JAWS: Auditing Predictive Uncertainty Under Covariate Shift

Drew Prinster
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21211
drew@cs.jhu.edu

Anqi Liu
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21211
aliu@cs.jhu.edu

Suchi Saria
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21211
ssaria@cs.jhu.edu

Abstract

We propose **JAWS**, a series of wrapper methods for distribution-free uncertainty quantification tasks under covariate shift, centered on the core method **JAW**, the **JA**ckknife+ **W**eighted with data-dependent likelihood-ratio weights. JAWS also includes computationally efficient **A**pproximations of JAW using higher-order influence functions: **JAWA**. Theoretically, we show that JAW relaxes the jackknife+'s assumption of data exchangeability to achieve the same finite-sample coverage guarantee even under covariate shift. JAWA further approaches the JAW guarantee in the limit of the sample size or the influence function order under common regularity assumptions. Moreover, we propose a general approach to repurposing predictive interval-generating methods and their guarantees to the reverse task: estimating the probability that a prediction is erroneous, based on user-specified error criteria such as a safe or acceptable tolerance threshold around the true label. We then propose **JAW-E** and **JAWA-E** as the repurposed proposed methods for this **E**rror assessment task. Practically, JAWS outperform state-of-the-art predictive inference baselines in a variety of biased real world data sets for interval-generation and error-assessment predictive uncertainty auditing tasks.

1 Introduction

Auditing the uncertainty under data shift Principled quantification of predictive uncertainty is crucial for enabling users to calibrate how much they should or should not trust a given prediction [Thiebes et al., 2021, Ghosh et al., 2021, Tomsett et al., 2020, Bhatt et al., 2021]. Uncertainty-based predictor auditing can be considered a type of uncertainty quantification performed *post-hoc*, for example by a regulator without detailed knowledge of a predictor's architecture and with limited resources [Schulam and Saria, 2019]. Data shift poses a major challenge to uncertainty quantification due to violation of the common assumption that the training and test data are exchangeable, or more specifically independent and identically distributed (i.i.d.) [Ovadia et al., 2019, Ulmer et al., 2020, Zhou and Levine, 2021, Chan et al., 2020]. Therefore, it is essential to develop convenient tools for users or regulators to audit the uncertainty of a given prediction even when training data is biased.

Predictor auditing: Interval generation In this work we distinguish between two types of predictive uncertainty auditing. We describe the first type as *interval generation*, which refers to a common

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

goal in the distribution-free uncertainty quantification literature: to generate a predictive confidence interval (or set) that covers the true label with at least a user-specified probability. For instance, an auditor might ask for predictive intervals that contain the true label with at least, say 90% frequency.

Predictor auditing: Error assessment While predictive interval generation has been a central focus of the distribution-free uncertainty quantification literature [Angelopoulos and Bates, 2021], in some applications the reverse computation may be more actionable: estimating the probability that the true test label is erroneous or not, based on user-specified error critieria such as a safe or acceptable tolerance region around the true label. We thus refer to this task as *error assessment*. For instance, take the setting of chemical or radiation therapy dose prediction for cancer treatment, where administering a dose within approximately $\pm 10\%$ of the optimal dose is considered safety-critical (see Appendix A.1). Whereas predictive interval generation could fail to provide safety assurance (e.g., if the predictive confidence interval is larger than the safe tolerance region), error assessment would give a worst-case probability of the prediction being safe. Similar examples could be formulated in other applications, such as incision planning in surgical robotics and autonomous vehicle navigation.

Coverage We assume a standard regression setup with a multiset of training data $\{(X_1, Y_1), ..., (X_n, Y_n)\}$ and a test point (X_{n+1}, Y_{n+1}), where $(X_i, Y_i) \in \mathbb{R}^d \times \mathbb{R}$ for all $i \in \{1, ..., n+1\}$. Also, we denote a predictor as $\widehat{\mu} = \mathcal{A}(\{(X_1, Y_1), ..., (X_n, Y_n)\})$, where $\mathcal{A}$ is a model-fitting algorithm. For predictive interval (or region) $\widehat{C}^{\text{audit}}_{n,\alpha} : \mathbb{R}^d \to \{\text{subsets of } \mathbb{R}\}$, a *coverage guarantee* gives a lower bound to the probability that the interval covers a test point's true label:

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}^{\text{audit}}_{n,\alpha}(X_{n+1})\} \geq 1 - \alpha. \tag{1}$$

The coverage guarantee provides the basis for both interval-generation and error-assessment auditing, though it is important to note that in this work we focus on marginal rather than conditional coverage (see [Foygel Barber et al., 2021] for more details on this distinction). Standard conformal prediction methods [Vovk et al., 2005, Shafer and Vovk, 2008, Vovk, 2013] along with the jackknife+ and related methods [Barber et al., 2021], which we refer to together as "predictive inference" methods, provide a framework for generating predictive intervals with finite-sample guaranteed coverage.

Exchangeability Standard conformal prediction and the jackknife+ rely on two crucial notions of *exchangeability*: data exchangeability, that is that the training and test data are all exchangeable (e.g., i.i.d.); and secondly that the model-fitting algorithm $\mathcal{A}$ treats the data symmetrically [Barber et al., 2022]. In common situations of dataset shift, however, the data exchangeability assumption is violated. Empirically, the coverage performance of standard conformal prediction methods can suffer under data shift [Tibshirani et al., 2019, Podkopaev and Ramdas, 2021].

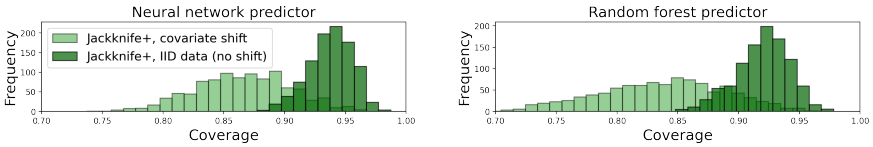


Figure 1: Jackknife+ loses coverage on the airfoil dataset under covariate shift (details in Section 4).

In this work, we build on the jackknife+ method due to its beneficial compromise between the statistical and computational limitations of other conformal prediction methods [Barber et al., 2021]. However, jackknife+ coverage performance can still degrade under data shift, such as shown in Figure 1, and in some applications its computational requirements can still be limiting. To address these concerns and make extensions to error assessment, we develop JAWS, a series of wrapper methods for distribution-free uncertainty quantification under covariate shift (see Table 1 for key properties).

Table 1: Summary of key properties for JAWS methods (details in Section 3).

Method	**Task**	**Guarantee (under covariate shift)**		**Avoids retraining**
		Finite sample	**Asymptotic**	
JAW	Interval generation	✓	✓	✗
JAWA	Interval generation	✗	✓	✓
JAW-E	Error assessment	✓	✓	✗
JAWA-E	Error assessment	✗	✓	✓

Our contributions can be summarized as follows:

1. We develop JAW: a jackknife+ method with data-dependent likelihood-ratio weights for predictive interval generation under covariate shift. We show that JAW achieves the same rigorous, finite-sample coverage guarantee as jackknife+ [Barber et al., 2021] while relaxing the data exchangeability assumption to allow for covariate shift.
2. We develop JAWA: a sequence of computationally efficient approximations to JAW that uses higher-order influence functions to avoid retraining. Under assumptions outlined in Giordano et al. [2019a] regarding the regularity of the data, Hessian of the objective (local strong convexity), and the existence and boundedness of higher order derivatives, we provide an asymptotic guarantee for the JAWA coverage in the limit of the sample size or influence function order.
3. We propose a general approach to repurposing any distribution-free predictive inference method to the error assessment task, with rigorous guarantees for the coverage probability estimation. Our approach applies to methods that assume exchangeable data and to methods like JAW and JAWA that allow for covariate shift—JAW-E and JAWA-E refer to the error assessment versions.
4. We demonstrate superior empirical performance of JAWS over other distribution-free predictive inference baselines on a variety of benchmark datasets under covariate shift.

2 Background and related work

2.1 Standard conformal prediction

Conformal prediction has grown into a broad research field since arising in the 1990s [Vovk et al., 2005, Shafer and Vovk, 2008, Balasubramanian et al., 2014, Angelopoulos and Bates, 2021]. Standard conformal prediction methods generate a prediction interval (or set) with a finite-sample coverage guarantee as in (1), which is *distribution-free* in the sense that the guarantee applies to any exchangeable data distribution [Lei and Wasserman, 2014, Lei et al., 2018]. With the exchangeability assumptions in Section 1, standard conformal prediction methods rely on a pre-fit score function $\widehat{S} : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ (in regression, the absolute-value residual score $\widehat{S}(x, y) = |y - \widehat{\mu}(x)|$ is commonly used). A conformal prediction interval at confidence level $1-\alpha$ is then determined by a corresponding quantile on a multiset of (exchangeable) score values.

Split conformal and full conformal are two main types of standard conformal prediction, and each bears its own limitation [Vovk et al., 2005, Shafer and Vovk, 2008]. Split conformal generates scores on labeled holdout data and is computationally efficient due to not requiring retraining, but sample splitting to obtain the holdout set can reduce model accuracy [Papadopoulos, 2008, Lei et al., 2018, Vovk, 2012]. On the other hand, full conformal prediction avoids the holdout set requirement, but at the heavy computational cost of retraining the model on every possible target value (or, in practice, on a fine grid of target values) [Ndiaye and Takeuchi, 2019, Zeni et al., 2020].

2.2 Covariate shift

Under the *covariate shift* assumption, the $Y|X$ distribution is assumed to be the same between training and test data but the marginal X distributions may change [Sugiyama et al., 2007, Shimodaira, 2000]:

$$(X_i, Y_i) \overset{\text{i.i.d.}}{\sim} P_X \times P_{Y|X}, i = 1, ..., n; \qquad (X_{n+1}, Y_{n+1}) \sim \tilde{P}_X \times P_{Y|X}, \text{ independently.} \tag{2}$$

Rich literature exist in this domain—see Appendix A.2 for more details. Uncertainty quantification is relatively less explored under covariate shift, though recent work [Ovadia et al., 2019, Zhou and Levine, 2021, Chan et al., 2020] emphasizes its importance, especially in deep learning.

2.3 Conformal prediction under covariate shift and beyond exchangeability

Tibshirani et al. [2019] develop the idea of *weighted exchangeability* for adapting conformal prediction to the covariate shift setting. Random variables $V_1, ..., V_n$ are weighted exchangeable with weight functions $w_1, ..., w_n$ if their joint density f can be factorized as $f(v_1, ..., v_n) = \prod_{i=1}^n w_i(v_i) \cdot g(v_1, ..., v_n)$, where g is independent of ordering on its inputs. For covariate shift as in (2), if $\tilde{P}_X$ is absolutely continuous with respect to P_X, then the data $\{(X_i, Y_i)\}$ are weighted exchangeable with weights given by the likelihood ratio $w(X_i) = \mathrm{d}\tilde{P}_X(X_i)/\mathrm{d}P_X(X_i)$ [Tibshirani et al., 2019].

If $\{v_i\}$ represents a set of scores for standard conformal prediction, then we can represent the empirical distribution of $\{v_i\}$ as $\frac{1}{n+1}\sum_{i=1}^{n}\delta_{v_i} + \frac{1}{n+1}\delta_{\infty}$, where δ_{v_i} denotes a point mass at v_i [Barber et al., 2022]. By extension, weighted conformal prediction uses the *weighted* empirical distribution defined as $\sum_{i=1}^{n} p_i^w(x)\delta_{v_i} + p_{n+1}^w(x)\delta_{\infty}$, with weights given by

$$p_i^w(x) = \frac{w(X_i)}{\sum_{j=1}^{n} w(X_j) + w(x)}, i = 1, ..., n; \qquad \text{and} \qquad p_{n+1}^w(x) = \frac{w(x)}{\sum_{j=1}^{n} w(X_j) + w(x)}, \quad (3)$$

where $p_i^w(x)$ can be thought of as a normalized likelihood ratio weight for each $i \in \{1, ..., n+1\}$. Corollary 1 in [Tibshirani et al., 2019] provides the coverage guarantee of weighted conformal prediction that takes the form of (1) but relaxes the exchangeable data assumption to allow for covariate shift. However, weighted split and weighted full conformal inherit the same statistical and computational limitations, respectively, from their standard (exchangeable) variants.

The recent work of [Barber et al., 2022] provides a novel extension of conformal prediction and the jackknife+ to unknown violations of the exchangeability assumption, including a "nonexchangeable jackknife+" defined with fixed weights. The key difference between the nonexchangeable jackknife+ in Barber et al. [2022] and our proposed JAW method is that Barber et al. [2022] use fixed weights to compensate for unknown exchangeability violations (not limited to covariate shift) but at the expense of a bounded but generally nonzero "coverage gap" (drop in guaranteed coverage relative to if the data were exchangeable), whereas our JAW method with data-dependent weights assumes covariate shift but does not suffer from any similar coverage gap. See Appendix A.3 for more details.

2.4 Jackknife+

The jackknife+ [Barber et al., 2021], which is closely related to cross conformal prediction [Vovk et al., 2018], offers a compromise between the statistical limitation of split conformal and the computational limitation of full conformal, at the cost of a slightly weaker coverage guarantee. The jackknife+ predictive interval can most easily be understood as a modification to a predictive interval from the classic jackknife resampling method [Miller, 1974, Steinberger and Leeb, 2018, 2016]. For a set of point masses $\{\delta_{v_i}\}$ at values $v_1, ..., v_n$, let $Q_\beta^-\{\frac{1}{n+1}\delta_{v_i}\}$ denote the level β quantile on the empirical distribution $\sum_{i=1}^{n}\frac{1}{n+1}\delta_{v_i} + \frac{1}{n+1}\delta_{-\infty}$ and let $Q_\beta^+\{\frac{1}{n+1}\delta_{v_i}\}$ denote the level β quantile on the empirical distribution $\sum_{i=1}^{n}\frac{1}{n+1}\delta_{v_i} + \frac{1}{n+1}\delta_{\infty}$. Then, denoting the model trained without the ith point as $\widehat{\mu}_{-i} = \mathcal{A}\big(\{(X_1, Y_1), ..., (X_{i-1}, Y_{i-1}), (X_{i+1}, Y_{i+1}), ..., (X_n, Y_n)\}\big)$ and the leave-one-out residual $R_i^{LOO} = |Y_i - \widehat{\mu}_{-i}(X_i)|$, the jackknife prediction interval can be written as

$$\widehat{C}_{n,\alpha}^{\text{jackknife}}(X_{n+1}) = \Big[Q_\alpha^-\Big\{\tfrac{1}{n+1}\delta_{\widehat{\mu}(X_{n+1}) - R_i^{LOO}}\Big\},\ Q_{1-\alpha}^+\Big\{\tfrac{1}{n+1}\delta_{\widehat{\mu}(X_{n+1}) + R_i^{LOO}}\Big\}\Big]. \quad (4)$$

In contrast, we obtain the jackknife+ predictive interval in Barber et al. [2021] by replacing the full model prediction $\widehat{\mu}(X_{n+1})$ in (4) with $\widehat{\mu}_{-i}(X_{n+1})$:

$$\widehat{C}_{n,\alpha}^{\text{jackknife+}}(X_{n+1}) = \Big[Q_\alpha^-\Big\{\tfrac{1}{n+1}\delta_{\widehat{\mu}_{-i}(X_{n+1}) - R_i^{LOO}}\Big\},\ Q_{1-\alpha}^+\Big\{\tfrac{1}{n+1}\delta_{\widehat{\mu}_{-i}(X_{n+1}) + R_i^{LOO}}\Big\}\Big]. \quad (5)$$

[Barber et al., 2021] prove that, with the same exchangeability assumptions as in standard conformal prediction, the jackknife+ prediction interval satisfies

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}_{n,\alpha}^{\text{jackknife+}}(X_{n+1})\} \geq 1 - 2\alpha. \quad (6)$$

2.5 Approximating leave-one-out models with higher-order influence functions

Influence functions (IFs) [Cook, 1977] have a long history in robust statistics for estimating the dependence of parameters on sample data. Recently, IFs have become more widespread in machine learning for uses including model interpretability Koh and Liang [2017] and approximating classic resampling-based uncertainty quantification methods including bootstrap [Schulam and Saria, 2019], jackknife, and leave-k-out cross validation [Giordano et al., 2019b,a]. In each of these cases, IFs enable approximation of the parameters that would be obtained if the model were retrained on resampled data by instead estimating the effect of a corresponding reweighting. In prior work Alaa and Van Der Schaar [2020] proposed approximating the leave-one-out models required by the jackknife+ with higher-order IFs, but their work assumes exchangeable or i.i.d. train and test data.

Let $\hat{\theta}$ denote the fitted parameters for predictor $\widehat{\mu}$ trained on the full training data. Given Assumptions 1-4 in Giordano et al. [2019a]—which require that $\hat{\theta}$ is a local minimum of the objective function, that the objective is $k+1$ times continuously differentiable with bounded norms, and that the objective is strongly convex in the neighborhood of $\hat{\theta}$—then the k-th order leave-one-out IF refers to the k-th order directional derivative of the model parameters $\hat{\theta}$ with respect to the data weights, in the direction of the leave-one-out change in weights (See A.4 for more details). With each of these kth order leave-one-out IFs for $k \in \{1, ..., K\}$, denoted with condensed notation $\delta^k_{-i}\hat{\theta}$, we can construct a K-th order Taylor series approximation to estimated the leave-one-out model parameters $\hat{\theta}_{-i}$:

$$\hat{\theta}^{\text{IF-}K}_{-i} := \hat{\theta} + \sum_{k=1}^{K} \frac{1}{k!} \delta^k_{-i}\hat{\theta}. \tag{7}$$

In this work we implement the algorithm proposed by Giordano et al. [2019a] to compute higher-order IFs, a recursive procedure based on foreward-mode automatic differentiation [Maclaurin et al., 2015] for memory efficiency in computing higher-order directional derivatives. Our introduction of IFs is highly simplified—we refer to Appendix A.4 and to Giordano et al. [2019a] for more details.

2.6 Error assessment

Whereas conformal prediction and related methods generate prediction intervals that control the error probability (miscoverage level α) at a user-specified level, we refer to the reverse task as *error assessment*: estimating the probability that a prediction is erroneous or not, based on user-specified error criteria. For instance, a user might define an error as any deviation between the prediction $\widehat{\mu}(X_{n+1})$ and the true label Y_{n+1} greater than some acceptable tolerance threshold τ: that is, when $|Y_{n+1} - \widehat{\mu}(X_{n+1})| > \tau$. In Section 3.3, we present a general approach to repurposing predictive inference methods with validity under covariate shift to error assessment.

We note that for score functions that are monotonic in y, such as $\widehat{S}(x, y) = y - \widehat{\mu}(x)$, guarantees for this error assessment task can be obtained using conformal predictive *distributions* as described by Vovk et al. [2017] (also see Vovk et al. [2020], Vovk and Bendtsen [2018], Xie and Zheng [2022]). In regression tasks assuming exchangeable data, CPDs generate a probability distribution for the label over $\mathbb{R}$. However, CPDs require score functions that are monotonic in y, whereas we allow for non-monotone conformity scores such as the commonly used absolute-value residual $|y - \widehat{\mu}(x)|$; Moreover, CPDs assume exchangeable data, whereas our approach extends to covariate shift.

3 Proposed approach and theoretical results

3.1 JAW: Jackknife+ weighted with data-dependent weights

We present **JAW**, the **JA**ckknife+ **W**eighted with data-dependent likelihood-ratio weights, defined by the following predictive interval:

$$\widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1}) = \Big[Q^-_\alpha\big\{p^w_i(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}(X_{n+1}) - R^{LOO}_i}\big\},\ Q^+_{1-\alpha}\big\{p^w_i(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}(X_{n+1}) + R^{LOO}_i}\big\}\Big], \tag{8}$$

where $R^{LOO}_i = \big|\widehat{\mu}_{-i}(X_i) - Y_i\big|$, where the $p^w_i(x)$ are as defined in (3), where $Q^-_\alpha\big\{p^w_i(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}(X_{n+1}) - R^{LOO}_i}\big\}$ denotes the level α quantile of the empirical distribution $\sum_{i=1}^n \big[p^w_i(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}(X_{n+1}) - R^{LOO}_i}\big] + p^w_{n+1}(X_{n+1}) \cdot \delta_{-\infty}$, and where $Q^+_{1-\alpha}\big\{p^w_i(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}(X_{n+1}) + R^{LOO}_i}\big\}$ is the level $1-\alpha$ quantile for $\sum_{i=1}^n \big[p^w_i(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}(X_{n+1}) + R^{LOO}_i}\big] + p^w_{n+1}(X_{n+1}) \cdot \delta_{\infty}$.

We show that $\widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1})$ satisfies the same coverage guarantee as the jackknife+ except relaxing the data exchangeability assumption to allow for covariate shift, which we state formally as follows:

Theorem 1. *Assume data under covariate shift from* (2)*. If $\tilde{P}_X$ is absolutely continuous with respect to P_X, then the JAW interval in* (8) *satisfies*

$$\mathbb{P}\big\{Y_{n+1} \in \widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1})\big\} \geq 1 - 2\alpha \tag{9}$$

Remark 1. The results from Tibshirani et al. [2019] do not directly imply Theorem 1. Applying the approach from Tibshirani et al. [2019] to the jackknife+ entails treating $\widehat{\mu}_{-i}(X_{n+1}) \pm R_i^{LOO}$ as implicit nonconformity scores. Tibshirani et al. [2019] assume that the nonconformity scores for the training data $\{V_1, ..., V_n\}$ are weighted exchangeable with the nonconformity score for the test point V_{n+1}. But, observe that for $i \in \{1, ..., n\}$, $\widehat{\mu}_{-i}$ is trained on $n-1$ datapoints, whereas $\widehat{\mu}_{-(n+1)} = \widehat{\mu}$ is trained on n datapoints. Thus, no reweighting can make $\widehat{\mu}_{-i}$ equivalent to $\widehat{\mu}$ in distribution, and therefore $\widehat{\mu}_{-i}(X_{n+1}) \pm R_i^{LOO}$ and $\widehat{\mu}(X_{n+1}) \pm R_{n+1}^{LOO}$ are not weighted exchangeable.

Proof sketch: Our proof technique for Theorem 1 extends the jackknife+ coverage guarantee proof in Barber et al. [2021] to the covariate shift setting for JAW using likelihood ratio weights as in Tibshirani et al. [2019]. The outline is as follows:

Setup: Following Barber et al. [2021], we define a set of leave-*two*-out models $\{\tilde{\mu}_{-(i,j)}\}$. We then generalize the notion of "strange" points described in Barber et al. [2021] to covariate shift.

1. *Bounding the total normalized weight of strange points*: We establish deterministically that the total normalized weight of strange points cannot exceed 2α.
2. *Weighted exchangeability using the leave-two-out models*: Using the leave-two-out model construction, we leverage weighted exchangeability to show that the probability that a test point $n+1$ is strange is thus bounded by 2α.
3. *Connection to JAW*: Lastly, we show that the JAW interval can only fail to cover the test label value Y_{n+1} if $n+1$ is a strange point.

While JAW assumes access to oracle likelihood ratio weights, in practice this information often has to be estimated. See Appendix D.5 for a discussion and experiments of JAW with estimated weights.

3.2 JAWA: Using higher-order influence functions to approximate JAW without retraining

For computationally efficient JAW **A**pproximations that avoid retraining n leave-one-out models, we propose the JAWA sequence, which approximates the leave-one-out models required by JAW using higher-order influence functions. For each training point $i \in \{1, ..., n\}$, define the K-th order influence function approximation to the leave-one-out refit parameters $\hat{\theta}_{-i}$, obtained from Algorithm 4 in Giordano et al. [2019a], as given by equation (7), and let $\widehat{\mu}_{-i}^{\text{IF-}K}$ be the model with with these approximated parameters $\hat{\theta}_{-i}^{\text{IF-}K}$ for each $i \in \{1, ..., n\}$. Then, the prediction interval for the K-th order JAWA (i.e., for JAWA-K) is given by

$$\widehat{C}_{n,\alpha}^{\text{JAWA-}K}(X_{n+1}) = \Big[Q_\alpha^-\big\{p_i^w(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}^{\text{IF-}K}(X_{n+1}) - R_i^{\text{IF-}K,LOO}}\big\}, \\ Q_{1-\alpha}^+\big\{p_i^w(X_{n+1}) \cdot \delta_{\widehat{\mu}_{-i}^{\text{IF-}K}(X_{n+1}) + R_i^{\text{IF-}K,LOO}}\big\}\Big], \tag{10}$$

with $R_i^{\text{IF-}K,LOO} = \big|\widehat{\mu}_{-i}^{\text{IF-}K}(X_i) - Y_i\big|$, $p_i^w(x)$ as in (3), and quantiles defined analogously to JAW.

We now provide an asymptotic coverage guarantee for $\widehat{C}_{n,\alpha}^{\text{JAWA-}K}(X_{n+1})$ that holds either in the limit of the sample size or in the limit of the influence function order, under regularity conditions formally described in Giordano et al. [2019a]. These assumptions concern the regularity and continuity of the training data, local convexity of the objective (or that the Hessian is strongly positive definite), and the existence and boundedness of the objective's 1st through $K+1$th order directional derivatives.

Theorem 2. *Assume data under covariate shift from* (2) *and that $\tilde{P}_X$ is absolutely continuous with respect to P_X. Let Assumptions 1 - 4 and either Condition 2 or Condition 4 from Giordano et al. [2019a] hold uniformly for all n. Then, in the limit of the training sample size $n \to \infty$ or in the limit of the influence function order $K \to \infty$, the JAWA-K interval in* (10) *satisfies*

$$\mathbb{P}\big\{Y_{n+1} \in \widehat{C}_{n,\alpha}^{\text{JAWA-}K}(X_{n+1})\big\} \geq 1 - 2\alpha. \tag{11}$$

We leave the proof to Appendix C.2, but we note that the result follows by combining Propositions 1 and 3 in Giordano et al. [2019a] with the JAW coverage guarantee that we present in Theorem 1.

3.3 Error assessment under covariate shift

We now propose a general approach to repurposing predictive inference methods with validity under covariate shift from predictive interval generation to the reverse task: estimating the probability that a

prediction is erroneous or not, based on user-specified error criteria. For example, consider a user that defines a prediction $\widehat{\mu}(X_{n+1})$ as wrong, relative to the true label Y_{n+1}, if $|Y_{n+1} - \widehat{\mu}(X_{n+1})| > \tau$. For this commonly-used error criterion, our approach to adapting a method such as JAW (8) or weighted split conformal prediction [Tibshirani et al., 2019] to error assessment reduces to first defining the set of labels that would not be considered erroneous, $\overline{E} = [\widehat{\mu}(X_{n+1}) - \tau,\ \widehat{\mu}(X_{n+1}) + \tau]$, and then finding the given method's largest predictive interval contained within $\overline{E}$, call it $\widehat{C}^{\text{w-audit}}_{n,\alpha_E}(X_{n+1})$. The coverage guarantee for $\widehat{C}^{\text{w-audit}}_{n,\alpha_E}(X_{n+1})$ then yields a bound on $\mathbb{P}\{Y_{n+1} \in \overline{E}\}$, the probability of no error. See Figure 2 for an illustration of this example.

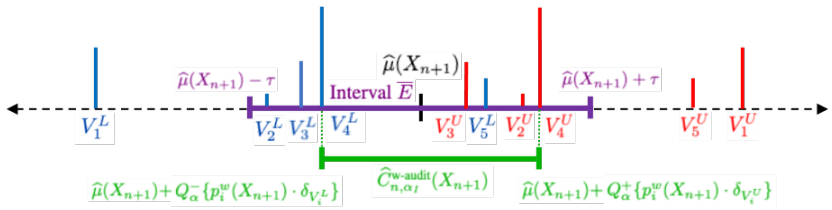


Figure 2: Illustration of approach to repurposing a predictive inference method "w-audit" to error assessment. The interval $\overline{E} = [\widehat{\mu}(X_{n+1}) - \tau,\ \widehat{\mu}(X_{n+1}) + \tau]$ is shown in violet, the lower score values $\{V_i^L\}$ in blue, the upper score values $\{V_i^U\}$ in red, and the interval $\widehat{C}^{\text{w-audit}}_{n,\alpha_E}(X_{n+1})$ in green. Each vertical line at a location V_i on the real line represents a point mass δ_{V_i} with height corresponding to the normalized likelihood ratio weight $p_i^w(X_{n+1})$.

More generally, our approach allows for other error criteria (e.g., that might not be defined symmetrically around the prediction) by requiring that a user specify upper and lower score functions $\widehat{S}^{\pm} : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ and corresponding upper and lower tolerance values $\tau^{\pm}$, such that that a prediction $\widehat{\mu}(X_{n+1})$ is considered erroneous if $\widehat{S}^-(X_{n+1}, Y_{n+1}) < \tau^-$ or if $\tau^+ < \widehat{S}^+(X_{n+1}, Y_{n+1})$. (Note that if a user wishes to specify the error criterion $|Y_{n+1} - \widehat{\mu}(X_{n+1})| > \tau$, this amounts to letting $\widehat{S}^{\pm}(x, y) = \pm|y - \widehat{\mu}(x)|$ and $\tau^{\pm} = \pm\tau$.) For a prediction $\widehat{\mu}(X_{n+1})$, let us define the values of y such that setting $Y_{n+1} = y$ would *not* cause $\widehat{\mu}(X_{n+1})$ to be considered an error as

$$\overline{E} = \left\{y \in \mathbb{R} : \tau^- \leq \widehat{S}^-(X_{n+1}, y),\ \widehat{S}^+(X_{n+1}, y) \leq \tau^+\right\}. \tag{12}$$

Now, let us assume access to a predictive inference method with predictive sets given by

$$\begin{aligned}\widehat{C}^{\text{w-audit}}_{n,\alpha}(X_{n+1}) = \big\{y : \widehat{Q}^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L}\} \leq \widehat{S}^-(X_{n+1}, y),\\ \widehat{S}^+(X_{n+1}, y) \leq \widehat{Q}^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^U}\}\big\}\end{aligned} \tag{13}$$

with coverage guarantee under covariate shift. Note that we can obtain the JAW prediction interval from (13) using score functions $\widehat{S}^{\pm}(x, y) = \pm|y - \widehat{\mu}(x)|$ and letting $V_i^L = \widehat{\mu}_{-i}(X_{n+1}) - R_i^{LOO}$ and $V_i^U = \widehat{\mu}_{-i}(X_{n+1}) + R_i^{LOO}$). Define $\alpha_E^{\text{w-audit}}$ as

$$\alpha_E^{\text{w-audit}} = \min\left(\left\{\alpha' \ :\ \tau^- \leq\ \widehat{Q}^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L}\}\ ,\ \widehat{Q}^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^U}\} \leq \tau^+\right\}\right). \tag{14}$$

We can then estimate the probability of $\widehat{\mu}(X_{n+1})$ *not* resulting in an error as in (12) as:

$$\widehat{p}\{Y_{n+1} \in \overline{E}\} = \begin{cases} 1 - \alpha_E^{\text{w-audit}} & \text{if } \alpha_E^{\text{w-audit}} \text{ exists} \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

While the target coverage for $\widehat{C}^{\text{w-audit}}_{n,\alpha_E}(X_{n+1})$ is used in (15), the following theorem gives the worst-case error assessment guarantee for covariate shift (proof in Appendix C.3). Corollary 1 in Appendix B.2 and Corollary 2 in Appendix B.3 give the error assessment guarantees for JAW-E and JAWA-E respectively. Theorem 4 in Appendix B.1 gives the analogous guarantee for exchangeable data.

Theorem 3. *Given a predictive inference method with predictive sets as in* (13) *with coverage guarantee* $\mathbb{P}\{Y_{n+1} \in \widehat{C}^{\text{w-audit}}_{n,\alpha}(X_{n+1})\} \geq 1 - c_1\alpha - c_2$ *under covariate shift, where* $c_1, c_2 \in \mathbb{R}$. *Define* $\overline{E}$ *as in* (12) *and* $\alpha_E^{\text{w-audit}}$ *as in* (14). *Then,*

$$\mathbb{P}\{Y_{n+1} \in \overline{E}\} \leq \begin{cases} 1 - c_1\alpha_E^{\text{w-audit}} - c_2 & \text{if } \alpha_E^{\text{w-audit}} \text{ exists and } \alpha_E^{\text{w-audit}} < \frac{1-c_2}{c_1} \\ 0 & \text{otherwise} \end{cases}. \tag{16}$$

4 Experiments[1]

4.1 Datasets and creation of covariate shift

We conduct experiments on five UCI datasets Dua and Graff [2017] with various dimensionality (Table 2): airfoil self-noise, red wine quality prediction [Cortez et al., 2009], wave energy converters, superconductivity [Hamidieh, 2018], and communities and crime [Redmond and Baveja, 2002].

Table 2: Statistics for the UCI datasets. Only the first 2000 samples were used for the wave and superconductivity datasets (for wave, the first 2000 samples of Adelaide data).

Dataset	# of samples	# of features	label range
Airfoil self-noise (airfoil)	1503	5	[103.38, 140.987]
Red wine quality (wine)	1599	11	[3, 8]
Wave energy converters (wave)	2000	48	[1226969, 1449349]
Superconductivity (superconduct)	2000	81	[0.2, 136.0]
Communities and crime (communities)	1994	99	[0, 1]

We use exponential tilting to induce covariate shift on the test data, based on the approach used in Tibshirani et al. [2019]. We first randomly sample 200 points for the training data, and then sample the biased test data from the remaining datapoints that are not used for training with probabilities proportional to exponential tilting weights. See Appendix D.1 for additional details.

4.2 Baselines

Baselines for comparison to JAW We compared JAW to the following baselines:

1. **Naive** estimates are based on training data residuals $|Y_i - \widehat{\mu}(X_i)|$, which suffers from overfitting.
2. **Jackknife** uses the classic Jackknife resampling as in (4).
3. **Jackknife+** follows (5), which replaces the prediction $\widehat{\mu}(X_{n+1})$ in jackknife with $\widehat{\mu}_{-i}(X_{n+1})$.
4. **Jackknife-mm** is a more conservative alternative to the jackknife+ method that guarantees coverage at the $1-\alpha$ level with exchangeable data, but usually with overly-wide intervals.
$$\widehat{C}^{\text{jackknife-mm}}_{n,\alpha}(X_{n+1}) = \Big[\min_{i=1,\dots,n} \widehat{\mu}_{-i}(X_{n+1}) - Q^{+}_{1-\alpha}\{R^{LOO}_i\}, \max_{i=1,\dots,n} \widehat{\mu}_{-i}(X_{n+1}) + Q^{+}_{1-\alpha}\{R^{LOO}_i\}\Big]$$
5. **Cross validation+** (CV+) is similar to jackknife+ except splits data into k portions and replaces the $\widehat{\mu}_{-i}(X_{n+1})$ with $\widehat{\mu}_{-k}(X_{n+1})$, the model trained with the kth subset removed.
6. **Split** method follows split conformal prediction, which uses half the data for training and the other half for generating the nonconformity scores.
7. **Weighted split** is a version of split conformal with likelihood ratio weights to maintain coverage under covariate shift, as in Tibshirani et al. [2019].

Baselines for comparison to JAWA For influence function orders $K \in \{1, 2, 3\}$, we compared the proposed JAWA-K method with K-th order influence function approximations of the jackknife-based baselines that we used as comparisons to JAW—we thus refer to these approximations as IF-K jackknife, IF-K jackknife+, and IF-K jackknife-mm. Each baseline compared to JAWA-K is thus also approximated with the same K-th order leave-one-out influence function models.

4.3 Experimental results

We report experimental results on the predictive interval-generation task for both JAW and JAWA and on the error assessment task for JAW, compared to baselines. Additional experimental details and supplementary experiments can be found in Appendix D, including for estimated likelihood ratio weights in D.5, ablation study with shift magnitudes in D.6, and coverage histograms in D.9.

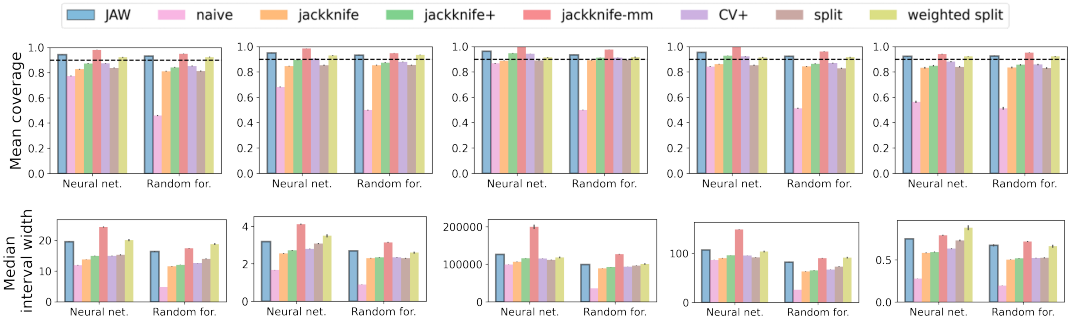
4.3.1 Interval generation results for JAW: Coverage and interval width

Figure 3 compares JAW and its baselines, firstly regarding mean coverage and secondarily regarding median interval width, on all five UCI datasets for both neural network and random forest predictors,

[1]Additional experimental analysis and a link to code for all experiments and included in Appendices D and E

averaged over 200 experimental replicates. See Appendix D.2 for predictor function details. Meeting the target coverage level of $1 - \alpha$ is the primary goal of the interval-generation audit task, but for methods that meet or nearly meet the target coverage level, smaller interval widths are more informative. Additionally, smaller variance in coverage indicates a more reliable or consistent method.

As seen in Figure 3, the JAW predictive interval coverage is above the target level of 0.9 across all datasets, for both random forest and neural network $\widehat{\mu}$ functions, along with the jackknife-mm and weighted split methods. However, JAW's interval widths are generally smaller and thus more informative than those of jackknife-mm (which are often overly large, as noted in Barber et al. [2021]). Weighted split and JAW perform similarly on mean coverage and median interval width (both methods have coverage guarantees under covariate shift), but JAW avoids sample splitting and as a result has lower coverage variance than weighted split for all dataset and predictor conditions (see Appendix D.3), which suggests that JAW's predictive intervals are more reliable.



(a) Airfoil (b) Wine (c) Wave (d) Superconduct (e) Communities

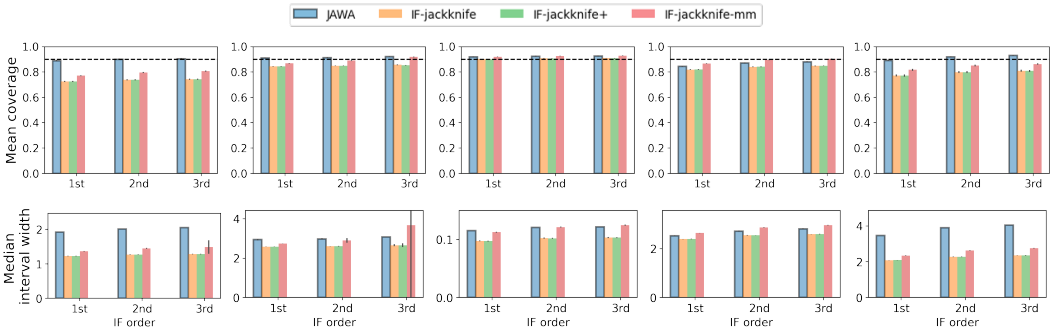
Figure 3: Mean coverage (first row) and median interval width (second row) for neural network and random forest predictors on UCI datasets. Dashed line is the target coverage level ($1 - \alpha$ = 0.9). Error bars show the standard error of 1000 repeated experiments. JAW maintains target coverage under covariate shift for all predictor and dataset conditions along with jackknife-mm and weighted split—however, JAW's intervals are generally smaller and thus more informative than jackknife-mm's, and JAW's coverage variance is smaller and thus more reliable than weighted split's (Appendix D.3).

4.3.2 Interval generation results for JAWA: Coverage and interval width

Figure 4 evaluates JAWA coverage and interval width compared to baselines for IF orders $K \in \{1, 2, 3\}$ with a neural network predictor (see Appendix D.2 for predictor details). As with the JAW experiments, coverage at the target level of $1-\alpha = 0.9$ is the primary goal, while secondarily, smaller intervals are more informative for methods that meet or nearly meet target coverage. For three of the five datasets (airfoil, wine, and communities), JAWA is the only method that consistently reaches or nearly reaches the target coverage level. JAWA and all the baselines perform well on the wave datasets, and in the superconduct dataset JAWA still outperforms approximations of jackknife and jackknife+ for all IF orders. Appendix D.7 provides an an example empirical comparison of JAWA and JAW runtimes, which demonstrates that JAWA can be orders of magnitude faster to compute.

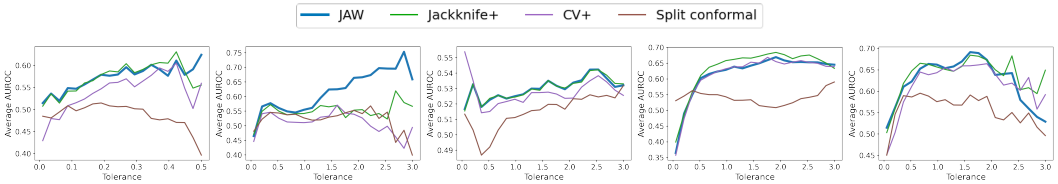
4.3.3 Error assessment results for JAW-E: AUC

We now turn to an error-assessment audit task where the goal is to evaluate a method's ability to estimate the probability that a given prediction is erroneous or not, based on the criterion $|Y_{n+1} - \widehat{\mu}(X_{n+1})| > \tau$. Let $\overline{E} = [\widehat{\mu}(X_{n+1}) - \tau,\ \widehat{\mu}(X_{n+1}) + \tau]$. Then, the goal is to estimate the probability that $\widehat{\mu}(X_{n+1})$ is correct, i.e., $Y_{n+1} \in \overline{E}$; or an error, i.e., $Y_{n+1} \notin \overline{E}$. For four predictive interval-generation methods repurposed to the error assessment task (JAW-E, jackknife+E, cross validation+E, and split conformal-E), Figure 5 reports the area under the receiver operating characteristic curve (AUROC) for values of τ selected with uniform spacing in the interval $[q^{-}_{n,0.05}\{R^{LOO}_i\},\ q^{+}_{n,0.05}\{R^{LOO}_i\}]$ (between the 0.05 and 0.95 quantiles of the leave-one-out residuals for a dataset) for five repeated experiments with the random forest predictor. Better performing methods have higher overall AUROC values for all values τ. JAW achieves AUROC values comparable to jackknife+ and CV+ for most tolerance levels and datasets. JAW moreover achieves higher



(a) Airfoil (b) Wine (c) Wave (d) Superconduct (e) Communities

Figure 4: Mean coverage (first row) and median interval width (second row) for JAWA and baselines for influence function orders $K \in \{1, 2, 3\}$. Dashed line is the target coverage level ($1 - \alpha = 0.9$). Error bar shows the standard error of 200 repeated experiments. JAWA is more consistent than baselines in reaching or nearly reaching the target coverage level across datasets and influence function orders, and it is more computationally efficient than JAW (Appendix D.7).



(a) Airfoil (b) Wine (c) Wave (d) Superconduct (e) Communities

Figure 5: AUROC values for tolerance levels τ across the three datasets for the random forest predictor, averaged across 5 experiment replicates. Results for neural net predictor in Appendix D.4.

AUROC values than its baselines in the wine dataset with the random forest predictor. These results suggest that, for error assessment, JAW likely performs at least as well as its baselines and may offer a marginal benefit depending on the dataset and predictor.

5 Conclusion

In this paper, we develop JAWS, a series of wrapper methods for distribution-free predictive uncertainty auditing tasks when the data exchangeability assumption is violated due to covariate shift. We also propose a general approach to repurposing any distribution-free predictive inference method to the error assessment task. We provide rigorous finite-sample guarantees for JAW and JAW-E on the interval generation and error assessment tasks respectively, and analogous asymptotic guarantees for the computationally efficient JAWA and JAWA-E. We moreover demonstrate superior performance of the JAWS series on a variety of datasets. In supplementary experiments we investigate a number of JAWS' limitations: weight estimation can address the assumed access to oracle weights with similar empirical performance (Appendix D.5), and JAW's increased coverage variance with covariate shift can be explained by reduced effective sample size due to importance weighting (Appendix D.6). Additionally, we note that JAW and JAWA share a limitation with weighted conformal prediction [Tibshirani et al., 2019] of potentially producing overly large intervals in extreme covariate shift cases where a test point's normalized likelihood ratio approaches or exceeds α. In the future, we aim to address the problems of reducing coverage variance and improving predictive interval sharpness.

Acknowledgments

This work was supported by the National Science Foundation grant IIS-1840088. We thank Yoav Wald for helpful discussions and advice, as well as Peter Schulam for sharing code that facilitated our influence function approximation implementation and AUC experiments.

References

Scott Thiebes, Sebastian Lins, and Ali Sunyaev. Trustworthy artificial intelligence. *Electronic Markets*, 31(2):447–464, 2021.

Soumya Ghosh, Q Vera Liao, Karthikeyan Natesan Ramamurthy, Jiri Navratil, Prasanna Sattigeri, Kush R Varshney, and Yunfeng Zhang. Uncertainty quantification 360: A holistic toolkit for quantifying and communicating the uncertainty of ai. *arXiv preprint arXiv:2106.01410*, 2021.

Richard Tomsett, Alun Preece, Dave Braines, Federico Cerutti, Supriyo Chakraborty, Mani Srivastava, Gavin Pearson, and Lance Kaplan. Rapid trust calibration through interpretable and uncertainty-aware ai. *Patterns*, 1(4):100049, 2020.

Umang Bhatt, Javier Antorán, Yunfeng Zhang, Q Vera Liao, Prasanna Sattigeri, Riccardo Fogliato, Gabrielle Melançon, Ranganath Krishnan, Jason Stanley, Omesh Tickoo, et al. Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 401–413, 2021.

Peter Schulam and Suchi Saria. Can you trust this prediction? auditing pointwise reliability after learning. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1022–1031. PMLR, 2019.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 2019.

Dennis Ulmer, Lotta Meijerink, and Giovanni Cinà. Trust issues: Uncertainty estimation does not enable reliable ood detection on medical tabular data. In *Machine Learning for Health*, pages 341–354. PMLR, 2020.

Aurick Zhou and Sergey Levine. Amortized conditional normalized maximum likelihood: Reliable out of distribution uncertainty estimation. In *International Conference on Machine Learning*, pages 12803–12812. PMLR, 2021.

Alex Chan, Ahmed Alaa, Zhaozhi Qian, and Mihaela Van Der Schaar. Unlabelled data improves bayesian uncertainty calibration under covariate shift. In *International Conference on Machine Learning*, pages 1392–1402. PMLR, 2020.

Anastasios N Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 10(2):455–482, 2021.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(3), 2008.

Vladimir Vovk. Transductive conformal predictors. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pages 348–360. Springer, 2013.

Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486–507, 2021.

Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Conformal prediction beyond exchangeability. *arXiv preprint arXiv:2202.13415*, 2022.

Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in neural information processing systems*, 32, 2019.

Aleksandr Podkopaev and Aaditya Ramdas. Distribution-free uncertainty quantification for classification under label shift. In *Uncertainty in Artificial Intelligence*, pages 844–853. PMLR, 2021.

Ryan Giordano, Michael I Jordan, and Tamara Broderick. A higher-order swiss army infinitesimal jackknife. *arXiv preprint arXiv:1907.12116*, 2019a.

Vineeth Balasubramanian, Shen-Shyang Ho, and Vladimir Vovk. *Conformal prediction for reliable machine learning: theory, adaptations and applications*. Newnes, 2014.

Jing Lei and Larry Wasserman. Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):71–96, 2014.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523): 1094–1111, 2018.

Harris Papadopoulos. *Inductive conformal prediction: Theory and application to neural networks*. INTECH Open Access Publisher Rijeka, 2008.

Vladimir Vovk. Conditional validity of inductive conformal predictors. In *Asian conference on machine learning*, pages 475–490. PMLR, 2012.

Eugene Ndiaye and Ichiro Takeuchi. Computing full conformal prediction set with approximate homotopy. *Advances in Neural Information Processing Systems*, 32, 2019.

Gianluca Zeni, Matteo Fontana, and Simone Vantini. Conformal prediction: a unified review of theory and new challenges. *arXiv preprint arXiv:2005.07972*, 2020.

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(5), 2007.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Vladimir Vovk, Ilia Nouretdinov, Valery Manokhin, and Alexander Gammerman. Cross-conformal predictive distributions. In *Conformal and Probabilistic Prediction and Applications*, pages 37–51. PMLR, 2018.

Rupert G Miller. The jackknife-a review. *Biometrika*, 61(1):1–15, 1974.

Lukas Steinberger and Hannes Leeb. Conditional predictive inference for high-dimensional stable algorithms. *arXiv preprint arXiv:1809.01412*, 2018.

Lukas Steinberger and Hannes Leeb. Leave-one-out prediction intervals in linear regression models with many variables. *arXiv preprint arXiv:1602.05801*, 2016.

R Dennis Cook. Detection of influential observation in linear regression. *Technometrics*, 19(1):15–18, 1977.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

Ryan Giordano, William Stephenson, Runjing Liu, Michael Jordan, and Tamara Broderick. A swiss army infinitesimal jackknife. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1139–1147. PMLR, 2019b.

Ahmed Alaa and Mihaela Van Der Schaar. Discriminative jackknife: Quantifying uncertainty in deep learning via higher-order influence functions. In *International Conference on Machine Learning*, pages 165–174. PMLR, 2020.

Dougal Maclaurin, David Duvenaud, and Ryan P Adams. Autograd: Effortless gradients in numpy. In *ICML 2015 AutoML workshop*, volume 238, 2015.

Vladimir Vovk, Jieli Shen, Valery Manokhin, and Min-ge Xie. Nonparametric predictive distributions based on conformal prediction. In *Conformal and Probabilistic Prediction and Applications*, pages 82–102. PMLR, 2017.

Vladimir Vovk, Ivan Petej, Ilia Nouretdinov, Valery Manokhin, and Alexander Gammerman. Computationally efficient versions of conformal predictive distributions. *Neurocomputing*, 397:292–308, 2020.

Vladimir Vovk and Claus Bendtsen. Conformal predictive decision making. In *Conformal and Probabilistic Prediction and Applications*, pages 52–62. PMLR, 2018.

Min-ge Xie and Zheshi Zheng. Homeostasis phenomenon in conformal prediction and predictive distribution functions. *International Journal of Approximate Reasoning*, 141:131–145, 2022.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4): 547–553, 2009.

Kam Hamidieh. A data-driven statistical model for predicting the critical temperature of a superconductor. *Computational Materials Science*, 154:346–354, 2018.

Michael Redmond and Alok Baveja. A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, 141(3):660–678, 2002.

Mary Feng, Gilmer Valdes, Nayha Dixit, and Timothy D Solberg. Machine learning in radiation oncology: opportunities, requirements, and needs. *Frontiers in oncology*, 8:110, 2018.

Elizabeth Huynh, Ahmed Hosny, Christian Guthier, Danielle S Bitterman, Steven F Petit, Daphne A Haas-Kogan, Benjamin Kann, Hugo JWL Aerts, and Raymond H Mak. Artificial intelligence in radiation oncology. *Nature Reviews Clinical Oncology*, 17(12):771–781, 2020.

Saul N Weingart, Lulu Zhang, Megan Sweeney, and Michael Hassett. Chemotherapy medication errors. *The Lancet Oncology*, 19(4):e191–e199, 2018.

Marcel Van Herk. Errors and margins in radiotherapy. In *Seminars in radiation oncology*, volume 14, pages 52–64. Elsevier, 2004.

Howard Gurney. How to calculate the dose of chemotherapy. *British journal of cancer*, 86(8): 1297–1302, 2002.

Michael R Cohen, Roger W Anderson, Richard M Attilio, Laurence Green, Raymond J Muller, and Jane M Pruemer. Preventing medication errors in cancer chemotherapy. *American Journal of Health-System Pharmacy*, 53(7):737–746, 1996.

Steffen Bickel, Michael Brückner, and Tobias Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10(9), 2009.

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4):5, 2009.

Yaoliang Yu and Csaba Szepesvári. Analysis of kernel mean matching under covariate shift. *arXiv preprint arXiv:1206.4650*, 2012.

Kai Zhang, Vincent Zheng, Qiaojun Wang, James Kwok, Qiang Yang, and Ivan Marsic. Covariate shift in hilbert space: A solution via sorrogate kernels. In *International Conference on Machine Learning*, pages 388–395. PMLR, 2013.

Yin Zhao, Longjun Cai, et al. Reducing the covariate shift by mirror samples in cross domain alignment. *Advances in Neural Information Processing Systems*, 34, 2021.

Anqi Liu and Brian Ziebart. Robust classification under sample selection bias. *Advances in neural information processing systems*, 27, 2014.

Xiangli Chen, Mathew Monfort, Anqi Liu, and Brian D Ziebart. Robust covariate shift regression. In *Artificial Intelligence and Statistics*, pages 1270–1279. PMLR, 2016.

John C Duchi, Tatsunori Hashimoto, and Hongseok Namkoong. Distributionally robust losses against mixture covariate shifts. *Under review*, 2, 2019.

Ashkan Rezaei, Anqi Liu, Omid Memarrast, and Brian D Ziebart. Robust fairness under covariate shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9419–9427, 2021.

Luisa Turrin Fernholz. *Von Mises calculus for statistical functionals*, volume 19. Springer Science & Business Media, 2012.

HG Landau. On dominance relations and the structure of animal societies: Iii the condition for a score structure. *The bulletin of mathematical biophysics*, 15(2):143–148, 1953.

Sashank Reddi, Barnabas Poczos, and Alex Smola. Doubly robust covariate shift correction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Samyadeep Basu, Philip Pope, and Soheil Feizi. Influence functions in deep learning are fragile. *arXiv preprint arXiv:2006.14651*, 2020.

Fengpei Li, Henry Lam, and Siddharth Prusty. Robust importance weighting for covariate shift. In *International Conference on Artificial Intelligence and Statistics*, pages 352–362. PMLR, 2020.

Anastasios Angelopoulos, Stephen Bates, Jitendra Malik, and Michael I Jordan. Uncertainty sets for image classifiers using conformal prediction. *arXiv preprint arXiv:2009.14193*, 2020.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Appendix E.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We discuss the limitations of our method in the experiments (Section 4) and the conclusion (Section 5). We aim to resolve them in our future work.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Our work are motivated from the predictor auditing perspective, which aims to facilitate more convenient quality estimation of the predictive models for the users and regulators.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] We include a proof sketch of Theorem 1 in Section 3.1, and have the full proof of all our theoretical results in Appendix C.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We include a link to our GitHub repository in the supplemental material, Appendix E.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4 and Appendix D.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Figure 3 and Figure 4. We repeat our experiments 1000 times for Figure 3 and 200 times for Figure 4 and report the standard error as error bars.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We include the computational details in the supplemental materials, Appenidx E
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We include an anonymous link to our code repository in the supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant errors, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Supplementary background details

A.1 Error assessment motivation: Concrete example

The error-assessment approach to predictor auditing may be more actionable than the interval-generation approach in error-critical or high-stakes decision-making situations where there is a clearly defined margin of error that is considered safe or acceptable. One example is chemical or radiation therapy dose prediction for cancer treatment, where administering the correct dosage within $5\% - 10\%$ is error-critical. Machine learning is being increasingly employed in cancer chemotherapy and radiotherapy for purposes including dose optimization [Feng et al., 2018, Huynh et al., 2020]. Dose errors are one of the most common types of errors in chemotherapy and radiotherapy, occurring when a patient is given a substantially higher or lower than optimal amount of chemical or radiation treatment [Weingart et al., 2018, Van Herk, 2004]. An overdose of either chemotherapeutics or radiation can be harmful or even lethal to a patient, whereas underdose can result in a reduced anticancer effect [Gurney, 2002]. A dose error is generally defined as a percentage deviation, between an administered dose and the truly optimal dose that should have been administered, beyond some error tolerance: 5% or 10% are commonly used deviation thresholds for defining errors [Cohen et al., 1996, Van Herk, 2004]. Accordingly, the probability that the optimal dosage level lies within say 10% of the predicted dosage level may be of greater interest to a provider and their patient than identifying a predictive interval with some predetermined coverage probability (which would provide no error assurance whenever the predictive interval extends beyond the safe threshold, say of $\pm 10\%$).

A.2 Supplementary background on covariate shift

Covariate shift is a type of dataset shift where the $Y|X$ distribution is the same between training and test data but the marginal X distributions are allowed to change [Sugiyama et al., 2007, Shimodaira, 2000]. This is a strong but common assumption for many dataset shift problems. Covariate shift is also closed related to data missingness and sample selection bias [Bickel et al., 2009]. The most prevalent method for correcting the shift is by applying likelihood ratio or "importance" weights [Sugiyama et al., 2007, Shimodaira, 2000]. Density ratio estimation is then a key subproblem of covariate shift correction [Sugiyama et al., 2012]. Other methods dealing with covariate shift include matching the (kernel) representation between the two distributions [Gretton et al., 2009, Yu and Szepesvári, 2012, Zhang et al., 2013, Zhao et al., 2021] and robust optimization [Liu and Ziebart, 2014, Chen et al., 2016, Duchi et al., 2019, Rezaei et al., 2021].

A.3 Supplementary comparison to Barber et al. [2022]

In the Section 2.3 of the main paper we contrast our JAW method to the results in Barber et al. [2022] regarding the nonexchangeable jackknife+. We emphasize that Barber et al. [2022] uses fixed weights and compensates for unknown violations of exchangeability at the expense of a coverage gap, whereas JAW uses data-dependent likelihood ratio weights and assumes covariate shift but does not suffer from a coverage gap. Additionally, it is also important to take note of and contrast our work with an extension of the framework in Barber et al. [2022] to data-dependent weights that the authors briefly discuss in their Section 5.3, subsection titled "Fixed versus data-dependent weights" (though this extension is not a primary focus of their work). In short, this extension from Barber et al. [2022] does not generalize to JAW beyond giving a trivial coverage guarantee.

In particular, Barber et al. [2022] do not propose a likelihood-ratio weighting of the jackknife+, but if one were to define the weights in their nonexchangeable jackknife+ as data-dependent, likelihood ratio weights like in our JAW method, then the extension discussed in Section 5.3 of Barber et al. [2022] would in general suffer a coverage gap that could approach 1 under covariate shift. That is, under covariate shift assumptions and with $w_i = w(X_i)$ representing the likelihood ratio for datapoint i, the conditional total variation distance between the original ordered data $Z = (Z_1, ..., Z_{n+1})$ and the swapped data $Z^i = (Z_1, ..., Z_{i-1}, Z_{n+1}, Z_{i+1}, ..., Z_n)$ can generally approach 1 for nontrivial covariate shift, i.e., $\mathrm{d}_{TV}(Z, Z^i | w_1, ..., w_n, t_1, ..., t_{n+1}) \to 1$. This is because under the covariate shift the training data $\{Z_1, ..., Z_n\}$ and test point Z_{n+1} are not exchangeable (they are weighted exchangeable, i.e., $wZ \stackrel{d}{=} w^i Z^i$), meaning that the unweighted data distributions Z and Z^i may have arbitrarily large total variation distance. The result would then be the trivial coverage guarantee (i.e., only guaranteeing coverage probability ≥ 0).

A.4 Supplemnetary background on influence functions

In this work we implement the algorithm proposed by Giordano et al. [2019a] to compute higher-order influence functions (IFs), so we refer Giordano et al. [2019a] for more comprehensive details and theory. However, in this supplementary section we provide additional details on basic IFs theory and our use of IFs for the convenience of the interested reader.

For a weight vector variable $\omega \in \mathbb{R}^n$ and a fixed instance of the variable $\omega = \tilde{\omega}$ representing a specific reweighting of the data, let us denote $\widehat{\mu}_{\tilde{\omega}}$ as the refitted model and $\hat{\theta}(\tilde{\omega})$ as the refitted model parameters that would be obtained by retraining the model with data weights $\tilde{\omega}$. With our notation in this section we maintain some similarity to the notation in Giordano et al. [2019a], but we use the Greek character ω rather than w to disambiguate the IF data weights ω from the likelihood-ratio weights w introduced in Section 2.3. For the leave-one-out weight vectors that are of primary interest for approximating the jackknife+ and related methods with influence functions, for ease of notation we say that $\tilde{\omega} = -i$ denotes the all ones vector except with zero in the i-th component so that $\widehat{\mu}_{-i}$ still denotes the leave-one-out retrained model, and we denote the corresponding leave-one-out parameters as $\hat{\theta}_{-i} = \hat{\theta}(-i)$.

For any specific weights $\tilde{\omega}$, influence functions assume that $\hat{\theta}(\tilde{\omega})$ is a local minimum of the objective function, and thus that $\hat{\theta}(\tilde{\omega})$ is the solution to the following system of equations, where G is the gradient of the objective function with respect to the model parameters:

$$\hat{\theta}(\tilde{\omega}) := \theta \text{ such that } G(\theta, \tilde{\omega}) := \frac{1}{n}\Big(g_0(\theta) + \sum_{i=1}^{n} \tilde{\omega}_i g_i(\theta)\Big) = 0, \tag{17}$$

where $g_i(\theta)$ is the gradient of the objective function for datapoint i and $g_0(\theta)$ is a prior or regularization term. For the predictor $\widehat{\mu} = \widehat{\mu}_{1_n}$ trained on the full, original dataset, we have $\tilde{\omega} = 1_n$ and can thus denote the model parameters for $\widehat{\mu}$ as $\hat{\theta} = \hat{\theta}(1_n)$. For a resampling-based uncertainty quantification method like the jackknife+ (or bootstrap, cross validation, or other jackknife methods), retraining the model for each new reweighting of the training data can sometimes be computationally burdensome or prohibitive. In these cases, we can instead estimate $\hat{\theta}(\omega)$ using influence functions to compute a Taylor series expansion in ω centered at 1_n (or more specifically a Von Mises expansion, see Fernholz [2012]). A first-order influence function—which we will denote as $\delta^1_\omega \hat{\theta}(1_n)$ for consistency with notation in Giordano et al. [2019a]—refers to the first-order directional derivative of the parameters $\hat{\theta}(\omega)$ with respect to the weights ω:

$$\delta^1_\omega \hat{\theta}(1_n) = \sum_{i=1}^{n} \frac{\partial \hat{\theta}(\omega)}{\partial \omega_i}\bigg|_{\omega = 1_n} \Delta\omega_i, \tag{18}$$

where $\Delta\omega = \omega - 1_n$ is the direction of change in weights relative to the original weights 1_n. The first-order influence function $\delta^1_\omega \hat{\theta}(1_n)$ thus enables a first-order Taylor series approximatinon of $\hat{\theta}(\omega)$, given by

$$\hat{\theta}^{\text{IF-1}}(\omega) := \hat{\theta}(1_n) + \delta^1_\omega \hat{\theta}(1_n). \tag{19}$$

Computing the influence function $\delta^1_\omega \hat{\theta}(1_n)$ requires differentiation through the chain rule because $\hat{\theta}(\omega)$ is only implicitly a function of ω through estimating equation (17). The first-order Taylor series approximation of $\hat{\theta}(\omega)$ given in (19) can then be rewritten as

$$\hat{\theta}^{\text{IF-1}}(\omega) := \hat{\theta}(1_n) - \hat{H}(\hat{\theta})^{-1} G(\hat{\theta})(w - 1_n). \tag{20}$$

where $\hat{H}(\hat{\theta}) = \hat{H}(\hat{\theta}(1_n), 1_n)$ and $G(\hat{\theta}) = G(\hat{\theta}(1_n), 1_n)$ are the Hessian and the gradient of the objective function.

Similarly, higher-order Taylor series approximations of $\hat{\theta}(\omega)$ can be obtained using higher order influence functions $\delta^k_\omega \hat{\theta}(1_n)$, where the K-th order Taylor series is given by

$$\hat{\theta}^{\text{IF-}K}(\omega) := \hat{\theta}(1_n) + \sum_{k=1}^{K} \frac{1}{k!} \delta^k_\omega \hat{\theta}(1_n). \tag{21}$$

Computing $\hat{\theta}^{\text{IF-K}}(\omega)$ requires several assumptions. See Giordano et al. [2019a] for a formal list, but informally we assume that $\hat{\theta}(1_n)$ is the solution to $G(\hat{\theta}(1_n), 1_n) = 0$, that $G(\theta, 1_n)$ is $K+1$ times continuously differentiable, that the hessian $H(\hat{\theta})$ is strongly positive definite (meaning that the objective function is strongly convex in the neighborhood of the local solution), and the norm of the derivative $d_\theta^k G(\theta, 1_n)$ has a finite upper bound for $1 \leq k \leq K+1$. In this work, we implement the recursive procedure based on forward-mode automatic differentiation to achieve memory-efficient computation of higher-order directional derivatives Maclaurin et al. [2015] as described in Giordano et al. [2019a].

While Alaa and Van Der Schaar [2020] propose a higher-order IF approximation of the jackknife+, their method assumes exchangeable (e.g., IID) train and test data and offer experiments with only first and second order IF approximations to the jackknife+. Our proposed JAWA sequence extends the IF approximation of the jackknife+ proposed by Alaa and Van Der Schaar [2020] to the setting of covariate shift, and we demonstrate the benefits of this extension on a variety of datasets and orders of influence function approximation.

B Supplementary theoretical results

B.1 Error assessment assuming exchangeable data

While in Section 3.3 of the main paper we present a general approach to repurposing predictive interval-generating methods with validity under covariate shift to the error assessment task, here we present the analogous results under the exchangeable data assumption. The results in this section directly apply to common predictive interval-generating methods including split conformal, jackknife+, and cross-validation+.

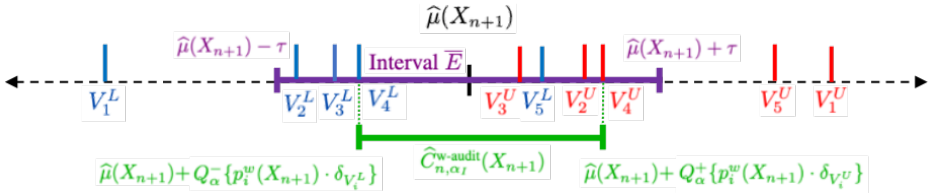


Figure 6: Illustration of terms involved in computing α_E^{audit} when errors are defined by the event $|Y_{n+1} - \widehat{\mu}(X_{n+1})| > \tau$. The interval $I = [\widehat{\mu}(X_{n+1}) - \tau, \widehat{\mu}(X_{n+1}) + \tau]$ is shown in violet, the values $\{V_i^L\}$ in blue, the values $\{V_i^U\}$ in red, and the interval $\widehat{C}_{n,\alpha}^{\text{audit}}(X_{n+1})$ in green. Each vertical line at a location V_i on the real line represents a point mass δ_{V_i} with height $\frac{1}{n+1}$.

The setup is the same as in the main paper Section 3.3, except now we assume exchangeable data and access to a predictive inference method with valid predictive intervals of the form

$$\widehat{C}_{n,\alpha}^{\text{w-audit}}(X_{n+1}) = \{y : \widehat{Q}_{\alpha'}^-\{\tfrac{1}{n+1}\delta_{V_i^L}\} \leq \widehat{S}^-(X_{n+1}, y), \\ \widehat{S}^+(X_{n+1}, y) \leq \widehat{Q}_{1-\alpha'}^+\{\tfrac{1}{n+1}\delta_{V_i^U}\}\} \tag{22}$$

Recall that we use $Q_\alpha^-\{\frac{1}{n+1}\delta_{V_i^L}\}$ to denote the level α quantile of the empirical distribution $\sum_i^n[\frac{1}{n+1}\delta_{V_i^L}] + \delta_{-\infty}$ and $Q_{1-\alpha}^+\{\frac{1}{n+1}\delta_{V_i^U}\}$ to denote the level $1-\alpha$ quantile of the empirical distribution $\sum_i^n[\frac{1}{n+1}\delta_{V_i^U}] + \delta_{\infty}$. Also, for intuition, note that when errors are defined by the event $|Y_{n+1} - \widehat{\mu}(X_{n+1})| > \tau$ (i.e., for score functions $\widehat{S}^{\pm}(x, y) = \pm|y - \widehat{\mu}(x)|$ and tolerance values $\tau^{\pm} = \pm\tau$), then for split conformal prediction, $V_i^L = -|Y_i - \widehat{\mu}(X_i)|$ and $V_i^U = |Y_i - \widehat{\mu}(X_i)|$.

Then, define α_E^{audit} as

$$\alpha_E^{\text{audit}} = \min\left(\left\{\alpha' \ : \ \tau^- \leq \ \widehat{Q}_{\alpha'}^-\{\tfrac{1}{n+1}\delta_{V_i^L}\} \ , \ \widehat{Q}_{1-\alpha'}^+\{\tfrac{1}{n+1}\delta_{V_i^U}\} \leq \tau^+\right\}\right). \tag{23}$$

We can then estimate the probability of $\widehat{\mu}(X_{n+1})$ *not* resulting in an error as in (12) as:

$$\widehat{p}\{Y_{n+1} \in \overline{E}\} = \begin{cases} 1 - \alpha_E^{\text{audit}} & \text{if } \alpha_E^{\text{audit}} \text{ exists} \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

While the target coverage for $\widehat{C}^{\text{audit}}_{n,\alpha_E}(X_{n+1})$ is used in (15), the following theorem gives the worst-case error assessment guarantee for exchangeable data (proof in Appendix C.4).

Theorem 4. *Given a predictive inference method for generating intervals* $\widehat{C}^{audit}_{n,\alpha}(X_{n+1}) = \left[\widehat{\mu}(X_{n+1}) + Q^-_\alpha\{\frac{1}{n+1}\delta_{V_i^L}\},\ \widehat{\mu}(X_{n+1}) + Q^+_{1-\alpha}\{\frac{1}{n+1}\delta_{V_i^U}\}\right]$ *with coverage guarantee* $\mathbb{P}\{Y_{n+1} \in \widehat{C}^{audit}_{n,\alpha}(X_{n+1})\} \geq 1 - c_1\alpha - c_2$ *assuming exchangeable data, where* $c_1, c_2 \in \mathbb{R}$ *are nonnegative constants. Define* $\overline{E}$ *as in* (12) *and* α_E^{audit} *as in* (23)*. Then,*

$$\mathbb{P}\{Y_{n+1} \in \overline{E}\} \leq \begin{cases} 1 - c_1\alpha_E^{audit} - c_2 & \text{if } \alpha_E^{audit} \text{ exists and } \alpha_E^{audit} < \frac{1-c_2}{c_1} \\ 0 & \text{otherwise} \end{cases}. \tag{25}$$

B.2 JAW-E error assessment guarantee

We now state the error assessment guarantee for JAW-E as Corollary 1, which follows directly from Theorem 3 in the covariate shift setting with normalized likelihood ratio weights $p_i^w(X_{n+1})$ when we define errors using the score functions $\widehat{S}^{\pm}(x, y) = \pm|y - \widehat{\mu}(x)|$, and when we let $V_i^L = \widehat{\mu}_{-i}(X_{n+1}) - R_i^{LOO}$ and $V_i^U = \widehat{\mu}_{-i}(X_{n+1}) + R_i^{LOO}$ for all $i \in \{1, ..., n\}$.

Corollary 1. *Given* $\widehat{S}^{\pm}(x, y) = \pm|y - \widehat{\mu}(x)|$ *and some* $\tau^-, \tau^+ \in \mathbb{R}$ *where* $\tau^- < \tau^+$*, define* $\overline{E}$ *as in* (12)*, and define*

$$\alpha_E^{JAW} = \min\Bigg(\Big\{\alpha' \ :\ \tau^- \leq Q^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{\widehat{\mu}_{-i}(X_{n+1}) - R_i^{LOO}}\},$$
$$Q^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{\widehat{\mu}_{-i}(X_{n+1}) + R_i^{LOO}}\} \leq \tau^+\Big\}\Bigg). \tag{26}$$

Then, we have

$$\mathbb{P}\{Y_{n+1} \in \overline{E}\} \geq \begin{cases} 1 - 2\alpha_E^{JAW} & \text{if } \alpha_E^{JAW} \text{ exists and } \alpha_E^{JAW} < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

B.3 JAWA-E error assessment guarantee

Lastly for our theoretical results, we state the error assessment guarantee for JAWA-E as Corollary 2. Whereas Corollary 1 holds for finite samples, Corollary 2 holds in the limit either of the number of samples or in the order of the influence function approximation.

Corollary 2. *Let Assumptions 1 - 4 and either Condition 2 or Condition 4 from Giordano et al. [2019a] hold uniformly for all* n*. Given* $\widehat{S}^{\pm}(x, y) = \pm|y - \widehat{\mu}(x)|$ *and some* $\tau^-, \tau^+ \in \mathbb{R}$ *where* $\tau^- < \tau^+$*, define* $\overline{E}$ *as in* (12)*, and define*

$$\alpha_E^{JAWA\text{-}K} = \min\Bigg(\Big\{\alpha' \ :\ \tau^- \leq\ Q^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{\widehat{\mu}^{IF\text{-}K}_{-i}(X_{n+1}) - R_i^{IF\text{-}K,LOO}}\}\ , \tag{28}$$
$$Q^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{\widehat{\mu}^{IF\text{-}K}_{-i}(X_{n+1}) + R_i^{IF\text{-}K,LOO}}\} \leq \tau^+\Big\}\Bigg). \tag{29}$$

Then, as either $n \to \infty$ *or as* $K \to \infty$*, we have*

$$\mathbb{P}\{Y_{n+1} \in \overline{E}\} \geq \begin{cases} 1 - 2\alpha_E^{JAWA\text{-}K} & \text{if } \alpha_E^{JAWA\text{-}K} \text{ exists and } \alpha_E^{JAWA\text{-}K} < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{30}$$

C Proofs for theoretical results

C.1 Proof of Theorem 1

Proof. We use (a) - (d) to denote four setup steps, and we use 1-3 to denote the main steps in the proof. Our first two initial setup steps (a) and (b) are identical to the corresponding setup steps in the proof for Theorem 1 in Barber et al. [2021]:

(a) First, we suppose the hypothetical case where in addition to the training data $\{(X_1, Y_1), ..., (X_n, Y_n)\}$, we also have access to the test point (X_{n+1}, Y_{n+1}). For each pair of indices $i, j \in \{1, ..., n+1\}$ with $i \neq j$, we define $\tilde{\mu}_{-(i,j)}$ as the regression function fitted on the training and test data except with the points i and j removed. (We follow the notation in Barber et al. [2021] where $\tilde{\mu}$ rather than $\widehat{\mu}$ reminds us that the former is fit on a subset of data $1, ..., n+1$ that may contain the test point $n+1$.) We note that $\tilde{\mu}_{-(i,j)} = \tilde{\mu}_{-(j,i)}$ for any $i \neq j$, and $\tilde{\mu}_{-(i,n+1)} = \widehat{\mu}_{-i}$ for any $i = 1, ..., n$.

(b) We also define the same matrix of residuals in Barber et al. [2021], $R \in \mathbb{R}^{(n+1)\times(n+1)}$, with entries

$$R_{ij} = \begin{cases} +\infty & i = j, \\ |Y_i - \tilde{\mu}_{-(i,j)}(X_i)| & i \neq j \end{cases}$$

such that the off-diagonal entries R_{ij} represent the residual for the ith datapoint where both i and j are not seen by the regression fitting.

At this point we begin to introduce some changes to the proof in Barber et al. [2021]:

(c) We define a weighted version of the comparison matrix in Barber et al. [2021], which we call $A^w \in \mathbb{R}^{(n+1)\times(n+1)}$, with entries $A^w_{ij} = w(X_i)w(X_j) \cdot \mathbb{1}\{R_{ij} > R_{ji}\}$. That is, $\mathbb{1}\{R_{ij} > R_{ji}\}$ is the indicator for the event that, when i and j are excluded from the regression fitting, i has larger residual than j; and A^w_{ij} is this indicator multiplied by $w(X_i)w(X_j)$, the product of the likelihood ratios for points i and j. For any $i, j \in \{1, ..., n+1\}$, note that $A^w_{ij} > 0$ implies $A^w_{ji} = 0$ for any $i, j \in \{1, ..., n+1\}$. Moreover, note that in the absence of covariate shift, $w(X_i) = w(X_j) = 1$ for all $i, j \in \{1, ..., n+1\}$ and the weighted comparison matrix A^w is equivalent to the unweighted comparison matrix A described in Barber et al. [2021].

(d) Next, as in Barber et al. [2021] we are interested in identifying points that have unusually large residuals and are thus hard to predict. Barber et al. [2021] defined such points with unusually large residuals as points i where $\mathbb{1}\{R_{ij} > R_{ji}\}$ for a sufficiently large fraction of other points j. However, in the covariate shift setting, we need to account for the fact that the informativeness of the comparison $\mathbb{1}\{R_{ij} > R_{ji}\}$ depends on the likelihood of j in the test distribution relative to the training distribution: If $w(X_j) > w(X_{j'})$ for some points $j, j' \in \{1, ..., n+1\} \backslash i,\ j \neq j'$, then the comparison $\mathbb{1}\{R_{ij} > R_{ji}\}$ should contain more information about how difficult i is to predict than the comparison $\mathbb{1}\{R_{ij'} > R_{j'i}\}$. In particular, we are interested in identifying points i where $\mathbb{1}\{R_{ij} > R_{ji}\}$ for a sufficiently large *total normalized weight* of other points j. With this motivation, we here define the set of "strange" points $\mathcal{S}(A^w) \subseteq \{1, ..., n+1\}$ in the following two equivalent ways that each serve a different illustrative purpose:

$$\begin{aligned} \mathcal{S}(A^w) &= \left\{ i \in \{1, ..., n+1\} \ : \ w(X_i) > 0, \quad \sum_{j=1}^{n+1} \left(p^w_j(X_{n+1}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\} \right) \geq 1 - \alpha \right\} \\ &= \left\{ i \in \{1, ..., n+1\} \ : \ w(X_i) > 0, \quad \frac{\sum_{j=1}^{n+1} A^w_{ij}}{w(X_i) \sum_{k=1}^{n+1} w(X_k)} \geq 1 - \alpha \right\} \end{aligned}$$

The first definition represents our intuition of $\mathcal{S}(A^w)$ as a set of "strange" points, which we have described (where $\mathbb{1}\{R_{ij} > R_{ji}\}$ for a sufficiently large total normalized weight of other points j). That is, in the first definition it is relatively straightforward to see how $\mathcal{S}(A^w) \subseteq \{1, ..., n+1\}$ is the set of points $i \in \{1, ..., n+1\}$ such that for all the points $j \in \{1, ..., n+1\}, j \neq i$ where $R_{ij} > R_{ji}$, that the sum of the normalized weights $p^w_j(X_{n+1})$ of all such points j is sufficiently large (at least $1 - \alpha$). On the other hand, the second definition represents how the set of strange points can be computed from the weighted comparison matrix A^w and the likelihood ratio weights w. Note that in the second definition, when $w(X_k) = 1$ for all $k = 1, ..., n+1$ it is straightforward to see that $\mathcal{S}(A^w)$ is equivalent to the set of strange points $\mathcal{S}(A)$ in the Barber et al. [2021] proof (to see this, observe that in this case $\sum_{k=1}^{n+1} w(X_k) = n+1$).

The following main steps in our proof take the following structure, similar to as in Barber et al. [2021]:

- Step 1: Establish deterministically that $\sum_{i \in \mathcal{S}(A^w)} p_i^w(X_{n+1}) \leq 2\alpha$. That is, for any comparison matrix A^w, it is impossible to have the total normalized weight of all the strange points exceed 2α.
- Step 2: Using the fact that the datapoints are weighted exchangeable, show that the probability that the test point $n+1$ is strange (i.e., $n+1 \in \mathcal{S}(A^w)$) is thus bounded by 2α.
- Step 3: Lastly, verify that the JAW interval can only fail to cover the test label value Y_{n+1} if $n+1$ is a strange point.

Step 1: Bounding the total normalized weight of the strange points. This proof step follows and generalizes the corresponding proof step for Theorem 1 in Barber et al. [2021], which relies on Landau's theorem for tournaments [Landau, 1953]. For each pair of points i and j where $i \neq j$, let us say that i "wins" its game against point j if $A_{ij}^w > 0$, that is if both i and j have nonzero density in the test distribution and if there is a higher residual on point i than on point j for the regression model $\tilde{\mu}_{-(i,j)}$. We say that i loses its game with j otherwise.

However, whereas Barber et al. [2021] derive a bound on the *number* of strange points from a bound on the *number of pairs* of strange points, we instead derive a bound on the *total normalized weight* of the strange points from a bound on the sum of the *product of normalized weights* for two strange points in a pair. As we will see, this idea generalizes the idea of counting pairs of points to account for continuous weights on the points: If all points have uniform unnormalized weight of 1, then, after adjusting for a normalizing constant in our construction, the product of unnormalized weights of points in a pair is 1 for all pairs and our construction reduces to bounding the number of distinct pairs of strange points.

Observe that, by the definition of a strange point, the points that each strange point $i \in \mathcal{S}(A^w)$ wins against must have total normalized weight greater than or equal to $(1-\alpha)$, and thus the points that each strange point $i \in \mathcal{S}(A^w)$ loses to can only have total normalized weight at most $\alpha - p_i^w(X_{n+1})$ (our definition does not allow i to lose to itself). That is:

$$\begin{array}{c}\text{Total normalized weight}\\ \text{of points that } i \text{ loses to}\end{array} = \sum_{j=1}^{n+1} \left(p_j^w(X_{n+1}) \cdot \mathbb{1}\{R_{ij} \leq R_{ji}\} \right) \leq \alpha - p_i^w(X_{n+1})$$

This inequality will help us obtain an upper bound on the sum of the product of normalized weights between strange points in a pair. To aid with intuition, it may be helpful to think about a correspondance between a product of two weights and the area of a rectangle with side lengths equal to each weight value. Suppose that for each strange point $i \in \mathcal{S}(A^w)$ we construct a rectangle L_i with width equal point i's normalized weight, $p_i^w(X_{n+1})$, and length equal to the largest total normalized weight that the points that i loses to could have, $\alpha - p_i^w(X_{n+1})$. In addition, suppose that we also construct a second rectangle L_i' for each strange point $i \in \mathcal{S}(A^w)$ with width equal to $p_i^w(X_{n+1})$—note that L_i' has the same width as L_i—but with length equal to half the total normalized weight of all of the strange points other than i, that is, $\frac{1}{2}\sum_{j \in \mathcal{S}(A^w)\setminus i} p_j^w(X_{n+1})$.

We now take a moment to describe the meaning of the total area of the set of rectangles $\{L_i\}$ in a way that we will soon make use of: The total area of $\{L_i\}$ is an upper bound on the sum of products of normalized weights for all points in a pair where one point is a strange point and the other point is a point that the strange point loses to. To see this, note that by construction the area of any rectangle L_i is the product of point i's normalized weight (i.e., $p_i^w(X_{n+1})$) with an upper bound on the total normalized weight that the points i loses to could have (i.e., $\alpha - p_i^w(X_{n+1})$). Thus, the area of L_i is by construction an upper bound on the product of point i's normalized weight (i.e., $p_i^w(X_{n+1})$) with the total normalized weight of the points that i *actually* loses to. To state with more precise notation that we will use again later, for each point j that i *actually* loses to, let us construct a rectangle L_{ij} with width $p_i^w(X_{n+1})$ and length $p_j^w(X_{n+1})$. Then, for all these points j, we can arrange the rectangles $\{L_{ij}\}$ so that they are contained within L_i and so that L_{ij} and L_{ij}' have zero overlapping area for all $j \neq j'$: that is, by this construction $\sum_{j:i \text{ loses to } j} \text{Area}(L_{ij}) \leq \text{Area}(L_i)$. So,

it is equivalent to describe the area of L_i as an upper bound on the sum, over all points j that i loses to, of the product of i's normalized weight with j's normalized weight; and thus by extension, the total area of $\{L_i\}$ is as we described earlier.

On the other hand, the total area of the set of rectangles $\{L'_i\}$ is the sum of the product of the normalized weights of two strange points in a pair over all pairs of strange points, where the factor of $\frac{1}{2}$ avoids double counting the pairs of strange points. To see this, note that for every pair of strange points $\{i, j\}$ there is a distinct subrectangle—call it L'_{ij}—that is contained in L'_i, such that L'_{ij} has width $p_i^w(X_{n+1})$ and length $\frac{1}{2}p_j^w(X_{n+1})$ (where we also assume that for any $j \neq j'$, L_{ij} and L'_{ij} overlapping area of zero). Moreover, for this pair of strange points $\{i, j\}$ there is also an analogous subrectangle L'_{ji} with width $p_j^w(X_{n+1})$ and length $\frac{1}{2}p_i^w(X_{n+1})$ contained in L'_j. Thus, the combined area of L'_{ji} and L'_{ij} is $\text{Area}(L'_{ij}) + \text{Area}(L'_{ji}) = p_i^w(X_{n+1}) \cdot p_j^w(X_{n+1})$, and the total area of the set of rectangles $\{L'_i\}$ is as described. (Furthermore, note that when the unnormalized weights are all equal to 1 as in Barber et al. [2021], the area of $\{L'_i\}$—adjusted by a normalization constant—is equivalent to the total number of pairs of strange points $s(s-1)/2$, where $s = |S(A^w)|$ is the number of strange points.)

Now, observe that any pair of two strange points is also a pair of points where one point is strange and the other is a point that the strange point loses to, so the set of pairs of points included in the construction of $\{L'_i\}$ is a subset of the set of pairs of points for which the area of $\{L_i\}$ is the upper bound previously described. To be more precise, let $\{i, j\}$ be a pair of strange points, where (without loss of generality) let us say i loses to j. Then, for the L'_{ij} and L'_{ji} as described before, there exists a distinct L_{ij} such that $\text{Area}(L'_{ij}) + \text{Area}(L'_{ji}) = \text{Area}(L_{ij})$. More generally, we see that the total area of all the subrectangles $\{L'_{ij}\}$ is bounded by the total area of the subrectangles $\{L_{ij}\}$, that is $\sum_{i,j \in \mathcal{S}(A^w),\ i \neq j} \text{Area}(L'_{ij}) = \sum_{i,j \in \mathcal{S}(A^w),\ i \neq j} \text{Area}(L_{ij}) \leq \sum_{i \in \mathcal{S}(A^w),\ i \text{ loses to } j} \text{Area}(L_{ij})$. Moreover, by construction $\sum_{i,j \in \mathcal{S}(A^w),\ i \neq j} \text{Area}(L'_{ij}) = \sum_{i \in \mathcal{S}(A^w)} \text{Area}(L'_i)$ and $\sum_{i \in \mathcal{S}(A^w),\ i \text{ loses to } j} \text{Area}(L_{ij}) \leq \sum_{i \in \mathcal{S}(A^w)} \text{Area}(L_i)$. Therefore, the area of the set of rectangles $\{L'_i\}$ is less than or equal to the area of rectangles $\{L_i\}$, which we can write as follows:

$$\sum_{i \in \mathcal{S}(A^w)} \text{Area}(L'_i) \leq \sum_{i \in \mathcal{S}(A^w)} \text{Area}(L_i)$$
$$\sum_{i \in \mathcal{S}(A^w)} \left(p_i^w(X_{n+1}) \cdot \frac{1}{2} \sum_{j \in \mathcal{S}(A^w) \setminus i} p_j^w(X_{n+1}) \right) \leq \sum_{i \in \mathcal{S}(A^w)} \left(p_i^w(X_{n+1}) \cdot \left(\alpha - p_i^w(X_{n+1}) \right) \right) \tag{C.1.1}$$

Recall that we defined $p_i^w(X_{n+1}) = w(X_i) / \sum_{k=1}^{n+1} w(X_k)\ \forall\, i \in \{1, ..., n+1\}$, so in the uniform weighted case where $w(X_i) = 1\ \forall\, i \in \{1, ..., n+1\}$ then $\sum_{k=1}^{n+1} w(X_k) = n+1$, and multiplying both sides of the inequality above by $(n+1)^2$ yields the analogous inequality in Barber et al. [2021] that bounds the number of pairs of points.

We now proceed to solve for an upper bound on $\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})$, the total normalized weight of strange points:

$$\begin{aligned}
\frac{1}{2}\sum_{i\in\mathcal{S}(A^w)}\Big(p_i^w(X_{n+1})\cdot\sum_{j\in\mathcal{S}(A^w)\setminus i} p_j^w(X_{n+1})\Big) &\leq \sum_{i\in\mathcal{S}(A^w)}\Big(p_i^w(X_{n+1})\cdot\big(\alpha - p_i^w(X_{n+1})\big)\Big)\\
\frac{1}{2}\sum_{i,j\in\mathcal{S}(A^w),\ i\neq j} p_i^w(X_{n+1})p_j^w(X_{n+1}) &\leq \alpha\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1}) - \sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2\\
\frac{1}{2}\Bigg(\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2 + \sum_{i,j\in\mathcal{S}(A^w)} p_i^w(X_{n+1})p_j^w(X_{n+1})\Bigg) &\leq \alpha\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})\\
\frac{1}{2}\Bigg(\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2 + \sum_{i\in\mathcal{S}(A^w)}\Big(p_i^w(X_{n+1})\cdot\sum_{j\in\mathcal{S}(A^w)} p_j^w(X_{n+1})\Big)\Bigg) &\leq \alpha\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})\\
\frac{1}{2}\Bigg(\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2 + \Big(\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})\Big)^2\Bigg) &\leq \alpha\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})\\
\frac{\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2}{\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})} + \Big(\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})\Big) &\leq 2\alpha\\
\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1}) &\leq 2\alpha - \frac{\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2}{\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})}
\end{aligned}$$

where because $0 \leq p_i^w(X_{n+1}) \leq 1\ \forall\, i = 1, ..., n{+}1$ and $p_i^w(X_{n+1}) > 0$ for some $i \in \{1, ..., n{+}1\}$, we have $0 \leq p_i^w(X_{n+1})^2 \leq p_i^w(X_{n+1})\ \forall\, i = 1, ..., n+1$ and thus $0 \leq \frac{\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})^2}{\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1})} \leq 1$, and we have

$$\sum_{i\in\mathcal{S}(A^w)} p_i^w(X_{n+1}) \leq 2\alpha \tag{31}$$

as desired.

Step 2: Weighted exchangeability of the datapoints. We now leverage the weighted exchangeability of the data to show that, since the total weight of the strange points is at most 2α, that a test point has at most 2α probability of being strange.

Note that the datapoints $(X_1, Y_1), ..., (X_{n+1}, Y_{n+1})$ are weighted exchangeable with a regression fitting algorithm $\mathcal{A}$ that is invariant to the ordering of the data. As a result, $\mathbb{1}\{R_{ij} > R_{ji}\}$ are weighted exchangeable random variables such that $w(X_i)w(X_j)\mathbb{1}\{R_{ij} > R_{ji}\}$ can be treated exchangeably, and thus it follows that $A^w \overset{\mathrm{d}}{=} \Pi A^w \Pi^T$ for any $(n+1)\times(n+1)$ permutation matrix Π, where $\overset{\mathrm{d}}{=}$ denotes equality in disribution.

Specifically, for any index $j \in \{1, ..., n+1\}$, suppose take Π to be the permutation matrix with $\Pi_{j,n+1} = 1$ (a permutation mapping $n+1$ to j). Then, deterministically

$$n+1 \in \mathcal{S}(A^w) \iff j \in \mathcal{S}(\Pi A^w \Pi^T)$$

so thus

$$\mathbb{P}\{n+1 \in \mathcal{S}(A^w)\} = \mathbb{P}\{j \in \mathcal{S}(\Pi A^w \Pi^T)\} = \mathbb{P}\{j \in \mathcal{S}(A^w)\}$$

for all $j = 1, ..., n+1$. That is, an arbitrary training point j is equally likely to be strange as the test point $n+1$. Multiplying by $\frac{w(X_j)}{\sum_{k=1}^{n+1} w(X_k)}$, we obtain

$$\frac{w(X_j)}{\sum_{k=1}^{n+1} w(X_k)}\mathbb{P}\{n+1 \in \mathcal{S}(A^w)\} = \frac{w(X_j)}{\sum_{k=1}^{n+1} w(X_k)}\mathbb{P}\{j \in \mathcal{S}(A^w)\}$$

And summing over j, we have

$$
\begin{aligned}
\sum_{j=1}^{n+1} \frac{w(X_j)}{\sum_{k=1}^{n+1} w(X_k)} \mathbb{P}\{n+1 \in \mathcal{S}(A^w)\} &= \sum_{j=1}^{n+1} \frac{w(X_j)}{\sum_{k=1}^{n+1} w(X_k)} \mathbb{P}\{j \in \mathcal{S}(A^w)\} \\
\mathbb{P}\{n+1 \in \mathcal{S}(A^w)\} \sum_{j=1}^{n+1} \frac{w(X_j)}{\sum_{k=1}^{n+1} w(X_k)} &= \sum_{j=1}^{n+1} p_j^w(X_{n+1}) \mathbb{P}\{j \in \mathcal{S}(A^w)\} \\
\mathbb{P}\{n+1 \in \mathcal{S}(A^w)\} &= \sum_{j=1}^{n+1} p_j^w(X_{n+1}) \mathbb{P}\{j \in \mathcal{S}(A^w)\} \\
&= \mathbb{E}\Big[\sum_{j \in \mathcal{S}(A^w)} p_j^w(X_{n+1}) \Big] \\
&\leq 2\alpha
\end{aligned}
$$

where the last line follows from Step 1.

Step 3: Connection to JAW: We would now like to connect our strange point result from step 2 to coverage of the JAW prediction interval. Following the approach of Barber et al. [2021], suppose that $Y_{n+1} \notin \widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1})$. Then, either

$$
\begin{aligned}
Y_{n+1} &> Q^+_{1-\alpha}\big\{p_i^w(X_{n+1})\delta_{\widehat{\mu}_{-i}(X_{n+1})+R_i^{LOO}}\big\} \\
&\implies \sum_{i=1}^{n} p_i^w(X_{n+1}) \cdot \mathbb{1}\big\{Y_{n+1} > \widehat{\mu}_{-i}(X_{n+1}) + R_i^{LOO}\big\} \geq 1-\alpha
\end{aligned}
$$

or otherwise

$$
\begin{aligned}
Y_{n+1} &< Q^-_{\alpha}\big\{p_i^w(X_{n+1})\delta_{\widehat{\mu}_{-i}(X_{n+1})+R_i^{LOO}}\big\} \\
&\implies \sum_{i=1}^{n} p_i^w(X_{n+1}) \cdot \mathbb{1}\big\{Y_{n+1} < \widehat{\mu}_{-i}(X_{n+1}) - R_i^{LOO}\big\} \geq 1-\alpha
\end{aligned}
$$

And we can write the union of these two events as

$$
\begin{aligned}
1-\alpha \leq &\sum_{i=1}^{n} p_i^w(X_{n+1}) \cdot \mathbb{1}\big\{Y_{n+1} \notin \widehat{\mu}_{-i}(X_{n+1}) \pm R_i^{LOO}\big\} \\
&= \sum_{i=1}^{n} p_i^w(X_{n+1}) \cdot \mathbb{1}\big\{\big|Y_i - \widehat{\mu}_{-i}(X_i)\big| < \big|Y_{n+1} - \widehat{\mu}_{-i}(X_{n+1})\big|\big\} \\
&= \sum_{i=1}^{n+1} p_i^w(X_{n+1}) \cdot \mathbb{1}\big\{R_{i,n+1} < R_{n+1,i}\big\}
\end{aligned}
$$

from which we see that $n+1 \in \mathcal{S}(A^w)$—that is, $n+1$ is a strange point. This result together with the result from Step 2 gives us

$$
\begin{aligned}
&\mathbb{P}\big\{Y_{n+1} \notin \widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1})\big\} \leq \mathbb{P}\big\{n+1 \in \mathcal{S}(A^w)\big\} \leq 2\alpha \\
\therefore\ &\mathbb{P}\big\{Y_{n+1} \in \widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1})\big\} \geq 1-2\alpha
\end{aligned}
$$

□

C.2 Proof of Theorem 2

Proof. First, assume that Assumptions 1 - 4 and Condition 2 from Giordano et al. [2019a] hold uniformly for all n (where n is the number of training points). Then, Proposition 1 from Giordano et al. [2019a] establishes that

$$
\max_{i \in [n]} \Big|\Big|\hat{\theta}^{\text{IF-}K}_{-i} - \hat{\theta}_{-i}\Big|\Big|_2 = O_p(N^{-\frac{1}{2}(K+1)}) \tag{32}
$$

So, for fixed K:

$$\lim_{N\to\infty} \max_{i\in[n]} \left|\left|\hat{\theta}_{-i}^{\text{IF-}K} - \hat{\theta}_{-i}\right|\right|_2 = O_p(N^{-\frac{1}{2}(K+1)}) = 0 \tag{33}$$

Or, for fixed N:

$$\lim_{K\to\infty} \max_{i\in[n]} \left|\left|\hat{\theta}_{-i}^{\text{IF-}K} - \hat{\theta}_{-i}\right|\right|_2 = O_p(N^{-\frac{1}{2}(K+1)}) = 0 \tag{34}$$

Thus, $\hat{\theta}_{-i}^{\text{IF-}K} \to \hat{\theta}_{-i}$ as either $N \to \infty$ or $K \to \infty$. This implies that $\widehat{\mu}_{-i}^{\text{IF-}K} \to \widehat{\mu}_{-i}$ as either $N \to \infty$ or $K \to \infty$ because the model $\widehat{\mu}_{-i}$ is fully determined by its parameters $\hat{\theta}_{-i}$. Therefore, $\widehat{C}_{n,\alpha}^{\text{JAWA-}K}(X_{n+1}) \to \widehat{C}_{n,\alpha}^{\text{JAW}}(X_{n+1})$ in the limit of N or K, and thus by Theorem 1, $\mathbb{P}\{Y_{n+1} \in \widehat{C}_{n,\alpha}^{\text{JAWA-}K}(X_{n+1})\} \geq 1 - 2\alpha$ as $N \to \infty$ or $K \to \infty$.

Now, separately assume that Assumptions 1 - 4 and Condition 4 from Giordano et al. [2019a] hold uniformly for all n. Then, Proposition 3 from Giordano et al. [2019a] gives that

$$\max_{i\in[n]} \left|\left|\hat{\theta}_{-i}^{\text{IF-}K} - \hat{\theta}_{-i}\right|\right|_2 = O(N^{-(K+1)}) \tag{35}$$

The rest follows from a similar argument as when we assumed Condition 2. □

C.3 Proof of Theorem 3

Proof. Recall that we define $\overline{E}$ as

$$\overline{E} = \left\{y \in \mathbb{R} : \tau^- \leq \widehat{S}^-(X_{n+1}, y),\ \widehat{S}^+(X_{n+1}, y) \leq \tau^+\right\}, \tag{36}$$

that we assume access to a predictive inference method with prediction sets given by

$$\begin{aligned}\widehat{C}_{n,\alpha}^{\text{w-audit}}(X_{n+1}) = \big\{y : \widehat{Q}^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L}\} \leq \widehat{S}^-(X_{n+1}, y),\\ \widehat{S}^+(X_{n+1}, y) \leq \widehat{Q}^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L}\}\big\},\end{aligned} \tag{37}$$

and moreover that we define $\alpha_E^{\text{w-audit}}$ as

$$\alpha_E^{\text{w-audit}} = \min\left(\left\{\alpha' \,:\, \tau^- \leq \widehat{Q}^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L}\}\,,\ \widehat{Q}^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^U}\} \leq \tau^+\right\}\right). \tag{38}$$

Then, if $\alpha_E^{\text{w-audit}}$ exists and $\alpha_E < \frac{1-c_2}{c_1}$, then by construction we can combine (38) with the definition of the prediction set $\widehat{C}_{n,\alpha_E}^{\text{w-audit}}(X_{n+1})$ to obtain

$$\begin{aligned}\widehat{C}_{n,\alpha_E}^{\text{w-audit}}(X_{n+1}) = \big\{y : \tau^- \leq \widehat{Q}^-_{\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L}\} \leq \widehat{S}^-(X_{n+1}, y),\\ \widehat{S}^+(X_{n+1}, y) \leq \widehat{Q}^+_{1-\alpha'}\{p_i^w(X_{n+1})\delta_{V_i^L} \leq \tau^+\}\big\},\end{aligned} \tag{39}$$

which shows that $\widehat{C}_{n,\alpha_E}^{\text{w-audit}}(X_{n+1}) \subseteq \overline{E}$. Thus, $\mathbb{P}\{Y_{n+1} \in \overline{E}\} \geq \mathbb{P}\{Y_{n+1} \in \widehat{C}_{n,\alpha_E}^{\text{w-audit}}(X_{n+1})\}$, and by the coverage guarantee for $\widehat{C}_{n,\alpha_E}^{\text{audit}}(X_{n+1})$ it follows that

$$\mathbb{P}\{Y_{n+1} \in \overline{E}\} \geq \mathbb{P}\{Y_{n+1} \in \widehat{C}_{n,\alpha_E}^{\text{w-audit}}(X_{n+1})\} \geq 1 - c_1\alpha_E^{\text{w-audit}} - c_2. \tag{40}$$

Otherwise, $\alpha_E^{\text{w-audit}}$ does not exist or $\alpha_E \geq \frac{1-c_2}{c_1} \implies 1 - c_1\alpha_E^{\text{audit}} - c_2 \leq 0$. Neither of these cases yield a nontrivial (positive) lower bound for $\mathbb{P}\{Y_{n+1} \in \overline{E}\}$, so for these cases

$$\mathbb{P}\{Y_{n+1} \in \overline{E}\} \geq \mathbb{P}\{Y_{n+1} \in \widehat{C}_{n,\alpha_E}^{\text{audit}}(X_{n+1})\} \geq 0. \tag{41}$$

□

C.4 Proof of Theorem 4

Proof. The proof proceeds similarly as the proof for Theorem 3 in Appendix C.3, except replacing the data-dependent weights $p_i^w(X_{n+1})$ with the uniform weights $\frac{1}{n+1}$.

□

D Additional experimental details and analysis

D.1 Creation of covariate shift

To induce covariate shift, test points were sampled from the set of points not used for training with exponential tilting weights such that the total number of test points was equal to half the number of points not used for training. For the relatively lower dimensional airfoil and wine datasets, the weights took the form $w(x) = \exp(x^T\beta)$, while for the relatively higher dimensional datasets the weights took the form $w(x) = \exp(x_{\text{PCA}}^T\beta)$ where x_{PCA} is some PCA-based representation of the covariates data x.

Figure 7 shows the distribution first and last features in the airfoil dataset before and after the exponential tilting is applied to induce covariate shift with parameter $\beta = (-1, 0, 0, 0, 1)$. In our main experiments, exponential tilting parameters were selected for each dataset so that the associated covariate shift would result in a similar loss in how informative the training set is regarding the biased test set, as assess by reduced effective sample size.

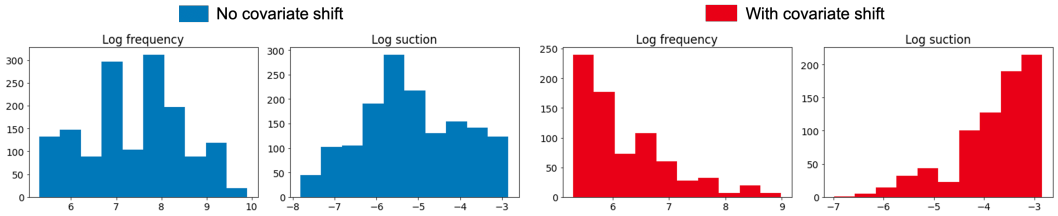


Figure 7: Distribution log frequency and log suction features of airfoil dataset before and after exponential tilting.

Specifically, for a training set size of 200 points for each dataset, the exponential tilting parameters were selected and tuned so that the estimated effective sample size of the training data was reduced to approximately 50, averaged across 1000 random train-test splits. For training data $X_1, ..., X_n$ and likelihood ratio w, the effective sample size was estimated using the following commonly-used heuristic $\widehat{n} = [\sum_{i=1}^n |w(X_i)|]^2 / \sum_{i=1}^n |w(X_i)|^2$ [Gretton et al., 2009, Reddi et al., 2015, Tibshirani et al., 2019].

The specific selections of β that resulted in approximately $\widehat{n} = 50$ for each dataset are as follows. For the airfoil dataset, unless otherwise specified the tilting parameter was $\beta_{\text{airfoil}} = (-0.85, 0, 0, 0, 0.85)$, which induced covariate shift such that points with low values of the first feature and high values of the last feature were more likely to appear in the test distribution (see Figure 7). The wine dataset was similarly tilted using the first and last components, with a tilting parameter of $\beta_{\text{wine}} = (-0.53, 0, ..., 0, 0.53)$. The wave dataset is composed of 48 total features, of which the first 32 features are latitude and longitude values, and where the remaining 16 features are absorbed power values. Accordingly, the first principal component of only the 32 location features was used for tilting along with the first principal component of only the 16 absorbed power values, with a tilting parameter of $\beta_{\text{wave}} = (-0.0000925, 0.0000925)$ unless otherwise specified. For the superconductivity dataset, only the first principal component of all of the data was used for tilting, with tilting parameter $\beta_{\text{superconduct}} = 0.00062$. Lastly, for the communities and crime dataset, the first two principal components of the whole dataset were used with tilting parameter $\beta_{\text{communities}} = (-0.825, 0.825)$.

D.2 Models

For all experiments with JAW and its baselines we used two different regression predictors $\widehat{\mu}$: random forests (scikit-learn RandomForestRegressor), and neural networks (scikit-learn MLPRegressor with LBFGS optimizer, logistic activation, and default parameters otherwise).

For all experiments comparing the coverage and interval width of JAWA to other influence function approximated baselines, we used a neural network predictor with one hidden layer consisting of 25 hidden units. Covariate and label data were centered and scaled. The neural network was trained for 2000 epochs with batch sizes of 50 and a learning rate of 0.0001, which generally resulted in convergence. The objective function for the neural network in JAWA is the negative log

likelihood with a Gaussian prior or L2 regularization term. The L2 regularization was added to satisfy assumptions for computing IFs described in Giordano et al. [2019a] and due to empirical findings of first-order IFs for neural networks requiring regularization for reliable results [Basu et al., 2020]. The L2 regularization λ parameter was tuned using a grid search prior to all experiments using a "tuning" validation set of 200 samples that were excluded from both the training and test sets in the experiments (see Appendix D.8 for more details regarding the L2 regularization tuning).

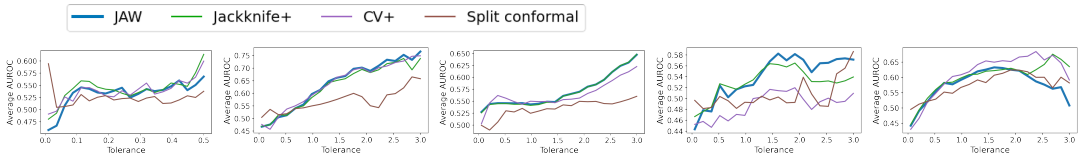
D.3 Comparison of coverage variance for JAW and weighted split

Table 3: Coverage variance for JAW and weighted split conformal prediction, averaged across 1000 experimental replicates (i.e., statistics are the variance of all of the 1000 mean coverage statistics, one for each experiment). Lower coverage variances indicate more reliable coverage. The coverage variance for JAW is lower than that of weighted split conformal prediction in all datasets and predictor conditions due to JAW avoiding the sample splitting required by weighted split.

Dataset	**Airfoil**		**Wine**		**Wave**		**Superconduct**		**Communities**	
Method	**NN**	**RF**	**NN**	**RF**	**NN**	**RF**	**NN**	**RF**	**NN**	**RF**
Weighted split	0.0022	0.0023	0.0019	0.0017	0.0030	0.0029	0.0040	0.0035	0.00194	0.0021
JAW	0.0010	0.0019	0.0013	0.0015	0.0005	0.0014	0.0021	0.0030	0.00189	0.0014

D.4 Additional AUC results

Due to space constraints, in the main paper we only report error assessment AUC results for the random forest predictor condition. For completeness, here we now present error assessment results for the neural network predictor, which are similar:



(a) Airfoil (b) Wine (c) Wave (d) Superconduct (e) Communities

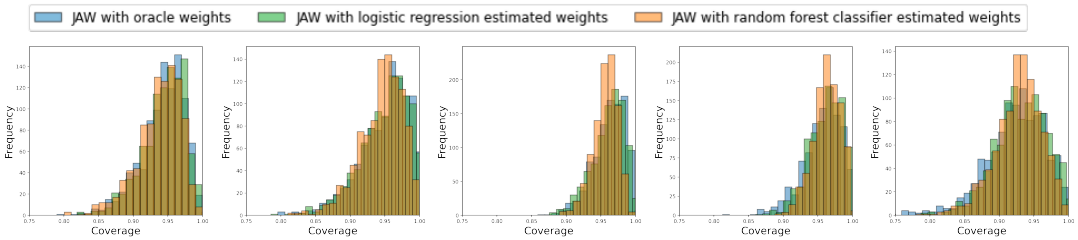
Figure 8: AUROC values for different tolerance levels τ across the three datasets for the random forest predictor, averaged across 5 experiment replicates, with random forest $\widehat{\mu}$. See Appendix D.4 for additional experiments with neural network predictor.

D.5 JAW-E: JAW with estimated weights

JAW assumes access to oracle likelihood ratio weights, but that in practice this information is often not available. In such cases, the likelihood ratios can be estimated through an approach such as probabilistic classification, moment matching, or minimization of ϕ-divergences (for a review of likelihood ratio estimation approaches see Sugiyama et al. [2012]). We refer to JAW with **E**stimated likelihood ratio weights as JAW-E to acknowledge that the method's coverage performance will depend on the quality of the likelihood ratio estimates.

The following experiments compare coverage histograms of JAW with oracle likelihood-ratio weights those of JAW-E with weights estimated from probabilstic classification. We follow the approach used in Tibshirani et al. [2019] for estimating the likelihood-ratio weights using logistic regression and random forest classifiers. Specifically, for training covariate data $X_1, ..., X_n$ and test covariate data $X_{n+1}, ..., X_{n+m}$ where $C_i = 0$ for $i = 1, ..., n$ and $C_i = 1$ for $i = n+1, ..., n+m$, the conditional odds ratio $\mathbb{P}(C = 1|X = x)/\mathbb{P}(C = 0|X = x)$ can be used as an equivalent substitute to the likelihood ratio weight function $w(x)$ due to the normalization of the weights for use in JAW. Thus, for an estimate $\widehat{p}(x) \approx \mathbb{P}(C = 1|X = x)$ obtained from a classifier such as logistic regression or random forest, then we can use the following estimated weight function in place of likelihood-ratio weights:

$$\widehat{w}(x) = \frac{\widehat{p}(x)}{1 - \widehat{p}(x)}. \tag{42}$$



(a) Airfoil (b) Wine (c) Wave (d) Superconduct (e) Communities

Figure 9: Comparison of JAW coverage under covariate shift with oracle versus JAW-E coverage with estimated likelihood ratio weights for neural network predictor across all datasets. Blue is oracle weights, green is weights estimated with logistic regression, and orange is weights estimated with random forest classifier. Histograms represent 1000 experimental replicates.

Figure 9 illustrates the coverage performance of JAW-E, or JAW with estimated likelihood ratio weights, compared to JAW for weights estimated using both logistic regression and random forest classifiers as described in Section D.2. Results are for both neural network and random forest regression predictors across all five UCI datasets. We observe that the coverage histograms for JAW-E with both weight estimation methods are largely directly overlapping with the coverage histogram for JAW with oracle weights. These results demonstrate the applicability of JAW-E for predictive inference under covariate shift when the true likelihood ratio is not known but can be estimated from the data.

D.6 Ablation studies on shift magnitudes

We demonstrate the effect of different magnitudes of covariate shift by comparing the coverage performance of JAW and the jackknife+ on the airfoil dataset with different magnitudes of the exponential tilting bias parameter β. Informed by these experiments depicted in Figure 10—where JAW's mean coverage remains consistent but the variance in coverage increases with increased covariate shift magnitude—we performed additional experiments to investigate the potential cause of JAW's increased variance. Specifically, we compare histograms of JAW's coverage at a fixed covariate shift magnitude to that of jackknife+ without covariate shift but with reduced "effective sample size", which is known to be reduced by likelihood ratio weighting. Tibshirani et al. [2019] made a similar comparison between weighted split conformal prediction under covariate shift and standard split conformal prediction with reduced effective sample size, and we use the same heuristic for effective sample size estimation [Gretton et al., 2009, Reddi et al., 2015] (which we also used for selecting exponential tilting parameter values for each dataset in Figure 3):

$$\widehat{n} = \frac{[\sum_{i=1}^{n} |w(X_i)|]^2}{\sum_{i=1}^{n} |w(X_i)|^2} = \frac{||w(X_{1:n})||_1^2}{||w(X_{1:n})||_2^2}.$$

Effect of different magnitudes of covariate shift As shown in Figure 10, the extent of covariate shift can be controlled by modifying a parameter in the exponential tilting weights so that weights are are more or less drastic. When the bias parameter is set to 0 this corresponds to no bias or IID train and test data. We can see that JAW is robust to different amounts of covariate shift, generating high coverage even under high level of shift.

Reduced effective sample size accounts for JAW increase in coverage variance under shift While JAW's mean coverage remains relatively consistent under different magnitudes of covariate shift as seen in Figure 10, we also observe that the variance in coverage is higher for higher levels of shift. We hypothesized that this increase in variance is due to the high variance issue associated with important weighting methods that is well known [Reddi et al., 2015, Li et al., 2020] in the literature.

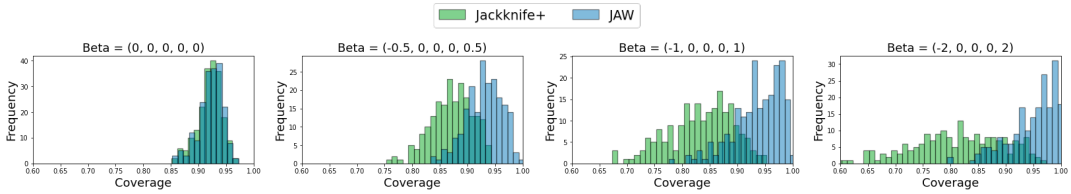


Figure 10: JAW performance compared to jackknife+ on the airfoil dataset with random forest $\widehat{\mu}$ function, under increasing magnitude of covariate shift (different β values), with 200 replicates.

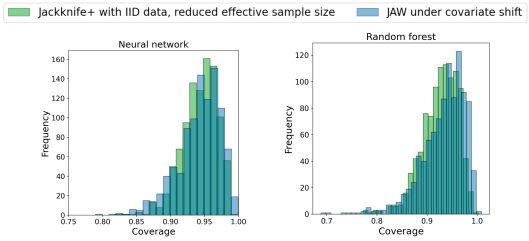


Figure 11: Comparison of JAW coverage histogram under covariate shift (blue) to jackknife+ coverage histogram (green) with no covariate shift but reduced effective sample size corresponding to the magnitude of covariate shift that JAW is evaluated on. Experiments are for both neural network (left) and random forest (right) predictors on the airfoil dataset, with 1000 experimental replicates. The largely overlapping histograms suggests that the increase in JAW coverage variance observed in Figure 10 is largely due to the decrease in effective sample size inherent to likelihood ratio weighting.

We evaluate this hypothesis with effective sample size experiments reported in Figure 11 that compare a histogram of JAW's coverage under covariate shift with the coverage of jackknife+ with IID data but reduced effective sample size corresponding to the magnitude of covariate shift that JAW is evaluated on (see Appendices D.1 and D.6 for details). In Figure 11 we see that the coverage histogram for JAW under covariate shift is nearly directly overlapping with the histogram for jackknife+ coverage with no shift but reduced effective sample size. This result suggests that the reduction of effective sample size due to likelihood ratio weighting is largely if not entirely responsible for the increase in JAW coverage variance for increased shift magnitudes. We leave the variance reduction of our work to the future work.

D.7 Empirical runtime of JAWA compared to JAW

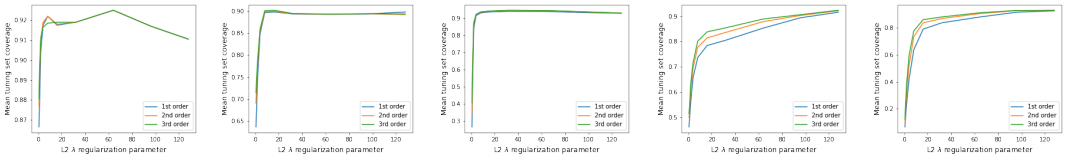
Whereas JAW requires retraining n leave-one-out models, JAWA does not require any retraining, and thus generally enjoys significantly faster runtime than JAW. In Table 4 we report the empirical runtime of JAW compared to JAWA for different orders of JAWA's influence function approximation. In these experiments, JAWA is orders of magnitude faster than JAWA regardless of whether the influence function approximation is first, second, or third order (though of course the specific runtime statistics depend on the model architecture, optimization scheme, or dataset). JAWA's runtime does not increase substantially (relative to JAW's runtime) with increased influence function orders for $K \in \{1, 2, 3\}$.

Table 4: Example empirical comparison between the runtime for JAW and JAWA-K for different influence function approximation orders $K \in \{1, 2, 3\}$ for the neural network predictor used in the JAWA experiments (see Appendix D.2), rounded up to the nearest second. This runtime experiment was performed on an 8-core personal computer with 32 GB of memory.

Method	Airfoil	Wine	Wave	Superconduct	Communities
JAW	58 min, 39 s	59 min, 18 s	1 hr, 24 min, 24 s	1 hr, 26 min, 53 s	1 hr, 25 min, 42 s
JAWA-1	1 s	2 s	4 s	7 s	8 s
JAWA-2	3 s	4 s	6 s	11 s	14 s
JAWA-3	11 s	12 s	16 s	21 s	23 s

D.8 L2 regularization for JAWA experiments

For the experiments involving JAWA and its baselines, the following L2 regularization tuning procedure was used for the neural network described in the second paragraph of D.2. The grid search evaluated the coverage of the first-order influence function approximation of the jackknife+ at different values of the regularization tuning parameter $\lambda \in \{0.5, 1, 2, 4, 8, 16, 32, 64, 96, 128\}$ for 10 train-test splits among all data for a dataset aside from the holdout tuning set. The smallest value of λ in the grid search for which the coverage of the first-order influence function approximation of the jackknife+ exceeded 0.875 was used. The coverage calibration threshold of 0.875 was selected because the change in coverage due to increased λ appeared to plateau just above or below the target coverage rate of 0.9 for each dataset, so setting the threshold slightly below 0.9 can help avoid over-regularizing. See Angelopoulos et al. [2020] for a discussion of calibrating uncertainty estimation in conformal prediction. This grid search procedure identified a separate λ regularization parameter for each dataset: $\lambda_{\text{air}} = 1, \lambda_{\text{win}} = 8, \lambda_{\text{wav}} = 4, \lambda_{\text{sup}} = 96, \lambda_{\text{com}} = 64$. Additionally, we also added a dampening term to the Hessian (for IFs computation) as in Koh and Liang [2017] so that the smallest eigenvalue of the Hessian was at least 0.5.



(a) Airfoil (b) Wine (c) Wave (d) Superconduct (e) Communities

Figure 12: Grid search plots for tuning the λ L2 regularization parameter for influence function coverage experiments. All experiments are done with 1st, 2nd, and 3rd order influence function approximations of the jackkinfe+ (denoted in blue, orange, and green lines in the figure). The y-axis for each plot is the average coverage on the tuning dataset for each L2 regularization parameter $\lambda \in \{0.5, 1, 2, 4, 8, 16, 32, 64, 96, 128\}$.

D.9 Histogram comparison of jackknife+ and JAW coverage under covariate shift

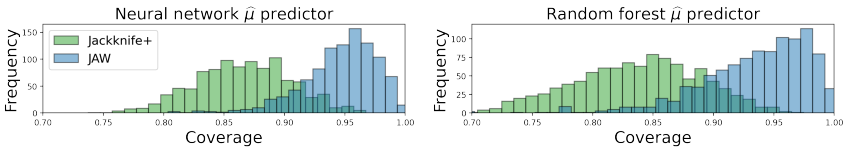


Figure 13: Jackknife+ versus JAW coverage under covariate shift for the airfoil dataset, when $\beta = (-1, 0, 0, 0, 1)$, for 1000 replicates. JAW maintains the high coverage under covariate shift.

D.10 Cases where jackknife+ may not lose coverage

Although JAW maintains significantly higher coverage than jackknife+ in most conditions, our results suggest that there are some cases when jackknife+ may not lose coverage despite lacking a coverage guarantee for covariate shift. For instance, in Figure 3 jackknife+ does lose coverage for the random forest $\widehat{\mu}$ predictor, but it does not appear to lose coverage below the target level with the neural network $\widehat{\mu}$ predictor. Figure 14 allows for a closer look at this observation, with the coverage histograms for JAW and jackknife+ on the superconductivity dataset for both random forest and neural network $\widehat{\mu}$ predictors. In Figure 14 there does appear to be a slight loss of coverage for the jackknife+ with neural network $\widehat{\mu}$ predictor, but not as significant of a loss of coverage as with a random forest $\widehat{\mu}$.

A stronger example were jackknife+ appears to not lose coverage under covariate shift is the wave dataset, where JAW and jackknife+ appear to have similar coverage (Figure 3). Figure 15 examines this observation more closely by comparing JAW and jackknife+ coverage histograms corresponding to increasing levels of covariate shift. For the wave dataset, jackknife+ does not seem to lose coverage regardless of the extent of covariate shift.

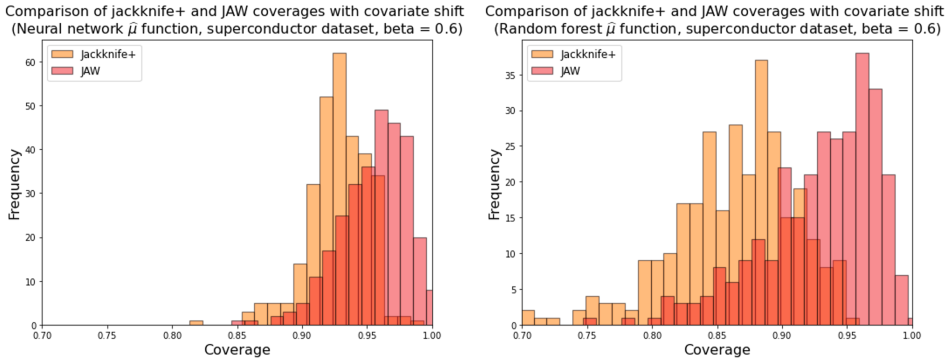


Figure 14: Comparison of the histogram of coverage on Superconductor dataset under covariate shift on the first principal component of the data, with tilting parameter $\beta = 0.6$. JAW still achieves high coverage while jackknife+ loses coverage significantly for the random forest $\widehat{\mu}$ predictor (right). For the neural network $\widehat{\mu}$ predictor (left), jackknife+ does not substantially lose coverage, while JAW has marginally higher coverage, illustrating minimal benefit of JAW over jackknife+ in this case. This is 300 replicates of the experiments.

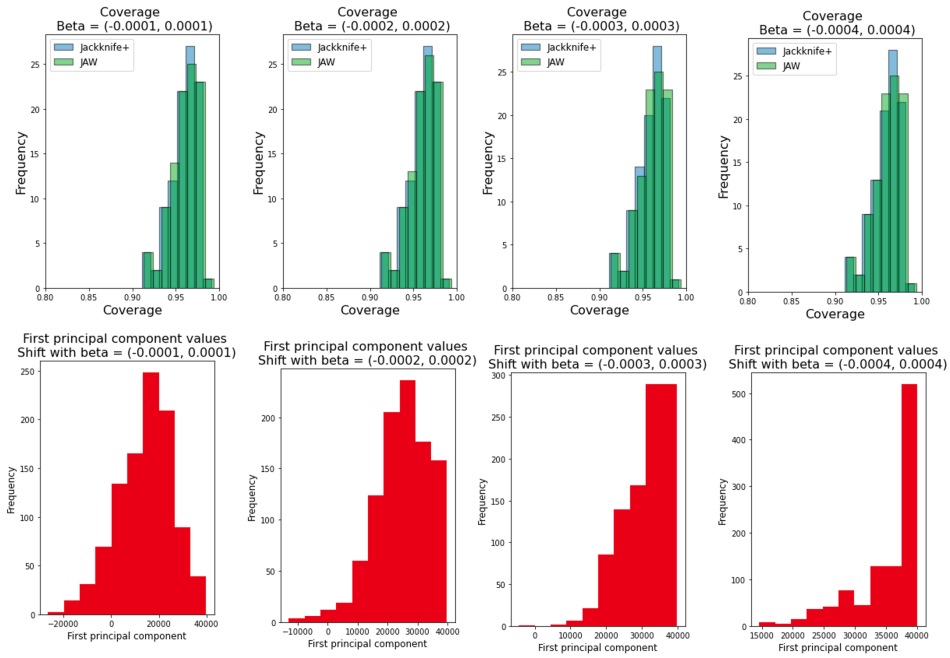


Figure 15: JAW and jackknife+ coverage for different levels of covariate shift levels on the wave energy converters dataset. Each column corresponds to a different level of shift, with increasing shift towards the right. The top row compares JAW (green) and jackknife+ (blue) coverage for a given shift level. The bottom row depicts the first principal component of the data at a given shift level. Neither jackknife+ nor JAW lose coverage at any tested shift level. This is 100 replicates of the experiments.

Though we leave detailed analysis of the conditions that cause jackknife+ to lose coverage or not for future work, we conjecture that jackknife+ loss of coverage may be related covariate shift that makes difficult-to-predict datapoints more likely in the test distribution, and conversely that jackknife+ may not lose coverage when covariate shift does not make difficult-to-predict datapoints more likely in the test distribution. That is, the covariate shift method we use—exponential tilting—causes rare training points to be more common in the test distribution based on the β used for tilting, but our conjecture is that the rarity of a datapoint in the training distribution does not necessarily determine how difficult that point is to predict. If rare but easy-to-predict datapoints are made more common

due to exponential tilting, then this could explain why jackknife+ does not lose coverage in some cases as in Figure 15, though this conjecture requires further investigation.

E Code and computational details

E.1 Code:

https://github.com/drewprinster/jaws.git

E.2 Computational details

All experiments, aside from the runtime comparison described in Appendix D.7 Table 4 were performed on an institutional high performance computing cluster using 10 CPUs with a total of 50GB of memory.