat{p}_k^{\max}, 1-\hat{p}_k^{\max}\right\}+\epsilon}\right).$$

Noting that $\left|\Theta\setminus\widehat{\mathcal{P}}_k^\epsilon\right| \leq |\Theta|$ completes the proof of Lemma C.10. □

D.6 Proof of Lemma C.12

Let $(z_1^*,\ldots,z_k^*)$ be a random subsample and $\mathbb{P}_*$ be the probability with respect to the subsampling randomness conditioned on the data and the algorithmic randomness. Consider the two probabilities

$$\mathbb{P}\left(\mathcal{A}(z_1,\ldots,z_k)\in\Theta^\delta\right),\ \mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right).$$

We have $1-\mathcal{E}_{k,\delta} = \mathbb{P}\left(\mathcal{A}(z_1,\ldots,z_k)\in\Theta^\delta\right)$, and the conditional probability

$$\mathbb{P}\left(\mathcal{S}\cap\Theta^\delta=\emptyset\middle|\mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right)\right) = \left(1-\mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right)\right)^{B_1}.$$

Therefore we can write

$$\begin{aligned}
\mathbb{P}\left(\mathcal{S}\cap\Theta^\delta=\emptyset\right) &= \mathbb{E}\left[\left(1-\mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right)\right)^{B_1}\right] \\
&\leq \mathbb{P}\left(\mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right)<\frac{1-\mathcal{E}_{k,\delta}}{e}\right)+\left(1-\frac{1-\mathcal{E}_{k,\delta}}{e}\right)^{B_1}, \quad (52)
\end{aligned}$$

where e is the base of the natural logarithm. Applying Lemma C.3 with $\kappa(z_1,\ldots,z_k;\omega) := \mathbb{1}\left(\mathcal{A}(z_1,\ldots,z_k;\omega)\in\Theta^\delta\right)$ gives

$$\mathbb{P}\left(\mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right)<\frac{1-\mathcal{E}_{k,\delta}}{e}\right)\leq\exp\left(-\frac{n}{2k}\cdot D_{\mathrm{KL}}\left(\frac{1-\mathcal{E}_{k,\delta}}{e}\middle\|1-\mathcal{E}_{k,\delta}\right)\right).$$

Further applying the bound (12) from Lemma C.4 to the KL divergence on the right-hand side leads to

$$D_{\mathrm{KL}}\left(\frac{1-\mathcal{E}_{k,\delta}}{e}\middle\|1-\mathcal{E}_{k,\delta}\right)\geq\frac{1-\mathcal{E}_{k,\delta}}{e}\ln\frac{1}{e}+1-\mathcal{E}_{k,\delta}-\frac{1-\mathcal{E}_{k,\delta}}{e}=\left(1-\frac{2}{e}\right)(1-\mathcal{E}_{k,\delta}),$$

and

$$
\begin{aligned}
& D_{\mathrm{KL}}\left(\frac{1-\mathcal{E}_{k,\delta}}{e}\middle\|1-\mathcal{E}_{k,\delta}\right) \\
= \;& D_{\mathrm{KL}}\left(1-\frac{1-\mathcal{E}_{k,\delta}}{e}\middle\|\mathcal{E}_{k,\delta}\right) \\
\geq \;& \left(1-\frac{1-\mathcal{E}_{k,\delta}}{e}\right)\ln\frac{1-(1-\mathcal{E}_{k,\delta})/e}{\mathcal{E}_{k,\delta}}-(1-\mathcal{E}_{k,\delta})+\frac{1-\mathcal{E}_{k,\delta}}{e} \text{ by bound (12)} \\
\geq \;& \left(1-\frac{1-\mathcal{E}_{k,\delta}}{e}\right)\ln\left(1-\frac{1-\mathcal{E}_{k,\delta}}{e}\right)-\left(1-\frac{1}{e}\right)\ln\mathcal{E}_{k,\delta}-1+\frac{1}{e} \\
\geq \;& \left(1-\frac{1}{e}\right)\ln\left(1-\frac{1}{e}\right)-\left(1-\frac{1}{e}\right)\ln\mathcal{E}_{k,\delta}-1+\frac{1}{e} \\
= \;& \left(1-\frac{1}{e}\right)\ln\frac{e-1}{e^2\mathcal{E}_{k,\delta}}.
\end{aligned}
$$

Combining the two bounds for the KL divergence we have

$$
\begin{aligned}
& \mathbb{P}\left(\mathbb{P}_*\left(\mathcal{A}(z_1^*,\ldots,z_k^*)\in\Theta^\delta\right)<\frac{1-\mathcal{E}_{k,\delta}}{e}\right) \\
\leq\; & \min\left\{\exp\left(-\frac{n}{2k}\cdot\left(1-\frac{2}{e}\right)(1-\mathcal{E}_{k,\delta})\right),\left(\frac{e^2\mathcal{E}_{k,\delta}}{e-1}\right)^{(1-1/e)\frac{n}{2k}}\right\}.
\end{aligned}
$$

Note that the second term on the right-hand side of (52) satisfies that $(1-(1-\mathcal{E}_{k,\delta})/e)^{B_1} \leq \exp\left(-B_1(1-\mathcal{E}_{k,\delta})/e\right)$. Thus, we derive that

$$
\begin{aligned}
& \mathbb{P}\left(\mathcal{S}\cap\Theta^\delta=\emptyset\right) \\
\leq\; & \min\left\{\exp\left(-\frac{n}{2k}\cdot\left(1-\frac{2}{e}\right)(1-\mathcal{E}_{k,\delta})\right),\left(\frac{e^2\mathcal{E}_{k,\delta}}{e-1}\right)^{(1-1/e)\frac{n}{2k}}\right\}+\exp\left(-\frac{B_1(1-\mathcal{E}_{k,\delta})}{e}\right) \\
\leq\; & \min\left\{\exp\left(-\frac{1-2/e}{1-1/e}\cdot(1-\mathcal{E}_{k,\delta})\right),\frac{e^2\mathcal{E}_{k,\delta}}{e-1}\right\}^{(1-1/e)\frac{n}{2k}}+\exp\left(-\frac{B_1(1-\mathcal{E}_{k,\delta})}{e}\right).
\end{aligned}
$$

The conclusion then follows by setting C_4, C_5, C_6, C_7 to be the appropriate constants. □

D.7 Proof of Lemma C.13

Let $\hat{L}_k(\theta) := \frac{1}{k}\sum_{i=1}^k l(\theta, z_i)$. Let θ^* be an optimal solution of (1). We have

$$
\max_{\theta\in\Theta} p_k(\theta) \geq p_k(\theta^*) = \mathbb{P}\left(\theta^*\in\widehat{\Theta}_k^\epsilon\right) \geq \mathbb{P}\left(\Theta^0\subseteq\widehat{\Theta}_k^\epsilon\right).
$$

To bound the probability on the right hand side, we write

$$
\begin{aligned}
\left\{\Theta^0\not\subseteq\widehat{\Theta}_k^\epsilon\right\} \;\subseteq\; & \bigcup_{\theta\in\Theta^0,\theta'\in\Theta}\left\{\hat{L}_k(\theta)>\hat{L}_k(\theta')+\epsilon\right\} \\
= \; & \bigcup_{\theta\in\Theta^0,\theta'\in\Theta}\left\{\hat{L}_k(\theta)-L(\theta)>\hat{L}_k(\theta')-L(\theta')+L(\theta')-L(\theta)+\epsilon\right\} \\
\subseteq \; & \bigcup_{\theta\in\Theta^0,\theta'\in\Theta}\left\{\hat{L}_k(\theta)-L(\theta)>\hat{L}_k(\theta')-L(\theta')+\epsilon\right\} \\
\subseteq \; & \bigcup_{\theta\in\Theta^0,\theta'\in\Theta}\left\{\hat{L}_k(\theta)-L(\theta)>\frac{\epsilon}{2}\text{ or }\hat{L}_k(\theta')-L(\theta')<-\frac{\epsilon}{2}\right\} \\
\subseteq \; & \bigcup_{\theta\in\Theta}\left\{\left|\hat{L}_k(\theta)-L(\theta)\right|>\frac{\epsilon}{2}\right\} \\
= \; & \left\{\max_{\theta\in\Theta}\left|\hat{L}_k(\theta)-L(\theta)\right|>\frac{\epsilon}{2}\right\},
\end{aligned}
$$

therefore

$$\max_{\theta\in\Theta} p_k(\theta) \geq \mathbb{P}\left(\max_{\theta\in\Theta}\left|\hat{L}_k(\theta) - L(\theta)\right| \leq \frac{\epsilon}{2}\right) \geq 1 - T_k\left(\frac{\epsilon}{2}\right). \tag{53}$$

This proves (35). To bound the other term $\max_{\theta\in\Theta\setminus\Theta^\delta} p_k(\theta)$, for any $\theta \in \Theta\setminus\Theta^\delta$ it holds that

$$p_k(\theta) = \mathbb{P}\left(\theta \in \widehat{\Theta}_k^\epsilon\right) \leq \mathbb{P}\left(\widehat{\Theta}_k^\epsilon \not\subseteq \Theta^\delta\right), \tag{54}$$

and hence $\max_{\theta\in\Theta\setminus\Theta^\delta} p_k(\theta) \leq \mathbb{P}\left(\widehat{\Theta}_k^\epsilon \not\subseteq \Theta^\delta\right)$. To bound the latter, we have

$$\begin{aligned}
\left\{\widehat{\Theta}_k^\epsilon \not\subseteq \Theta^\delta\right\} &\subseteq \bigcup_{\theta,\theta'\in\Theta \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_k(\theta') \leq \hat{L}_k(\theta) + \epsilon\right\} \\
&= \bigcup_{\theta,\theta'\in\Theta \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_k(\theta') - L(\theta') + L(\theta') - L(\theta) \leq \hat{L}_k(\theta) - L(\theta) + \epsilon\right\} \\
&\subseteq \bigcup_{\theta,\theta'\in\Theta \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_k(\theta') - L(\theta') + \delta < \hat{L}_k(\theta) - L(\theta) + \epsilon\right\} \\
&\subseteq \bigcup_{\theta,\theta'\in\Theta \text{ s.t. } L(\theta')-L(\theta)>\delta} \left\{\hat{L}_k(\theta') - L(\theta') < -\frac{\delta-\epsilon}{2} \text{ or } \hat{L}_k(\theta) - L(\theta) > \frac{\delta-\epsilon}{2}\right\} \\
&\subseteq \bigcup_{\theta\in\Theta}\left\{\left|\hat{L}_k(\theta) - L(\theta)\right| > \frac{\delta-\epsilon}{2}\right\} \\
&= \left\{\max_{\theta\in\Theta}\left|\hat{L}_k(\theta) - L(\theta)\right| > \frac{\delta-\epsilon}{2}\right\},
\end{aligned}$$

therefore

$$\max_{\theta\in\Theta\setminus\Theta^\delta} p_k(\theta) \leq \mathbb{P}\left(\max_{\theta\in\Theta}\left|\hat{L}_k(\theta) - L(\theta)\right| > \frac{\delta-\epsilon}{2}\right) \leq T_k\left(\frac{\delta-\epsilon}{2}\right). \tag{55}$$

This immediately gives (36). (37) is obvious given (35) and (36). □

E Improving Tail Decay in Linear Regression

In this section, we present another example demonstrating that our algorithm is capable of turning a polynomial-tailed base learner into exponential. We consider the linear regression problem

$$\min_{\theta\in[-1,1]} \mathbb{E}\big[(x\theta - y)^2\big], \tag{56}$$

where the data $\{(x_i, y_i)\}_{i=1}^n$ are i.i.d. samples such that $x_i \in \{-1, 1\}$, and $y_i = x_i\theta^* + \epsilon_i$.

Assumption E.1. The unknown true coefficient of problem (56) is $\theta^* = 0$. The random variables $\{\epsilon_i\}_{i=1}^n$ are i.i.d. distributed with zero mean and symmetric with respect to 0. The second and forth moments of ϵ_i are finite, denoted as $\sigma^2 = \mathbb{E}\left[\epsilon_i^2\right]$ and $\mu_4 = \mathbb{E}\left[\epsilon_i^4\right]$. Moreover, there exist constants $C > 0$ and $\alpha > 0$ such that $P(\epsilon_i > t) > C(t+1)^{-\alpha}$ for all $t > 0$, i.e., ϵ_i has a polynomial tail.

Under the setting described by (56) and Assumption E.1, the least-squares estimator of θ is given by

$$\theta_n^{LS} = \mathcal{P}_{[-1,1]}\left(\frac{\sum_{i=1}^n x_i y_i}{\sum_{i=1}^n x_i^2}\right) = \mathcal{P}_{[-1,1]}\left(\frac{\sum_{i=1}^n x_i \epsilon_i}{\sum_{i=1}^n x_i^2} + \theta^*\right) = \mathcal{P}_{[-1,1]}\left(\frac{\sum_{i=1}^n x_i \epsilon_i}{n}\right),$$

where $\mathcal{P}_{[-1,1]}(\cdot)$ denotes the projection operator onto the interval $[-1, 1]$. Since the true coefficient $\theta^* = 0$, for any estimator $\hat{\theta}$ that takes values between $[-1, 1]$, its excess risk is equal to $(\hat{\theta})^2$. For instance, the excess risk of θ_n^{LS} can be expressed as

$$\left(\theta_n^{LS}\right)^2 = \min\left\{\left(\frac{\sum_{i=1}^n x_i \epsilon_i}{n}\right)^2, 1\right\}.$$

Theorem E.2. *Under Assumption E.1, the followings hold true.*

- *The excess risk of the least-squares estimator θ_n^{LS} exhibits a polynomial tail in n. Specifically, for every $\delta \in (0,1)$ and $n > 1$, it holds that*

$$\mathbb{P}\left((\theta_n^{LS})^2 > \delta\right) > C(n\sqrt{\delta}+1)^{-\alpha}. \tag{57}$$

- *Under our ensemble method, the excess risk of the output estimator $\hat{\theta}_n$ has an exponentially decreasing tail.*

E.1 Proof of Theorem E.2

We first show the polynomial tail of excess risk for the least-squares estimator θ_n^{LS}. For $k \le n$, let $\bar{\epsilon}_k := 1/k \cdot \sum_{i=1}^k \epsilon_i$ be the sample average of the first k noise terms. Then, it holds that $\bar{\epsilon}_{k+1} = (k\bar{\epsilon}_k + \epsilon_{k+1})/(k+1)$.

For every $\delta \in (0,1)$ and $n > 1$, we have that

$$\mathbb{P}\left((\theta_n^{LS})^2 > \delta\right) = \mathbb{P}\left(\left(\frac{\sum_{i=1}^n x_i\epsilon_i}{n}\right)^2 > \delta\right) = \mathbb{P}\left(\left(\frac{\sum_{i=1}^n \epsilon_i}{n}\right)^2 > \delta\right) = 2\mathbb{P}\big(\bar{\epsilon}_n > \sqrt{\delta}\big), \tag{58}$$

where we used the symmetry of ϵ_i and that $x_i \in \{-1,1\}$. Then, using the recursive relation between $\bar{\epsilon}_n$ and $\bar{\epsilon}_{n-1}$, we can further show that

$$\begin{aligned}
\mathbb{P}\big(\bar{\epsilon}_n > \sqrt{\delta}\big) &\ge \mathbb{P}\left(\frac{(n-1)\bar{\epsilon}_{n-1}}{n} \ge 0 \text{ and } \frac{\epsilon_n}{n} > \sqrt{\delta}\right) \\
&= \mathbb{P}\left(\frac{(n-1)\bar{\epsilon}_{n-1}}{n} \ge 0\right) \cdot \mathbb{P}\left(\frac{\epsilon_n}{n} > \sqrt{\delta}\right) \\
&> \mathbb{P}\left(\bar{\epsilon}_{n-1} \ge 0\right) \cdot C(n\sqrt{\delta}+1)^{-\alpha},
\end{aligned} \tag{59}$$

where the second line is due to the independence between $\bar{\epsilon}_{n-1}$ and ϵ_n in the second line, and the last line uses Assumption E.1. Since each ϵ_i is symmetric, we have $\mathbb{P}\left(\bar{\epsilon}_{n-1} \ge 0\right) = 1/2$. Hence, the proof of (57) is completed by combining (58) and (59).

Now, we proceed to show the exponential tail of excess risk for our ensemble method (Algorithm 2), where the proof is based on Theorem C.11, i.e., the formal version of Theorem 2.3. To apply the bound (33) from Theorem C.11, we need to derive upper bounds for the following two quantities: the empirical process tail $T_k(\cdot)$, and the excess risk tail of the base learner, i.e., the least-squares estimator θ_k^{LS} with k samples.

For any $t > 0$, we can show that the empirical process tail satisfies that

$$\begin{aligned}
T_k(t) &= \mathbb{P}\left(\sup_{\theta\in[-1,1]} \left|\frac{1}{k}\sum_{i=1}^k (x_i\theta - y_i)^2 - \mathbb{E}[(x\theta - y)^2]\right| > t\right) \\
&= \mathbb{P}\left(\sup_{\theta\in[-1,1]} \left|\frac{1}{k}\sum_{i=1}^k (x_i\theta - \epsilon_i)^2 - (\theta^2+\sigma^2)\right| > t\right) \\
&= \mathbb{P}\left(\sup_{\theta\in[-1,1]} \left|\frac{1}{k}\sum_{i=1}^k \epsilon_i^2 - \sigma^2 - 2\theta\cdot\frac{1}{k}\sum_{i=1}^k x_i\epsilon_i\right| > t\right) \\
&\le \mathbb{P}\left(\left|-\sigma^2 + \frac{1}{k}\sum_{i=1}^k \epsilon_i^2 - \frac{2}{k}\sum_{i=1}^k x_i\epsilon_i\right| > t\right) + \mathbb{P}\left(\left|-\sigma^2 + \frac{1}{k}\sum_{i=1}^k \epsilon_i^2 + \frac{2}{k}\sum_{i=1}^k x_i\epsilon_i\right| > t\right),
\end{aligned} \tag{60}$$

where the last inequality uses the union bound and the observation that the maximum of the absolute value term is achieved either at $\theta = 1$ or $\theta = -1$. Using the symmetry of ϵ_i and the fact that

$x_i \in \{-1, 1\}$, we further derive from (60) that

$$\begin{aligned} T_k(t) &\leq 2\mathbb{P}\left(\left|\frac{1}{k}\sum_{i=1}^{k}\epsilon_i^2 - \sigma^2 - \frac{2}{k}\sum_{i=1}^{k}\epsilon_i\right| > t\right) \\ &\leq 2\mathbb{P}\left(\left|\frac{1}{k}\sum_{i=1}^{k}\epsilon_i^2 - \sigma^2\right| > \frac{t}{2}\right) + 2\mathbb{P}\left(\left|\frac{2}{k}\sum_{i=1}^{k}\epsilon_i\right| > \frac{t}{2}\right) \\ &\leq \frac{8\mu_4}{kt^2} + \frac{32\sigma^2}{kt^2}, \end{aligned} \tag{61}$$

where we apply the union bound again in the second line and use the Markov's inequality in the last line.

Now, we assess the excess risk tail of the least-squares estimator θ_k^{LS}. For $\delta < 1$, similar as (58), we can apply the Markov's inequality to show that

$$\mathcal{E}_{k,\delta} = \mathbb{P}\left((\theta_k^{LS})^2 > \delta\right) = \mathbb{P}\left(\left(\frac{\sum_{i=1}^{k}\epsilon_i}{k}\right)^2 > \delta\right) \leq \frac{\sigma^2}{k\delta}. \tag{62}$$

By instantiating (33) in Theorem C.11 with the tail bounds on $T_k(t)$ and $\mathcal{E}_{k,\delta}$ given by (61) and (62), we can finally obtain the following tail bound on the excess risk of our estimator $\hat{\theta}_n$:

$$\begin{aligned} \mathbb{P}\left((\hat{\theta}_n)^2 > \delta\right) \leq B_1 &\left(3\min\left\{e^{-2/5}, C_1\frac{32(\mu_4+4\sigma^2)}{k_2\min\{\underline{\epsilon}, \delta-\overline{\epsilon}\}^2}\right\}^{\frac{n}{2C_2k_2}} + e^{-B_2/C_3}\right) \\ &+ \min\left\{e^{-\left(1-\frac{\sigma^2}{k_1\delta}\right)/C_4}, C_5\frac{\sigma^2}{k_1\delta}\right\}^{\frac{n}{2C_6k_1}} + e^{-B_1\left(1-\frac{\sigma^2}{k_1\delta}\right)/C_7}, \end{aligned} \tag{63}$$

for every $k_1, k_2 \leq n$ and $\delta \in (0, 1)$ such that $\delta > \overline{\epsilon}$, $\frac{32(\mu_4+4\sigma^2)}{k_2(\delta-\overline{\epsilon})^2} + \frac{32(\mu_4+4\sigma^2)}{k_2\underline{\epsilon}^2} < \frac{1}{5}$, and $\frac{\sigma^2}{k_1\delta} < 1$. Note that these conditions guarantee that the upper bound in (63) is meaningful and that $T_{k_2}((\delta - \overline{\epsilon})/2) + T_{k_2}(\underline{\epsilon}/2) < 1/5$, as required by Theorem 2.3. Therefore, we conclude that the excess risk for the output solution $\hat{\theta}_n$ of our ensemble method has an exponential tail, which completes the proof of Theorem E.2. □

F Additional Numerical Experiments

This section supplements Section 3. We first provide details for the architecture of the neural networks in Section F.1, and the considered stochastic programs in Section F.2. Section F.3 presents a comprehensive profiling of hyperparameters of our methods, and Section F.4 provides additional experimental results that evaluate our algorithms from various perspectives.

F.1 MLP Architecture

The input layer of our MLPs has the same number of neurons as the input dimension, and the output layer is a single neuron that gives the final prediction. All activations are ReLU. The architecture of hidden layers is as follows under different numbers of hidden layers H:

- $H = 2$: Each hidden layer has 50 neurons.
- $H = 4$: There are 50, 300, 300, 50 neurons from the first to the fourth hidden layer.
- $H = 6$: There are 50, 300, 500, 500 300, 50 neurons from the first to the sixth hidden layer.
- $H = 8$: There are 50, 300, 500, 800, 800 500 300, 50 neurons from the first to the eighth hidden layer.

F.2 Stochastic Programming Problems

Resource Allocation [86]. The decision maker wants to choose a subset of m projects. A quantity q of low-cost resource is available to be allocated, and any additional resource can be obtained at an incremental unit cost c. Each project i has an expected reward r_i. The amount of resource required by each project i is a random variable, denoted by W_i. We can formulate the problem as

$$\max_{\theta \in \{0,1\}^m} \left\{ \sum_{i=1}^{m} r_i \theta_i - c\mathbb{E}\left[\sum_{i=1}^{m} W_i \theta_i - q \right]^+ \right\}. \tag{64}$$

In the experiment, we consider the three-product scenario, i.e., $m = 3$, and assume that the random variable W_i follows the Pareto distribution.

Supply Chain Network Design [61, Chapter 1.5]. Consider a network of suppliers, processing facilities, and customers, where the goal is to optimize the overall supply chain efficiency. The supply chain design problem can be formulated as a two-stage stochastic optimization problem

$$\min_{\theta \in \{0,1\}^{|P|}} \sum_{p \in P} c_p \theta_p + \mathbb{E}[Q(\theta, z)], \tag{65}$$

where P is the set of processing facilities, c_p is the cost of opening facility p, and z is the vector of (random) parameters, i.e., (h, q, d, s, R, M) in (66). Function $Q(\theta, z)$ represents the total processing and transportation cost, and it is equal to the optimal objective value of the following second-stage problem:

$$\begin{aligned} \min_{y \geq 0, z \geq 0} \quad & q^\top y + h^\top z \\ \text{s.t.} \quad & Ny = 0, \\ & Cy + z \geq d, \\ & Sy \leq s, \\ & Ry \leq M\theta, \end{aligned} \tag{66}$$

where N, C, S are appropriate matrices that describe the network flow constraints. More details about this example can be found in [61, Chapter 1.5]. In our experiment, we consider the scenario of 3 suppliers, 2 facilities, 3 consumers, and 5 products. We choose supply s and demand d as random variables that follow the Pareto distribution.

Maximum Weight Matching and Stochastic Linear Program. We explore both the maximum weight matching problem and the linear program that arises from it. Let $G = (V, E)$ be a general graph, where each edge $e \in E$ is associated with a (possibly) random weight w_e. For each node $v \in V$, denote $E(v)$ as the set of edges incident to v. Based on this setup, we consider the following linear program

$$\begin{aligned} \max_{\theta \in [0,1]^{|E|}} \quad & \mathbb{E}\left[\sum_{e \in E} w_e \theta_e \right] \\ \text{subject to} \quad & \sum_{e \in E(v)} a_e \theta_e \leq 1, \qquad \forall v \in V, \end{aligned} \tag{67}$$

where a_e is some positive coefficient. When $a_e = 1$ for all $e \in E$ and θ is restricted to the discrete set $\{0, 1\}^{|E|}$, (67) is equivalent to the maximum weight matching problem. For the maximum weight matching, we consider a complete bipartite graph with 5 nodes on each side (the dimension is 25). The weights of nine edges are Pareto distributed and the remaining are prespecified constants. For the linear programming problem, we consider a 28-dimensional instance (the underlying graph is an 8-node complete graph), where all w_e follows the Pareto distribution.

Mean-Variance Portfolio Optimization. Consider constructing a portfolio based on m assets. Each asset i has a rate of return r_i that is random with mean μ_i. The goal is to minimize the variance of the portfolio while ensuring that the expected rate of return surpasses a target level b. The problem can be formulated as

$$\begin{aligned} \min_{\theta} \quad & \mathbb{E}\left[(\textstyle\sum_{i=1}^{m} (r_i - \mu_i)\theta_i)^2 \right] \\ \text{subject to} \quad & \textstyle\sum_{i=1}^{m} \mu_i \theta_i \geq b, \\ & \textstyle\sum_{i=1}^{m} \theta_i = 1, \\ & \theta_i \geq 0, \quad \forall i = 1, \ldots, m, \end{aligned} \tag{68}$$

where θ is the decision variable and each μ_i is assumed known. In the experiment, we consider a scenario with 10 assets, i.e., $m = 10$, where each rate of return r_i is a linear combination of the rates of return of 100 underlying assets in the form $r_i = \tilde{r}_{10(i-1)+1}/2 + \sum_{j=1}^{100} \tilde{r}_j/200$. Each of these underlying assets has a Pareto rate of return $\tilde{r}_j, j = 1, \ldots, 100$.

F.3 Hyperparameter Profiling

We test the effect of different hyperparameters in our ensemble methods, including subsample sizes k, k_1, k_2, ensemble sizes B, B_1, B_2, and threshold ϵ. Throughout this profiling stage, we use the sample average approximation (SAA) as the base algorithm. To profile the effect of subsample sizes and ensemble sizes, we consider the resource allocation problem.

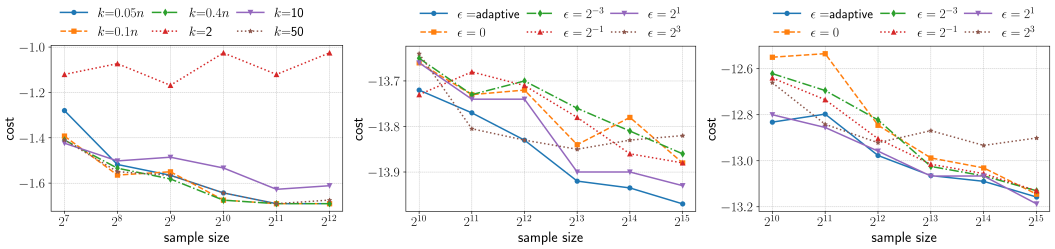
Subsample Size. We explored scenarios where k (equivalently k_1 and k_2) is both dependent on and independent of the total sample size n (see Figures 6a, 7a, and 7b). The results suggest that a constant k generally suffices, although the optimal k varies by problem instance. For example, Figures 7a and 7b show that $k = 2$ yields the best performance; increasing k degrades results. Conversely, in Figure 6a, $k = 2$ proves inadequate, with larger k delivering good results. The underlying reason is that the effective performance of MoVE requires $\theta^* \in \arg\max_{\theta \in \Theta} p_k(\theta)$. In the former, this is achieved with only two samples, enabling MoVE to identify θ^* with a subsample size of 2. For the latter, a higher number of samples is required to meet this condition, explaining the suboptimal performance at $k = 2$. In Figure 8, we simulate $p_k(\theta)$ for the two cases, which further explains the influence of the subsample size.

Ensemble Size. In Figure 9, we illustrate the performance of MoVE and ROVE under different B, B_1, B_2, where we set $k = k_1 = k_2 = 10$ and $\epsilon = 0.005$. From the figure, we find that the performance of our ensemble methods is improving in the ensemble sizes.

Threshold ϵ. The optimal choice of ϵ in ROVE and ROVEs is problem-dependent and related to the number of (near) optimal solutions. This dependence is illustrated by the performance of ROVE shown in Figures 6b and 6c. Hence, we propose an adaptive strategy defined as follows: Let $g(\epsilon) := 1/B_2 \cdot \sum_{b=1}^{B_2} \mathbb{1}(\hat{\theta}_n(\epsilon) \in \widehat{\Theta}_{k_2}^{\epsilon,b})$, where we use $\hat{\theta}_n(\epsilon)$ to emphasize the dependency of $\hat{\theta}_n$ on ϵ. Then, we select $\epsilon^* := \min\{\epsilon : g(\epsilon) \geq 1/2\}$. By definition, $g(\epsilon)$ is the proportion of times that $\hat{\theta}_n(\epsilon)$ is included in the "near optimum set" $\widehat{\Theta}_{k_2}^{\epsilon,b}$. The choice of ϵ^* makes it more likely for the true optimal solution to be included in the "near optimum set", instead of being ruled out by suboptimal solutions. Practically, ϵ^* can be efficiently determined using a binary search as an intermediate step between Phases I and II. To prevent data leakage, we compute ϵ^* using $\mathbf{z}_{1:\lfloor \frac{n}{2} \rfloor}$ (Phase I data) for ROVEs. From Figure 6, we observe that this adaptive strategy exhibits decent performance for all scenarios. Similar behaviors can also be observed for ROVEs in Figure 10.

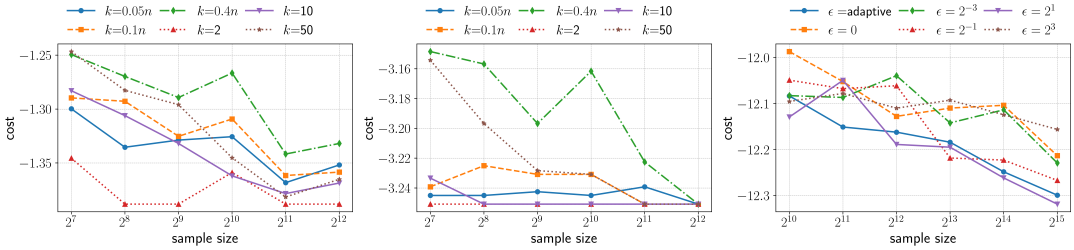
Recommended Configurations. Based on the profiling results, we summarize the recommended configurations used in all other experiments presented in the paper (unless specified otherwise):

- For discrete space Θ, use $k = \max(10, n/200), B = 200$ for MoVE, and $k_1 = k_2 = \max(10, n/200), B_1 = 20, B_2 = 200$ for ROVE and ROVEs.
- For continuous space Θ, use $k_1 = \max(30, n/2), k_2 = \max(30, n/200), B_1 = 50, B_2 = 200$ for ROVE and ROVEs.
- The ϵ in ROVE and ROVEs is selected such that $\max_{\theta \in \mathcal{S}} (1/B_2) \sum_{b=1}^{B_2} \mathbb{1}(\theta \in \widehat{\Theta}_{k_2}^{\epsilon,b}) \approx 1/2$.



(a) Profiling for k (MoVE). (b) Profiling for ϵ (multiple optima). (c) Profiling for ϵ (unique optima).

Figure 6: Profiling for subsample size k and threshold ϵ. (a): Resource allocation problem, where $B = 200$; (b) and (c): Linear program, where $k_1 = k_2 = \max(10, 0.005n)$, $B_1 = 20$, and $B_2 = 200$.



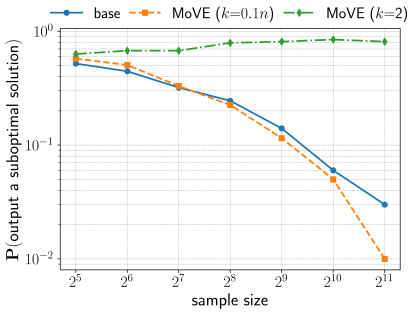
(a) Profiling for k (instance 1). (b) Profiling for k (instance 2). (c) Profiling for ϵ (near optima).

Figure 7: Profiling results for subsample size k and threshold ϵ. (a) and (b): Resource allocation problem using MoVE, where $B = 200$; (c): Linear program with multiple near optima using ROVE, where $k_1 = k_2 = \max(10, 0.005n)$, $B_1 = 20$, and $B_2 = 200$.

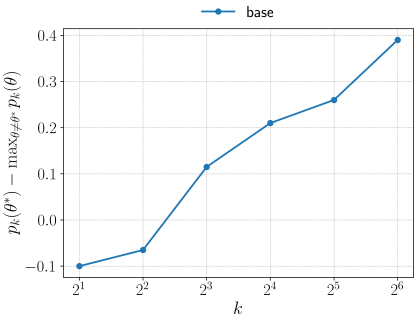
F.4 Additional Experimental Results

Here, we present additional figures that supplement the experiments and discussions in Section 3. Recall that MoVE refers to Algorithm 1, ROVE refers to Algorithm 2 without data splitting, and ROVEs refers to Algorithm 2 with data splitting. We briefly introduce each figure below and refer the reader to the figure caption for detailed discussions. Figures 11-19 all follow the recommended configuration listed in Section 3.

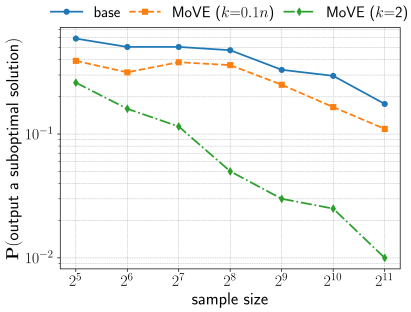
- Figure 11 supplements the results in Figure 1 with MLPs with $H = 2, 4$ hidden layers.
- Figure 12 supplements the results in Figure 3 with a different synthetic example than in Section 3.1.
- Figure 13 supplements the results in Figure 4 with three other real datasets: *Wine Quality* [79] *Online News* [83], *Appliances Energy* [73]. ROVE and the base algorithm perform comparably on these three datasets, potentially because they are lighter tailed than those in Figure 4.
- Figure 14 shows results for MLP regression on a slightly different example than in Section 3.1.
- Figures 15 and 16 show results for regression with least squares regression and Ridge regression as the base learning algorithms respectively.
- Figure 17 shows results on the stochastic linear program example with light-tailed uncertainties.
- Figure 18 contains additional results on the supply chain network design example for different choices of hyperparameters and a different problem instance with strong correlation between solutions.
- In Figure 19, we apply our ensemble methods to resource allocation and maximum weight matching using DRO with Wasserstein metric as the base algorithm. This result, together with Figure 5 where the base algorithm is SAA, demonstrates that the benefit of our ensemble methods is agnostic to the underlying base algorithm.
- In Figure 20, we simulate the generalization sensitivity $\bar{\eta}_{k,\delta}$, defined in (27), which explains the superior performance of ROVE and ROVEs in the presence of multiple optimal solutions.



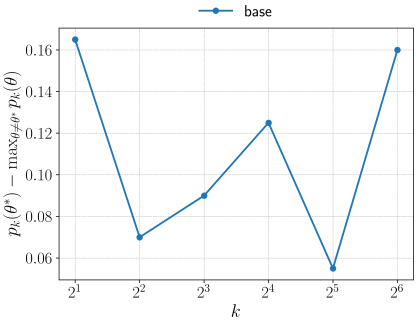
(a) Figure 6a: suboptimal probability.



(b) Figure 6a: $p_k(\theta^*) - \max_{\theta \neq \theta^*} p_k(\theta)$.

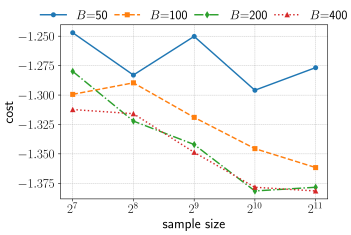


(c) Figure 7a: suboptimal probability.

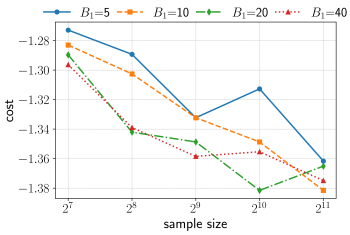


(d) Figure 7a: $p_k(\theta^*) - \max_{\theta \neq \theta^*} p_k(\theta)$.

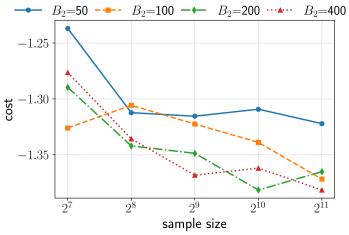
Figure 8: Performance of MoVE ($B = 200$) in resource allocation, corresponding to the two instances in Figures 6a and 7a. Subfigures (b) and (d) explain the behaviors of MoVE with different subsample sizes k: In (b), we find that $p_k(\theta^*) - \max_{\theta \neq \theta^*} p_k(\theta) < 0$ for $k \leq 4$, which results in the poor performance of MoVE with $k = 2$ in Figure 6a; In (d), we have $p_2(\theta^*) - \max_{\theta \neq \theta^*} p_2(\theta) \approx 0.165$, thereby enabling MoVE to distinguish the optimal solution only using subsamples of size two, which results in the good performance of MoVE with $k = 2$ in Figure 7a.



(a) MoVE.

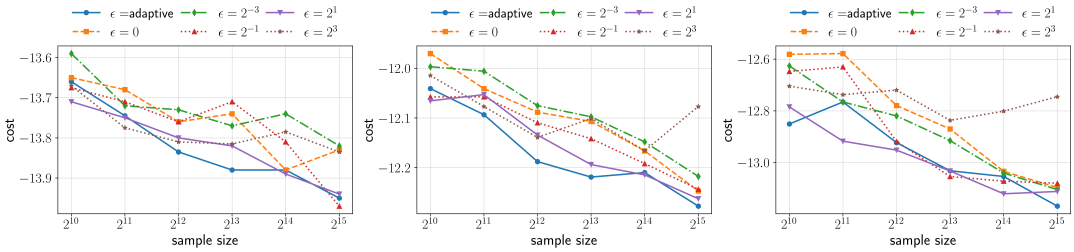


(b) ROVE with $B_2 = 200$.



(c) ROVE with $B_1 = 20$.

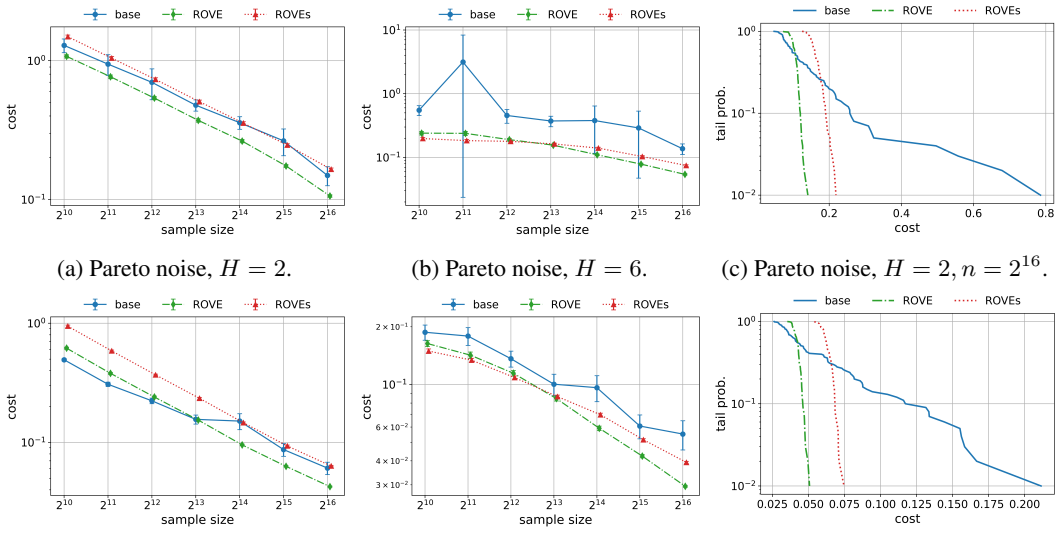
Figure 9: Profiling for ensemble sizes B, B_1, B_2 in resource allocation. Subsample size is chosen as $k = k_1 = k_2 = 10$.



(a) Multiple optima. (b) Multiple near optimum. (c) Single optimum.

Figure 10: Performance of ROVEs in three instances of linear programs under different thresholds ϵ. The setting is identical to that of Figures 6b, 6c, and 7c for ROVE. Hyperparameters: $k_1 = k_2 = \max(10, 0.005n)$, $B_1 = 20$, and $B_2 = 200$. Compared with profiling results for ROVE, we observe that the value of ϵ has similar impacts on the performance of ROVEs. Moreover, the proposed adaptive strategy also behaves well for ROVEs.

- In Figure 21, we demonstrate how the tail heaviness of the problem affects the algorithm performance. The figure shows that the performance gap between ROVE, ROVEs, and the base algorithm becomes increasingly significant as the tail of the uncertainty becomes heavier. This supports the effectiveness of ROVE and ROVEs in handling heavy-tailed uncertainty, where the base algorithm's performance suffers. Note that here MoVE behaves similarly as the base learner due to optima multiplicity.



(a) Pareto noise, $H = 2$. (b) Pareto noise, $H = 6$. (c) Pareto noise, $H = 2, n = 2^{16}$.

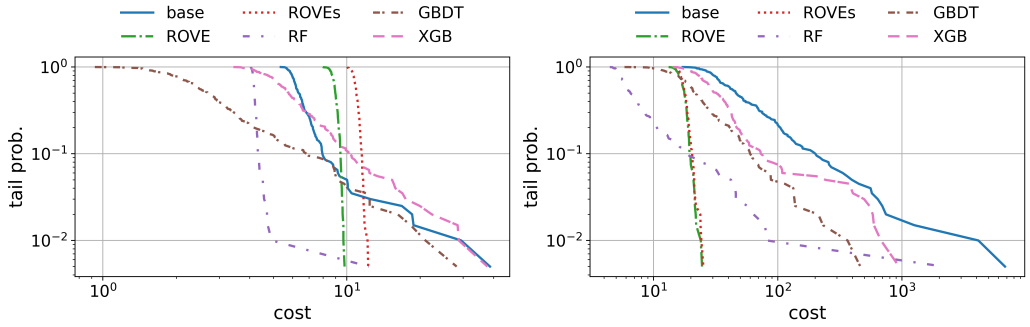
(d) Gaussian noise, $H = 2$. (e) Gaussian noise, $H = 6$. (f) Gaussian noise, $H = 2, n = 2^{16}$.

Figure 11: Results of neural networks with the same setup described in Section 3.1. (a)(b)(d)(e): Expected out-of-sample costs (MSE) with 95% confidence intervals under different noise distributions and varying numbers of hidden layers (H). (c) and (f): Tail probabilities of out-of-sample costs. In (a), ROVEs slightly underperforms the base learner probably due to the weak expressiveness and hence high bias of the MLP with 2 hidden layers.

References

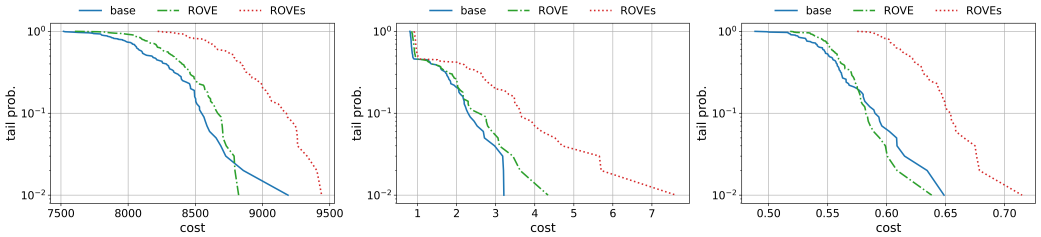
[66] Edward Anderson and Harrison Nguyen. When can we improve on sample average approximation for stochastic optimization? *Operations Research Letters*, 48(5):566–572, 2020.

[67] Irwan Bello, Hieu Pham, Quoc V Le, Mohammad Norouzi, and Samy Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2016.



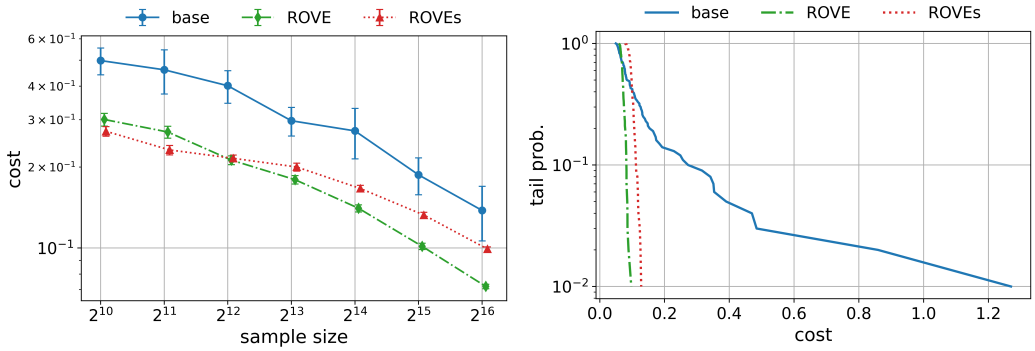
(a) Pareto shape = 2.0, $n = 2^{15}$. (b) Pareto shape = 1.5, $n = 2^{15}$.

Figure 12: Results of decision trees in terms of tail probabilities of out-of-sample costs (MSE). The data generation is $Y = 10\sum_{i=0}^{9} 2^{-i}\sin(2\pi X_i) + \epsilon_1 - \epsilon_2$, where each X_i is independently and uniformly drawn from $[0, 1]$, and ϵ_1, ϵ_2 are independent Pareto variables of the same parameters. Hyperparameters: $k_1 = \max(30, n/10), k_2 = \max(30, n/200), B_1 = 50, B_2 = 200$.



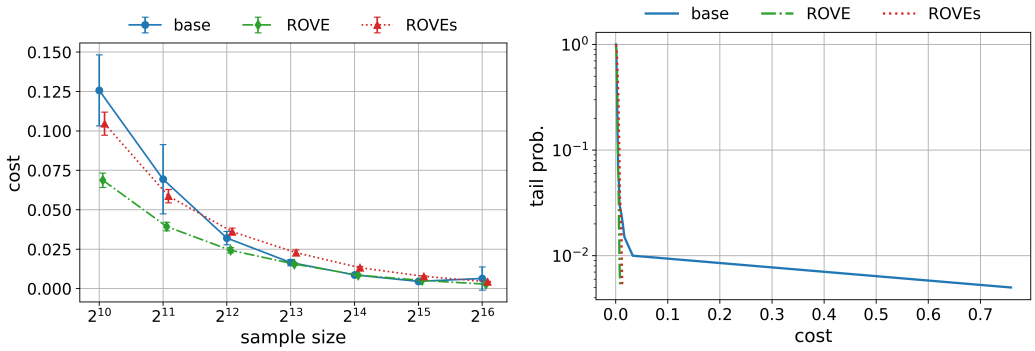
(a) *Appliances Energy*. (b) *Online News*. (c) *Wine Quality*.

Figure 13: Results of neural networks with 4 hidden layers on another three real datasets, in terms of tail probabilities of out-of-sample costs (MSE).



(a) Average performance with 95% CIs. (b) Tail performance for $n = 2^{16}$.

Figure 14: Results with an MLP of $H = 4$ hidden layers. The setup is the same as in Section 3.1 except that the dimension of X is now 30 and the data generation becomes $Y = (1/30) \cdot \sum_{j=1}^{30} \log(X_j + 1) + \varepsilon$, where each X_j is drawn independently from $\mathrm{Unif}(0, 2 + 198(j-1)/29)$.



(a) $d = 10$.

(b) Tail performance. $d = 10, n = 2^{16}$.

(c) $d = 50$.

(d) Tail performance. $d = 50, n = 2^{16}$.

Figure 15: Linear regression with least squares regression as the base learning algorithm. Given the input dimension d, the data generation is $Y = \sum_{i=1}^{d}(-10 + 20(i-1)/(d-1))X_i + \varepsilon_1 - \varepsilon_2$ where each X_i is independent $\mathrm{Unif}(0, 2 + 18(i-1)/(d-1))$ and each $\varepsilon_j, j = 1, 2$ is $\mathrm{Pareto}(2.1)$ and independent of X. (a) and (c): Expected out-of-sample error with 95% confidence interval. (b) and (d): Tail probabilities of out-of-sample errors.

[68] Yoshua Bengio, Andrea Lodi, and Antoine Prouvost. Machine learning for combinatorial optimization: a methodological tour d'horizon. *European Journal of Operational Research*, 290(2):405–421, 2021.

[69] Timo Berthold and Gregor Hendel. Learning to scale mixed-integer programs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):3661–3668, 2021.

[70] Max Biggs, Rim Hariss, and Georgia Perakis. Constrained optimization of objective functions determined from random forests. *Production and Operations Management*, 32(2):397–415, 2023.

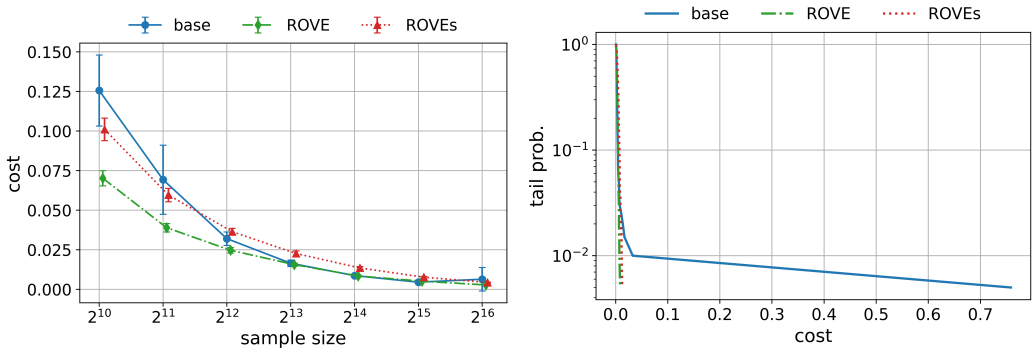
[71] Max Biggs and Georgia Perakis. Tightness of prescriptive tree-based mixed-integer optimization formulations. *arXiv preprint arXiv:2302.14744*, 2023.

[72] John R Birge. Uses of sub-sample estimates to reduce errors in stochastic optimization models. *arXiv preprint arXiv:2310.07052*, 2023.

[73] Luis Candanedo. Appliances Energy Prediction. UCI Machine Learning Repository, 2017. DOI: https://doi.org/10.24432/C5VC8G.

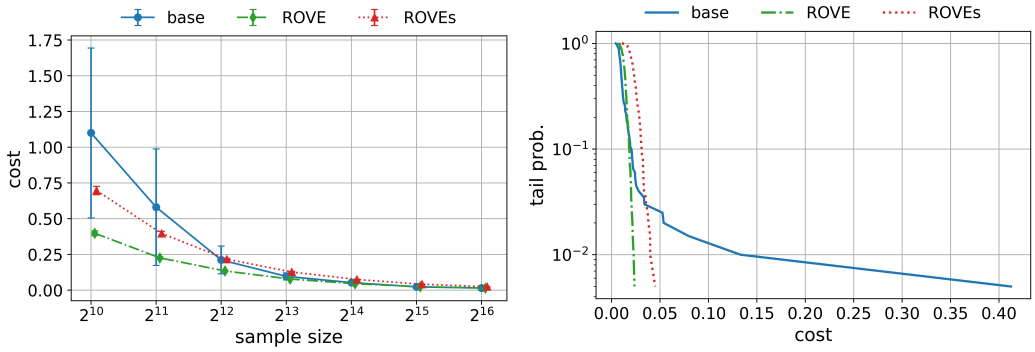
[74] Jessie XT Chen and Miles Lopes. Estimating the error of randomized newton methods: A bootstrap approach. In *International Conference on Machine Learning*, pages 1649–1659. PMLR, 2020.

[75] Tianlong Chen, Xiaohan Chen, Wuyang Chen, Howard Heaton, Jialin Liu, Zhangyang Wang, and Wotao Yin. Learning to optimize: A primer and a benchmark. *Journal of Machine Learning Research*, 23(189):1–59, 2022.



(a) $d = 10$.

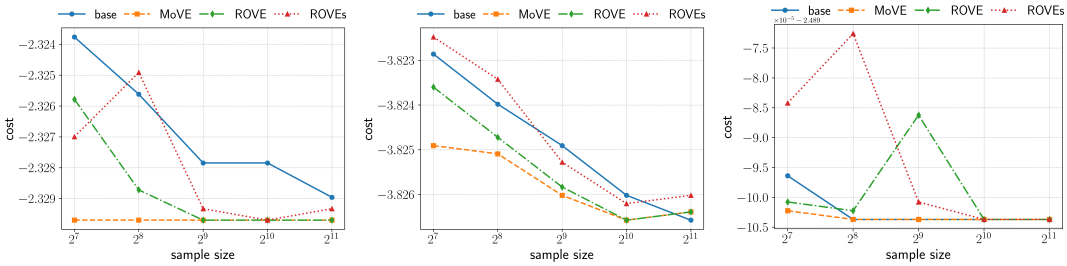
(b) Tail performance. $d = 10, n = 2^{16}$.



(c) $d = 50$.

(d) Tail performance. $d = 50, n = 2^{16}$.

Figure 16: Linear regression with Ridge regression as the base learning algorithm. The same data generation as in Figure 15. (a) and (c): Expected out-of-sample error with 95% confidence interval. (b) and (d): Tail probabilities of out-of-sample errors.

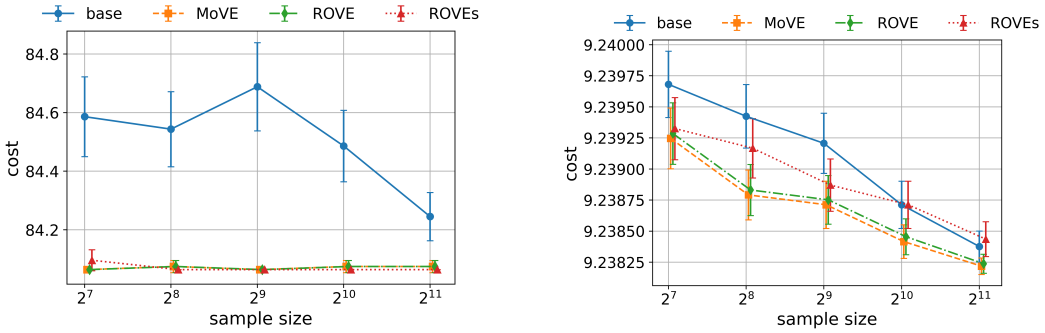


(a) Instance 1.

(b) Instance 2.

(c) Instance 3.

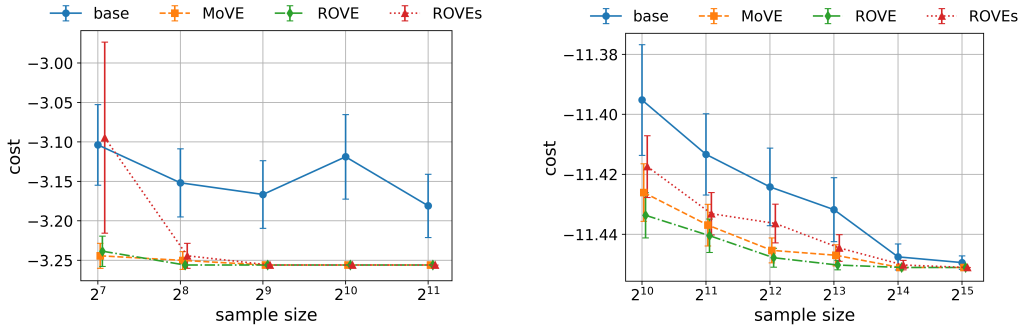
Figure 17: Results for linear programs with light-tailed objectives. The base algorithm is SAA.



(a) $k = k_1 = k_2 = \max(10, n/10)$.

(b) A different instance with strong correlation.

Figure 18: Results for supply chain network design. (a): The same problem instance as in Section 3.2 under a different hyperparameter choice: $k = \max(10, n/10)$, $B = 200$ for MoVE and $k_1 = k_2 = \max(10, n/10)$, $B_1 = 20$, $B_2 = 200$ for ROVE and ROVEs. (b): The same setup as in Section 3.2 but on a different problem instance for which the objectives under different solutions are strongly correlated. The strong correlation cancels out most of the heavy-tailed noise between solutions, making the base algorithm less susceptible to these noises, thus our ensemble methods appear less effective.



(a) Resource allocation.

(b) Maximum weight matching.

Figure 19: Results for resource allocation and maximum weight matching when the base algorithm is DRO using 1-Wasserstein metric with the l_∞ norm.

[76] Xiaohan Chen, Jialin Liu, and Wotao Yin. Learning to optimize: A tutorial for continuous and mixed-integer optimization. *Science China Mathematics*, pages 1–72, 2024.

[77] Xiaotie Chen and David L Woodruff. Software for data-based stochastic programming using bootstrap estimation. *INFORMS Journal on Computing*, 35(6):1218–1224, 2023.

[78] Xiaotie Chen and David L Woodruff. Distributions and bootstrap for data-based stochastic programming. *Computational Management Science*, 21(1):33, 2024.

[79] Paulo Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Wine Quality. UCI Machine Learning Repository, 2009. DOI: https://doi.org/10.24432/C56S3T.

[80] Arnaud Deza and Elias B Khalil. Machine learning for cutting planes in integer programming: a survey. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 6592–6600, 2023.

[81] Andreas Eichhorn and Werner Römisch. Stochastic integer programming: Limit theorems and confidence intervals. *Mathematics of Operations Research*, 32(1):118–135, 2007.

[82] Yixin Fang, Jinfeng Xu, and Lei Yang. Online bootstrap confidence intervals for the stochastic gradient descent estimator. *Journal of Machine Learning Research*, 19(78):1–21, 2018.

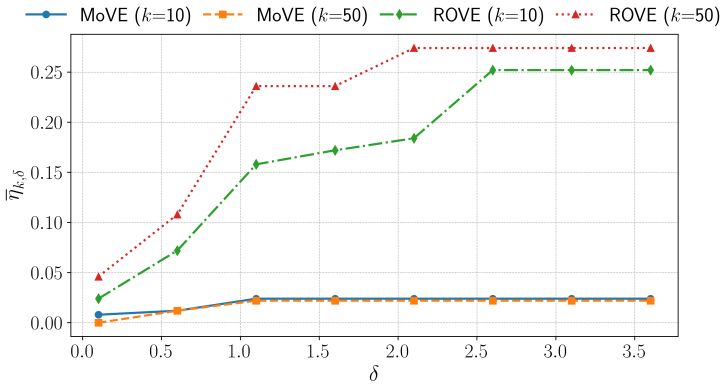


Figure 20: Comparison of $\bar{\eta}_{k,\delta}$ for MoVE and ROVE in a linear program with multiple optima (corresponds to the instance in Figure 5d). Threshold ϵ is chosen as $\epsilon = 4$ when $k = k_1 = k_2 = 10$ and $\epsilon = 2.5$ when $k = k_1 = k_2 = 50$, according to the adaptive strategy. Note that $\bar{\eta}_{k,\delta} = \max_{\theta\in\Theta} p_k(\theta) - \max_{\theta\in\Theta\setminus\Theta^\delta} p_k(\theta)$ by (27), which measures the generalization sensitivity. For MoVE, we have $p_k(\theta) = \mathbb{P}(\hat{\theta}_k^{SAA} = \theta)$; and for ROVE, we have $p_k(\theta) = \mathbb{P}(\theta \in \widehat{\Theta}_k^\epsilon)$, where $\widehat{\Theta}_k^\epsilon$ is the ϵ-optimal set of SAA defined in (23). From the figure, we can observe that the issue brought by the presence of multiple optimal solutions can be alleviated using the two-phase strategy in ROVE.

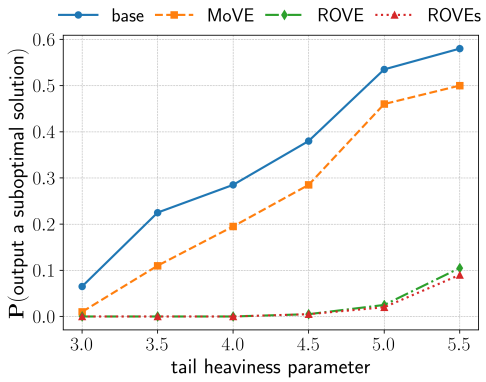


Figure 21: Influence of tail heaviness in the stochastic linear program with multiple optima with $n = 10^6$. Hyperparameters: $k = 50, B = 2000$ for MoVE, $k_1 = k_2 = 50, B_1 = 200, B_2 = 5000$ for ROVE and ROVEs. The tail heaviness parameter corresponds to the mean of the Pareto random coefficient.

[83] Kelwin Fernandes, Pedro Vinagre, Paulo Cortez, and Pedro Sernadela. Online News Popularity. UCI Machine Learning Repository, 2015. DOI: https://doi.org/10.24432/C5NS3V.

[84] He He, Hal Daume III, and Jason M Eisner. Learning to search in branch and bound algorithms. *Advances in neural information processing systems*, 27, 2014.

[85] Elias Khalil, Pierre Le Bodic, Le Song, George Nemhauser, and Bistra Dilkina. Learning to branch in mixed integer programming. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), 2016.

[86] Anton J Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on optimization*, 12(2):479–502, 2002.

[87] Henry Lam and Huajie Qian. Assessing solution quality in stochastic optimization via bootstrap aggregating. In *Proceedings of the 2018 Winter Simulation Conference*, pages 2061–2071. IEEE, 2018.

[88] Henry Lam and Huajie Qian. Bounding optimality gap in stochastic optimization via bagging: Statistical efficiency and stability. *arXiv preprint arXiv:1810.02905*, 2018.

[89] Miles Lopes, Shusen Wang, and Michael Mahoney. Error estimation for randomized least-squares algorithms via the bootstrap. In *International Conference on Machine Learning*, pages 3217–3226. PMLR, 2018.

[90] Georgia Perakis and Leann Thayaparan. Umotem: Upper bounding method for optimizing over tree ensemble models. *Available at SSRN 3972341*, 2021.

[91] Lara Scavuzzo, Feng Chen, Didier Chételat, Maxime Gasse, Andrea Lodi, Neil Yorke-Smith, and Karen Aardal. Learning to branch with tree mdps. *Advances in Neural Information Processing Systems*, 35:18514–18526, 2022.

[92] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

[93] Yunzhuang Shen, Yuan Sun, Andrew Eberhard, and Xiaodong Li. Learning primal heuristics for mixed integer programs. In *2021 international joint conference on neural networks (ijcnn)*, pages 1–8. IEEE, 2021.

[94] Yunhao Tang, Shipra Agrawal, and Yuri Faenza. Reinforcement learning for integer programming: Learning to cut. In *International conference on machine learning*, pages 9367–9376. PMLR, 2020.

[95] Keliang Wang, Leonardo Lozano, Carlos Cardonha, and David Bergman. Optimizing over an ensemble of neural networks. *arXiv preprint arXiv:2112.07007*, 2021.

[96] Jiayi Zhang, Chang Liu, Xijun Li, Hui-Ling Zhen, Mingxuan Yuan, Yawen Li, and Junchi Yan. A survey for solving mixed integer programming via machine learning. *Neurocomputing*, 519:205–217, 2023.

[97] Yanjie Zhong, Todd Kuffner, and Soumendra Lahiri. Online bootstrap inference with nonconvex stochastic gradient descent estimator. *arXiv preprint arXiv:2306.02205*, 2023.