From Stacked Predictions to Decisions: A Contextual Optimization Approach

Yanru Guo[1] **Ruiwei Jiang**[1] **Siqian Shen**[1]
[1] Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, MI 48109
{yanruguo,ruiwei,siqian}@umich.edu

Abstract

Optimization models and decision frameworks have seen increased use of the combination of multiple first-level learners using ensemble methods to improve predictive performance. In this work, we cast stacking as a contextual optimization problem, where a collection of T first-level learners $\{h_t\}_{t=1}^T$ serve as context, a meta-learner estimates the target Y conditional on this context, and decisions are selected to minimize the resulting conditional expected cost. We propose an empirical residual-based sample average approximation (ER-SAA) pipeline that incorporates the predictive uncertainty into the decision stage. Specifically, we (i) obtain out-of-sample first-level learners through cross-fitting, (ii) fit a meta-learner and estimate a possibly heteroskedastic scale map, and (iii) add back cross-fitted residuals as decision scenarios. This construction preserves stacking's modeling flexibility and at the same time aligns learning with the conditional decision objective.

1 Introduction

Decision-making models and optimization approaches under uncertainty increasingly rely on multiple first-level learners trained for the same predictive task but exhibiting different inductive biases and error patterns. A widely used ensemble method to integrate these learners is stacking [Zhou, 2025], where a meta-learner is trained on their outputs to enhance predictive performance [Wolpert, 1992, Breiman, 1996, Smyth and Wolpert, 1997]. Yet, decision quality depends not only on point prediction accuracy but also on how predictive uncertainty interacts with costs and operational constraints. Relying solely on plug-in point estimates can obscure these interactions and ultimately lead to suboptimal actions.

In this work, we propose to view stacking through the lens of contextual optimization. First-level predictions form the context that informs a conditional decision problem. Instead of optimizing at a single meta-prediction, we incorporate uncertainty from the stacked model to the downstream optimizer. Specifically, we adopt the empirical residuals–based sample average approximation (ER–SAA) method [Kannan et al., 2025], where cross-fitted residuals from the meta-learner are added back as scenarios at decision time, optionally scaled by a learned map to reflect heteroskedasticity. This preserves the modeling flexibility of stacking while exposing the optimization model to realistic, context-dependent variation around the point prediction.

The classical stochastic optimization assumes a known distribution for the uncertainty and optimizes the expected performance [Birge and Louveaux, 2011]; robust optimization protects against worst cases within a specified uncertainty set [Ben-Tal and Nemirovski, 2002, Bertsimas et al., 2011]; distributionally robust optimization (DRO) optimizes over an ambiguity set of plausible distributions and, with, e.g., Wasserstein balls that often yield tractability and finite-sample guarantees [Kuhn et al.,

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: ML×OR: Mathematical Foundations and Operational Integration of Machine Learning for Uncertainty-Aware Decision-Making.

2024, Esfahani and Kuhn, 2018, Xie, 2021, Shen and Jiang, 2023, Chen et al., 2024]. The distribution of uncertain parameters is typically treated as a single upstream object across these methods, and they do not model or exploit how that distribution is constructed by combining multiple sources (i.e., first-level learners). Recent works have begun to incorporate such combination structure into optimization. For example, multi-source DRO calibrates ambiguity using per-source information (e.g., intersections of source-wise Wasserstein balls), and parametric fusion aggregates source moments into one distribution [e.g., Rychener et al., 2024, Guo et al., 2024]. These methods bring source-awareness into the uncertainty model, but they largely focus on how to robustify across sources rather than on how the meta-learner's learned residual uncertainty should drive downstream decisions.

We take a complementary perspective to the aforementioned methods. In particular, we consider the first-level predictions as the decision context and use ER–SAA to convert residuals into decision scenarios. This preserves stacking's modeling flexibility and exposes the optimizer to decision-relevant uncertainty (including heteroskedasticity). It can also be layered with DRO (e.g., ambiguity sets around the residual distribution) when additional robustness is desired [Kannan et al., 2024].

Notation. We denote scalars in lightface, vectors in bold lowercase (e.g., $\boldsymbol{u} \in \mathbb{R}^n$), and matrices in bold uppercase (e.g., $\boldsymbol{A} \in \mathbb{R}^{m\times n}$). For $N \in \mathbb{N}$, we have $[N] := \{1, \ldots, N\}$. Let $\boldsymbol{I}_d$ denote the $d \times d$ identity, $\mathbf{0}$ a zero vector (dimension clear from context). Random variables are uppercase (e.g., X, Y), and their realizations are lowercase (e.g., $\boldsymbol{x}, \boldsymbol{y}$). For a set $\mathcal{Y}$, $\mathrm{proj}_{\mathcal{Y}}(\cdot)$ denotes Euclidean projection. Truth is marked by * (e.g., f^*, Q^*) and estimators carry hats (e.g., $\hat{f}_n, \hat{Q}_n$).

2 Problem Description

We study a contextual optimization problem in which a decision is chosen after observing covariates. Given $X \in \mathcal{X} \subseteq \mathbb{R}^{d_x}$ observed prior to the decision, our goal is to approximate the solution to

$$\min_{\boldsymbol{z}\in\mathcal{Z}} \mathbb{E}\left[c(\boldsymbol{z}, Y) \mid X = \boldsymbol{x}\right], \tag{1}$$

where $\boldsymbol{z} \in \mathcal{Z} \subseteq \mathbb{R}^{d_z}$ is the decision, $Y \in \mathcal{Y} \subseteq \mathbb{R}^{d_y}$ collects the uncertain problem parameters, and $c(\boldsymbol{z}, Y)$ is a cost function. Ideally, for a newly observed context $X = \boldsymbol{x}$, we would minimize (1) under the ground-truth conditional distribution $\mu_{Y|\boldsymbol{x}}$. However, $\mu_{Y|\boldsymbol{x}}$ is often unknown in practice, and instead we observe a training sample $\mathcal{D}_n = \{(\boldsymbol{x}^i, y^i)\}_{i=1}^n$ of past contexts and outcomes.

Stacking representation. We model $Y \mid X$ via stacking and treat stacked predictions as the context that drives the conditional decision. Let $h_t : \mathcal{X} \to \mathbb{R}^{d_y}$ for $t \in [T]$ be first-level learners that produce out-of-sample predictions from X. We use K-fold cross-fitting so that $h_t(\boldsymbol{x}^i)$ is always computed from a model that did not train on $(\boldsymbol{x}^i, y^i)$. We collect these into the stacked feature:

$$\boldsymbol{\zeta} := H(X) := \left[h_1(X)^\mathsf{T}, \ldots, h_T(X)^\mathsf{T}\right]^\mathsf{T} \in \mathbb{R}^{Td_y}. \tag{2}$$

A meta-regression $F^* : \mathbb{R}^{Td_y} \to \mathbb{R}^{d_y}$ captures the conditional mean, and a covariance-root map $Q^* : \mathbb{R}^{Td_y} \to \mathbb{R}^{d_y\times d_y}$ captures (possibly heteroskedastic) dispersion. We posit

$$Y = F^*(\boldsymbol{\zeta}) + Q^*(\boldsymbol{\zeta})\,\boldsymbol{\epsilon}, \quad \mathbb{E}[\boldsymbol{\epsilon}] = \mathbf{0}, \quad \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\mathsf{T}] = \boldsymbol{I}_{d_y}. \tag{3}$$

with $\boldsymbol{\epsilon}$ being independent of X. Thus, $F^*(\boldsymbol{\zeta}) = \mathbb{E}[Y \mid \boldsymbol{\zeta}]$ and $Q^*(\boldsymbol{\zeta})Q^*(\boldsymbol{\zeta})^\mathsf{T} = \mathrm{Cov}(Y \mid \boldsymbol{\zeta})$.

Decision risk at fixed meta-features. For any realization $\boldsymbol{\zeta}$ as in (2), we define the induced decision risk under (3) as

$$g(\boldsymbol{z}; \boldsymbol{\zeta}) := \mathbb{E}\left[c\left(\boldsymbol{z}, F^*(\boldsymbol{\zeta}) + Q^*(\boldsymbol{\zeta})\boldsymbol{\epsilon}\right)\right], \tag{4}$$

where the expectation is taken with respect to $\boldsymbol{\epsilon}$. The contextual objective in (1) can then be written as the plug-in problem $\min_{\boldsymbol{z}\in\mathcal{Z}} g(\boldsymbol{z}; H(\boldsymbol{x}))$.

Full-information SAA (FI-SAA). Suppose, hypothetically, that F^* and Q^* are known. For each training point $i \in [n]$, we form the meta-features $\boldsymbol{\zeta}^i := H(\boldsymbol{x}^i)$ and the pseudo-error

$$\boldsymbol{\epsilon}^{i,*} := \left[Q^*(\boldsymbol{\zeta}^i)\right]^{-1}\left(y^i - F^*(\boldsymbol{\zeta}^i)\right), \quad i \in [n], \tag{5}$$

At any target $\boldsymbol{\zeta}$, the FI-SAA objective is

$$g_n^*(\boldsymbol{z};\boldsymbol{\zeta}) \;:=\; \frac{1}{n}\sum_{i=1}^{n} c\big(\boldsymbol{z},\, F^*(\boldsymbol{\zeta}) + Q^*(\boldsymbol{\zeta})\,\boldsymbol{\epsilon}^{\,i,*}\big)\,. \tag{6}$$

where $\boldsymbol{\epsilon}^{i,*}$ is defined in (5). Under standard SAA conditions, $g_n^*(\cdot;\boldsymbol{\zeta})$ converges uniformly to $g(\cdot;\boldsymbol{\zeta})$ on $\mathcal{Z}$.

Estimation and residuals for ER-SAA. In practice, F^* and Q^* are typically unknown. We obtain cross-fitted first-level predictions $\widehat{\boldsymbol{\zeta}}^i$, fit a meta-learner $\hat{F}_n$ on $(\widehat{\boldsymbol{\zeta}}^i, y^i)$, estimate a covariance-root $\hat{Q}_n$ on the residuals $y^i - \hat{F}_n(\widehat{\boldsymbol{\zeta}}^i)$, and compute cross-fitted empirical residuals

$$\hat{\boldsymbol{\epsilon}}^i := \Big[\hat{Q}_n(\widehat{\boldsymbol{\zeta}}^i)\Big]^{-1}\Big(y^i - \hat{F}_n(\widehat{\boldsymbol{\zeta}}^i)\Big)\,. \tag{7}$$

Given a new context $\boldsymbol{x}$ with $\widehat{\boldsymbol{\zeta}} = \widehat{H}(\boldsymbol{x})$, ER-SAA constructs conditional scenarios $\hat{F}_n(\widehat{\boldsymbol{\zeta}}) + \hat{Q}_n(\widehat{\boldsymbol{\zeta}})\hat{\boldsymbol{\epsilon}}^i$ and solves the resulting SAA approximation to (1). The complete ER-SAA program is presented in Section 3.

Assumptions. We summarize mild regularity conditions used for analysis. The convergence for ER-SAA is presented in Appendix A.1.

- **A1 Problem regularity.** $\mathcal{X}$ is compact; $\mathcal{Y}$ is nonempty, closed, and convex; $\mathcal{Z}$ is nonempty and compact. For almost every $\boldsymbol{\zeta}$, $g(\boldsymbol{z};\boldsymbol{\zeta})$ in (4) is finite and lower semi-continuous in $\boldsymbol{z}$. There exists $L : \mathcal{Z} \to \mathbb{R}_+$ with $\sup_{\boldsymbol{z}\in\mathcal{Z}} L(\boldsymbol{z}) < \infty$ such that $|c(\boldsymbol{z},\boldsymbol{y}') - c(\boldsymbol{z},\boldsymbol{y})| \le L(\boldsymbol{z})\|\boldsymbol{y}' - \boldsymbol{y}\|$ for all $\boldsymbol{z} \in \mathcal{Z}$, $\boldsymbol{y}, \boldsymbol{y}' \in \mathcal{Y}$. Under these conditions, the FI-SAA averages $g_n^*(\cdot;\boldsymbol{\zeta})$ converge to $g(\cdot;\boldsymbol{\zeta})$ uniformly on $\mathcal{Z}$.
- **A2 Meta well-specification/approximation.** Either (a) $F^* \in \mathcal{F}$, or (b) F^* admits a best $L^2(P_{\boldsymbol{\zeta}})$ projection onto $\mathcal{F}$. Approximation error is tracked explicitly.
- **A3 Noise and scale.** $\mathbb{E}[\boldsymbol{\epsilon} \mid X] = \mathbf{0}$ and $\mathrm{Cov}(\boldsymbol{\epsilon} \mid X) = \boldsymbol{I}_{d_y}$ (hence also conditional on $\boldsymbol{\zeta}$). In heteroscedastic settings, $Q^*(\boldsymbol{\zeta}) \succ 0$ for all $\boldsymbol{\zeta}$ with $\sup_{\boldsymbol{\zeta}} \|Q^*(\boldsymbol{\zeta})\| < \infty$ and $\sup_{\boldsymbol{\zeta}} \|[Q^*(\boldsymbol{\zeta})]^{-1}\| < \infty$.
- **A4 Identifiability.** For a chosen meta feature map $\phi : \mathbb{R}^{Td_y} \to \mathbb{R}^p$ (e.g., $\phi(\boldsymbol{\zeta}) = \boldsymbol{\zeta}$ for linear stacking), the Gram matrix $\mathbb{E}[\phi(\boldsymbol{\zeta})\phi(\boldsymbol{\zeta})^\mathsf{T}]$ is positive definite. Optional constraints such as non-negativity or simplex weights can be imposed for interpretability.
- **A5 Estimation consistency.** With K-fold cross-fitting for first-level and meta-learners,

$$\sup_{\boldsymbol{x}\in\mathcal{X}} \|\hat{h}_{t,n}(\boldsymbol{x}) - h_t(\boldsymbol{x})\| \xrightarrow{p} 0 \;\; (\forall t \in [T]), \quad \sup_{\boldsymbol{\zeta}} \|\hat{F}_n(\boldsymbol{\zeta}) - F^*(\boldsymbol{\zeta})\| \xrightarrow{p} 0,$$
$$\hat{Q}_n(\boldsymbol{\zeta}) \succ 0, \quad \sup_{\boldsymbol{\zeta}} \|\hat{Q}_n(\boldsymbol{\zeta}) - Q^*(\boldsymbol{\zeta})\| \xrightarrow{p} 0, \quad \sup_{\boldsymbol{\zeta}} \|[\hat{Q}_n(\boldsymbol{\zeta})]^{-1} - [Q^*(\boldsymbol{\zeta})]^{-1}\| \xrightarrow{p} 0.$$

- **A6 Residual moments/tails.** $\frac{1}{n}\sum_{i=1}^n \|\boldsymbol{\epsilon}^i\|^2 \xrightarrow{p} \mathbb{E}\|\boldsymbol{\epsilon}\|^2$, with optional sub-Gaussian (or light-tail) conditions for sharper concentration bounds.

3 ER-SAA for Stacked Predictions

We conduct the residual-incorporation approach introduced in Section 2. We first obtain out-of-sample first-level predictions via cross-fitting and fit the meta-learner. Then, we extract cross-fitted residuals and estimate a heteroskedasticity scale map. Finally, we add back residuals to the target meta-features to create decision scenarios.

Cross-fitted first-level and meta training. Using the K-fold procedure in Algorithm 1, for each fold j we fit first-level learners on $D_{(-j)}$ and compute out-of-sample first-level predictions for $(\boldsymbol{x}^i, y^i) \in D_j$:

$$\boldsymbol{\zeta}^i = H_{(-j)}(\boldsymbol{x}^i) := \Big[\hat{h}_{1,(-j)}(\boldsymbol{x}^i)^\mathsf{T}, \ldots, \hat{h}_{T,(-j)}(\boldsymbol{x}^i)^\mathsf{T}\Big]^\mathsf{T}.$$

We collect $D' = \{(\boldsymbol{\zeta}^i, y^i)\}_{i=1}^n$, fit the meta-learner $\hat{F}_n \in \mathcal{F}$ on D' and estimate a covariance-root map $\hat{Q}_n$ from the resulting residuals.

Algorithm 1: Stacking with Cross-Fitted First-level Learners

Input: Data $\mathcal{D}_n = \{(\boldsymbol{x}^i, y^i)\}$; first-level algorithms $\{\mathcal{L}_t\}_{t=1}^T$; meta algorithm $\mathcal{L}_{\text{meta}}$.

Split into K folds $\{D_j\}_{j=1}^K$.

for $j = 1$ **to** K **do**

  Train $\hat{h}_{t,(-j)} = \mathcal{L}_t(D_{(-j)})$ for $t \in [T]$.

  for $(\boldsymbol{x}^i, y^i) \in D_j$ **do**

    $\boldsymbol{\zeta}^i \leftarrow \left[\hat{h}_{1,(-j)}(\boldsymbol{x}^i)^\intercal, \ldots, \hat{h}_{T,(-j)}(\boldsymbol{x}^i)^\intercal\right]^\intercal$.

Fit $\hat{F}_n \leftarrow \mathcal{L}_{\text{meta}}\big(\{(\boldsymbol{\zeta}^i, y^i)\}_{i=1}^n\big)$.

Refit $\hat{h}_t \leftarrow \mathcal{L}_t(\mathcal{D}_n)$ for $t \in [T]$.

Define $\widehat{H}(\boldsymbol{x}) := \left[\hat{h}_1(\boldsymbol{x})^\intercal, \ldots, \hat{h}_T(\boldsymbol{x})^\intercal\right]^\intercal$.

Output: $\hat{F}_n(\widehat{H}(\boldsymbol{x}))$.

Cross-fitted residuals. For each $i \in [n]$, we compute

$$\hat{\boldsymbol{\epsilon}}^i := \left[\hat{Q}_n(\boldsymbol{\zeta}^i)\right]^{-1} \left(y^i - \hat{F}_n(\boldsymbol{\zeta}^i)\right). \tag{8}$$

Cross-fitting ensures that $\boldsymbol{\zeta}^i$ and hence the residual $y^i - \hat{F}_n(\boldsymbol{\zeta}^i)$ are evaluated out-of-sample with respect to the base learners, mitigating overfitting in subsequent scenario construction.

Residual add back and scenario generation. Given a new context $\boldsymbol{x}$, we form target meta-features $\hat{\boldsymbol{\zeta}} = \widehat{H}(\boldsymbol{x})$ and generate decision-relevant scenarios by adding back residuals to $\hat{\boldsymbol{\zeta}}$:

$$\tilde{\boldsymbol{y}}^i(\boldsymbol{x}) := \hat{F}_n(\hat{\boldsymbol{\zeta}}) + \hat{Q}_n(\hat{\boldsymbol{\zeta}})\hat{\boldsymbol{\epsilon}}^i, \quad i \in [n]. \tag{9}$$

Intuitively, $\hat{\boldsymbol{\epsilon}}^i$ captures shape information of $Y \mid \boldsymbol{\zeta}$, while $\hat{Q}_n(\hat{\boldsymbol{\zeta}})$ adapts its scale to the deployment context.

Empirical residual-based SAA. We define the objective at meta-features $\boldsymbol{\zeta}$ by

$$\hat{g}_n^{\text{ER}}(\boldsymbol{z}; \boldsymbol{\zeta}) := \frac{1}{n} \sum_{i=1}^n c\left(\boldsymbol{z}, \text{proj}_{\mathcal{Y}}\left(\tilde{\boldsymbol{y}}^i(\boldsymbol{\zeta})\right)\right). \tag{10}$$

The projection $\text{proj}_{\mathcal{Y}}$ is optional and only used if some $\tilde{\boldsymbol{y}}^i(\boldsymbol{x})$ fall outside $\mathcal{Y}$ due to model misspecification. Given $\boldsymbol{x}$ with $\widehat{\boldsymbol{\zeta}} = \widehat{H}(\boldsymbol{x})$, the ER-SAA value is

$$\hat{v}_n^{\text{ER}}(\boldsymbol{x}) := \min_{\boldsymbol{z} \in \mathcal{Z}} \hat{g}_n^{\text{ER}}(\boldsymbol{z}; \widehat{\boldsymbol{\zeta}}). \tag{11}$$

4 Conclusion and Discussion

We studied stacking through the lens of contextual stochastic programs. Rather than optimizing at a single meta-prediction, we use ER-SAA to integrate the ensemble's residual uncertainty to the downstream decision making. This preserves stacking's modeling flexibility while aligning learning with the conditional decision objective.

In the future work, for computational experiments, we will compare stacking in ER-SAA with a plug-in stacking (optimize at the meta-prediction) and a best single first-level learner. We will evaluate each method's decision cost relative to an FI-SAA oracle and compare their runtime. In addition, we will further investigate placing an ambiguity set around the empirical residual distribution to stabilize ER-SAA [Kannan et al., 2024].

Overall, stacking in ER-SAA offers a simple, solver-agnostic recipe to convert stacked predictions into decision-relevant scenarios, aiming to improve operational performance precisely.

References

Aharon Ben-Tal and Arkadi Nemirovski. Robust optimization–methodology and applications. *Mathematical Programming*, 92:453–480, 2002. doi: 10.1007/s101070100286. URL https://doi.org/10.1007/s101070100286.

Dimitris Bertsimas, David B. Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011. doi: 10.1137/080734510. URL https://doi.org/10.1137/080734510.

John R. Birge and François Louveaux. *Introduction to Stochastic Programming*. Springer Science & Business Media, New York, 2011.

Leo Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, 1996. URL https://doi.org/10.1007/BF00117832.

Zhi Chen, Daniel Kuhn, and Wolfram Wiesemann. Data-driven chance constrained programs over Wasserstein balls. *Operations Research*, 72(1):410–424, 2024. URL https://doi.org/10.1287/opre.2022.2330.

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. doi: 10.1007/s10107-017-1172-1. URL https://doi.org/10.1007/s10107-017-1172-1.

Yanru Guo, Bo Zhou, Ruiwei Jiang, Xi Yang, and Siqian Shen. Distributionally robust resource allocation with trust-aided parametric information fusion. In *Proceedings of the 63rd IEEE Conference on Decision and Control (CDC 2024)*, 2024. URL https://doi.org/10.1109/CDC56724.2024.10886572.

Rohit Kannan, Güzin Bayraksan, and James R Luedtke. Residuals-based distributionally robust optimization with covariate information. *Mathematical Programming*, 207(1):369–425, 2024. URL https://doi.org/10.1007/s10107-023-02014-7.

Rohit Kannan, Güzin Bayraksan, and James R Luedtke. Data-driven sample average approximation with covariate information. *Operations Research*, 2025. URL https://doi.org/10.1287/opre.2020.0533.

Daniel Kuhn, Soroosh Shafiee, and Wolfram Wiesemann. Distributionally robust optimization, 2024. URL https://arxiv.org/abs/2411.02549.

Yves Rychener, Adrian Esteban-Perez, Juan M. Morales, and Daniel Kuhn. Wasserstein distributionally robust optimization with heterogeneous data sources, 2024. URL https://arxiv.org/abs/2407.13582.

Haoming Shen and Ruiwei Jiang. Chance-constrained set covering with Wasserstein ambiguity. *Mathematical Programming*, 198(1):621–674, 2023. URL https://doi.org/10.1007/s10107-022-01788-6.

Padhraic Smyth and David Wolpert. Stacked density estimation. In *Advances in Neural Information Processing Systems*, 1997. URL https://proceedings.neurips.cc/paper_files/paper/1997/file/ee8374ec4e4ad797d42350c904d73077-Paper.pdf.

David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992. URL https://doi.org/10.1016/S0893-6080(05)80023-1.

Weijun Xie. On distributionally robust chance constrained programs with Wasserstein distance. *Mathematical Programming*, 186(1):115–155, 2021. URL https://doi.org/10.1007/s10107-019-01445-5.

Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. CRC press, 2025.

A Technical Appendices and Supplementary Material

A.1 Consistency of ER-SAA

For a fixed meta-feature vector $\boldsymbol{\zeta}$, we define

$$v^*(\boldsymbol{\zeta}) := \inf_{\boldsymbol{z}\in\mathcal{Z}} g(\boldsymbol{z};\boldsymbol{\zeta}), \quad S^*(\boldsymbol{\zeta}) := \arg\min_{\boldsymbol{z}\in\mathcal{Z}} g(\boldsymbol{z};\boldsymbol{\zeta}),$$

and let $\hat{v}_n^{\mathrm{ER}}(\boldsymbol{\zeta})$ and $\hat{S}_n^{\mathrm{ER}}(\boldsymbol{\zeta})$ denote the ER-SAA value and solution set. We use the deviation distance $D(A,B) := \sup_{\boldsymbol{a}\in A}\inf_{\boldsymbol{b}\in B}\|\boldsymbol{a}-\boldsymbol{b}\|$ for nonempty compact sets $A, B \subseteq \mathbb{R}^{d_z}$.

Proposition 1 (Uniform convergence of the ER-SAA objective) *Suppose **A1** and **A3–A5** hold. Then, for $P_{\boldsymbol{\zeta}}$-almost every $\boldsymbol{\zeta}$,*

$$\sup_{\boldsymbol{z}\in\mathcal{Z}} \left|\hat{g}_n^{\mathrm{ER}}(\boldsymbol{z};\boldsymbol{\zeta}) - g(\boldsymbol{z};\boldsymbol{\zeta})\right| \xrightarrow{p} 0. \tag{12}$$

Let g_n^* be the full-information SAA in (6). By the decomposition $\sup_{\boldsymbol{z}}|\hat{g}_n^{\mathrm{ER}} - g| \le \sup_{\boldsymbol{z}}|\hat{g}_n^{\mathrm{ER}} - g_n^*| + \sup_{\boldsymbol{z}}|g_n^* - g|$, the second term vanishes by standard SAA uniform convergence under **A1**. The first term is bounded via the Lipschitz in $\boldsymbol{y}$ property in **A1** and the mean deviation between adding back residual scenarios and the FI scenarios. This deviation goes to zero by consistency of $(\hat{F}_n, \hat{Q}_n)$ and stability of residuals under **A3**–**A5**.

Theorem 1 (Value and solution consistency at fixed $\boldsymbol{\zeta}$) *Under the assumptions of Proposition 1,*

$$\hat{v}_n^{\mathrm{ER}}(\boldsymbol{\zeta}) \xrightarrow{p} v^*(\boldsymbol{\zeta}), \quad D\left(\hat{S}_n^{\mathrm{ER}}(\boldsymbol{\zeta}), S^*(\boldsymbol{\zeta})\right) \xrightarrow{p} 0, \quad \sup_{\boldsymbol{z}\in\hat{S}_n^{\mathrm{ER}}(\boldsymbol{\zeta})} g(\boldsymbol{z};\boldsymbol{\zeta}) \xrightarrow{p} v^*(\boldsymbol{\zeta}). \tag{13}$$

We combine Proposition 1 with standard argmin continuity on compact $\mathcal{Z}$ (the proof follows Theorem 1 in Kannan et al. [2025]). In homoscedastic settings ($Q^* \equiv I$), heteroscedasticity-specific conditions are unnecessary.

Corollary 1 (Deployed consistency with $\widehat{\boldsymbol{\zeta}} = \widehat{H}(\boldsymbol{x})$) *If, in addition, $\sup_{\boldsymbol{x}\in\mathcal{X}}\|\widehat{H}(\boldsymbol{x}) - H(\boldsymbol{x})\| \xrightarrow{p} 0$ (implied by **A5**), then for P_X-almost every $\boldsymbol{x}$,*

$$\sup_{\boldsymbol{z}\in\mathcal{Z}} \left|\hat{g}_n^{\mathrm{ER}}(\boldsymbol{z};\widehat{H}(\boldsymbol{x})) - g(\boldsymbol{z};H(\boldsymbol{x}))\right| \xrightarrow{p} 0,$$

and consequently $\hat{v}_n^{\mathrm{ER}}(\boldsymbol{x}) \xrightarrow{p} v^*(\boldsymbol{x})$ and $D(\hat{S}_n^{\mathrm{ER}}(\boldsymbol{x}), S^*(\boldsymbol{x})) \xrightarrow{p} 0$.

We apply Proposition 1 at $\boldsymbol{\zeta} = H(\boldsymbol{x})$ and use the continuous mapping argument with $\widehat{H}(\boldsymbol{x}) \to H(\boldsymbol{x})$ uniformly. The Lipschitz bound from **A1** transfers uniform convergence from fixed $\boldsymbol{\zeta}$ to the deployed map $\widehat{H}(\cdot)$, while consistency of $(\hat{F}_n, \hat{Q}_n)$ and residual stability (**A3**–**A5**) ensures scenario perturbations vanish (the proof follows Proposition EC.1 in Kannan et al. [2025]).