Efficient Distribution-Free Predictive Inference for Standard and Feedback Covariate Shift

Anonymous Authors[1]

Abstract

We propose a collection of efficient wrapper methods for distribution-free predictive inference under both standard and feedback covariate shift (FCS)—that is, covariate shift that allows for feedback-loop dependencies between the training and test distributions, which are present in many decision-making scenarios like experimental design—that are computationally and statistically efficient for arbitrary black-box predictors. Theoretically, our proposed JAW-FCS method achieves a rigorous, finite-sample coverage guarantee under feedback covariate shift. We moreover propose two approaches to relaxing JAW-FCS's computational demands that also apply under standard covariate shift: one building on K-fold cross validation+ (K-WCV+) and a second leveraging K leave-one-out models (JAW-KLOO). Practically, we demonstrate that JAW-FCS and its computational relaxations outperform state-of-the-art baselines on a variety of real-world datasets under standard and feedback covariate shift, including for a protein experimental design scenario.

1. Introduction

Motivation for uncertainty quantification under feedback loops Deployment of machine learning (ML) systems in high-stakes, real-world decision making requires communicating to users whether a given prediction can be trusted, often via reliable and practically efficient quantification of the prediction's uncertainty. The mere use of ML-generated insights to inform future decisions, however, can create feedback loops between the training (e.g., development) and test (e.g., deployment) distributions that can invalidate assumptions of standard uncertainty quantification methods. For example, take an experimental design scenario, where a protein engineer aims to propose a novel protein sequence with high "fitness"—e.g., strong expression of a desired property such as fluorescence—informed by the predictions of an ML model that was trained on a dataset of protein sequences with experimentally-labeled fitness values. This protein design problem requires balancing the exploration of novel sequences that are far from the training data and the exploitation of the sequences the model already can predict a high "fitness" with high confidence. Uncertainty quantification plays a crucial role in navigating this balance. However, by selecting (for downstream experimental design) the novel protein sequences according to the models' predicted "fitness", the engineer induces a dependency between the training and test (designed sequence) distributions, which violates the common ML assumptions of the data being independent and identically distributed (i.i.d.). Similar feedback loops are also common in other decision-making scenarios, including active learning, ML for scientific discovery, and safe exploration in reinforcement learning.

Distribution-free predictive inference for non-iid data In this work we focus on quantifying uncertainty with predictive confidence intervals (or sets, more generally), and hereon we refer to methods for generating such predictive intervals as *predictive inference* methods. Previous work proposes variants of conformal predictions under covariate shift (Tibshirani et al., 2019) which utilize the concept of weighted exchangeability to develop weighted conformal predictors with the same coverage guarantees of i.i.d. data. JAW (Prinster et al., 2022) is further developed to provide an alternative solution under covariate shift based on the jackknife+. These works focus on the standard covariate shift (SCS) when there is no dependency between the testing and the training distributions.

Computationally and statistically efficient predictive inference under standard and feedback covariate shift To be useful in practice, predictive inference methods need to adhere to application-specific resource constraints such as computational budget and scarce data availability. Computational limitations are particularly salient for nonlinear models such as neural networks with expensive training requirements. On the other hand, statistical efficiency refers to the efficient use of available data. However, popular

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Table 1. Summary of key properties for proposed predictive inference methods for standard and feedback covariate shift

"Proposed" or Reference if Baseline	Method Name	Finite-Sample Coverage Guarantee for Standard & Feedback Covariate Shift	General Computational Cost: # Retrained Predictors for n train & m test points	Statistical Efficiency: No Sample Splitting
Fannjiang et al. (2022)	Full-FCS	✓	$(n+1)\cdot\lvert\mathcal{Y}\rvert\cdot m$	✓
Proposed	JAW-FCS	✓	n	✓
Proposed	JAW-KLOO	✓	$K \leq n$	✓
Proposed	WCV+	✓	$K \leq n$	✓
Tibshirani et al. (2019)	Weighted Split	✓	0	✗

predictive inference methods like split conformal prediction split the labeled data into training data and a labeled, holdout "calibration" set for constructing predictive intervals. This splitting is especially detrimental when labeled data are scarce or expensive to collect, such as in many experimental design scenarios. The work of Fannjiang et al. (2022) formalizes a type of feedback-loop data shift they call feedback covariate shift (FCS), and the authors propose an extension of conformal prediction (Vovk et al., 2005) that generates valid predictive intervals under FCS. However, the primary extension of conformal prediction proposed in Fannjiang et al. (2022) is statistically efficient but computationally impractical for nonlinear predictors, while a secondary, data-splitting method described in Fannjiang et al. (2022) is computationally cheap but statistically inefficient.

In this work, we address these gaps by proposing a collection of methods for valid predictive inference under both FCS and standard covariate shift that are both statistically efficient and computationally practical for arbitrary, potentially nonlinear predictor functions. Table 1 summarizes key properties of our methods.

Our contributions can be summarized as follows:

- We propose a statistically efficient and computationally practical approach to distribution-free predictive inference under FCS for arbitrary, black-box predictors. This approach generalizes both the jackknife+ (Barber et al., 2021) and JAW (**ja**ckknife+ **w**eighted) (Prinster et al., 2022) methods to feedback covariate shift (FCS) while achieving the same rigorous, finite-sample coverage guarantee: we call our method as JAW-FCS.

- We propose two computational relaxations of JAW-FCS that also apply to the standard covariate shift (SCS) setting (Prinster et al., 2022). The first approach, K-fold WCV+FCS, generalizes K-fold cross validation+ (CV+) developed in (Barber et al., 2021) to FCS and SCS with a weaker coverage guarantee. The second relaxation approach leverages $K \leq n$ leave-one-out models and maintains the JAW-FCS guarantee.

- Empirically, we demonstrate that JAW-FCS and its computational relaxations outperform state-of-the-art baselines on a variety of real-world datasets under standard and feedback covariate shift, including for a protein experimental design task. In particular, JAW-FCS and K-fold WCV+ maintain target coverage levels under covariate shift with sharper (more informative) predictive intervals than baselines.

2. Background and Related Work

2.1. Predictive Inference

We assume a multiset of training data $Z_{1:n} = \{Z_1, ..., Z_n\} = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ and a test point $Z_{n+1} = (X_{n+1}, Y_{n+1})$ with unknown label Y_{n+1}, where $(X_i, Y_i) \in \mathcal{X} \times \mathcal{Y}$ for all $i \in \{1, ..., n+1\}$ (and for a standard regression setup $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \mathbb{R}$). Moreover, let $\widehat{\mu} = \mathcal{A}(\{(X_1, Y_1), ..., (X_n, Y_n)\})$ denote a black-box predictor of interest, where $\mathcal{A}$ is a model-fitting algorithm. Then, a predictive interval is a function $\widehat{C}_{n,\alpha} : \mathbb{R}^d \rightarrow \{\text{subsets of } \mathbb{R}\}$ that maps a test point X_{n+1} to an interval $\widehat{C}_{n,\alpha}(X_{n+1})$ around the prediction $\widehat{\mu}(X_{n+1})$, for some significance level $\alpha \in (0, 1)$. A predictive interval $\widehat{C}_{n,\alpha}(X_{n+1})$ has *valid coverage* if it is guaranteed to contain the true label Y_{n+1} with high probability, such as satisfying

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

for all $\alpha \in (0, 1)$. A predictive inference method is a *distribution-free* if its coverage validity holds without assumptions on the distribution family. It is important to note that we focus on marginal rather than conditional coverage (see Foygel Barber et al. (2021) for more on this distinction).

2.2. Standard Conformal Prediction

Conformal prediction is an increasingly popular approach to distribution-free predictive inference (Vovk et al., 2005; Shafer & Vovk, 2008; Angelopoulos & Bates, 2021). Traditional conformal prediction methods rely on two key assumptions of *exchangeability*: data exchangeability, that is that the training and test data are all exchangeable (i.i.d. being a special case), and secondly that the model-fitting algorithm $\mathcal{A}$ treats all the data symmetrically (Barber et al., 2022). Conformal methods then use a score function $\widehat{S} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ to quantify the extent to which labeled points "conform" to previous data, thereby enabling the construction of predictive confidence intervals from subsets of $\mathcal{Y}$ whose scores are contained within a carefully constructed quantile on the empirical distribution of score values.

Full and split conformal are the two main types of standard conformal prediction (Vovk et al., 2005; Shafer & Vovk, 2008), and together these methods represent polar opposite ends on a spectrum of tradeoffs between computational versus statistical efficiency in predictive inference. Full conformal prediction avoids sample-splitting and allows all of the labeled data to be used for model training, but its statistical efficiency comes at a heavy computational price of rerunning the model-training algorithm $\mathcal{A}$ once for every possible label value $y \in \mathcal{Y} \subseteq \mathbb{R}$ (or in practice for a fine grid of y). In contrast, split conformal prediction is computationally efficient in that it does not require rerunning $\mathcal{A}$, but at the statistical cost of setting aside a labeled holdout "calibration" dataset.

2.3. Jackknife+ and Cross Validation+

The jackknife+ and cross validation+ (CV+) methods Barber et al. (2021), which are closely related to cross-conformal prediction (Vovk, 2015), offer a range of beneficial, intermediate tradeoffs between the poles of computational and statistical properties that correspond to full and split conformal prediction. The jackknife+ is a modified version of classic leave-one-out or "jackknife" resampling (Miller, 1974; Steinberger & Leeb, 2018; 2016), which requires rerunning the model-training algorithm $\mathcal{A}$ a total of n times, once for each possible leave-one-out model. With the same exchangeability assumptions as in standard conformal prediction, (Barber et al., 2021) prove that the jackknife+ prediction interval satisfies a slightly weaker coverage guarantee than standard conformal methods, namely

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

CV+ offers a computational relaxation of jackknife+ to allow for retraining $K \leq n$ models that each withhold $m = \frac{n}{K}$ datapoints from training, where it is assumed that n is divisible by K (Barber et al., 2021); CV+ can thus be understood as a modification to K-fold cross validation resampling. For each training point $i \in \{1, ..., n\}$, let $k(i) \in \{1, ..., K\}$ denote the index of a cross-validation fold, where the multisets of points in each fold are denoted $\{S_1, ..., S_K\}$. Let $\widehat{\mu}_{-S_k} = \mathcal{A}\big((X_i, Y_i) : i \in \{1, ..., n\} \backslash S_k\big)$ denote the predictor trained with the k-th cross-validation fold S_k removed, and denote the residuals for the model $\widehat{\mu}_{-S_k}$ applied to points in its left-out fold S_k as $R_i^{CV} = \big|\widehat{\mu}_{-S_{k(i)}}(X_i) - Y_i\big|$ for $i : k(i) = k$. Then, the CV+ interval is defined as

$$\widehat{C}^{\text{CV+}}_{n,K,\alpha}(x) =$$

$$\Big[Q_\alpha\Big(\sum_{i=1}^{n} \big[\tfrac{1}{n+1}\delta_{\widehat{\mu}_{-S_{k(i)}}(x) - R_i^{CV}}\big] + \tfrac{1}{n+1}\delta_{-\infty}\Big),$$

$$Q_{1-\alpha}\Big(\sum_{i=1}^{n} \big[\tfrac{1}{n+1}\delta_{\widehat{\mu}_{-S_{k(i)}}(x) + R_i^{CV}}\big] + \tfrac{1}{n+1}\delta_{+\infty}\Big)\Big], \tag{3}$$

where δ_v denotes a point mass at v and $Q_\beta(\cdot)$ denotes the level β empirical quantile function. Barber et al. (2021) provide a finite sample coverage guarantee for the CV+ (under exchangeability assumptions) that depends on K, where the strength of the guarantee approaches that of the jackknife+ in (2) as $K \to n$.

2.4. Standard and Feedback Covariate Shift

Under the standard assumption of covariate shift, which we hereon refer to as standard covariate shift (or SCS), the conditional $Y \mid X$ 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). Crucially, SCS assumes that the test data distribution is independent of the training data:

$$(X_i, Y_i) \overset{\text{i.i.d.}}{\sim} P_X \times P_{Y|X}, i = 1, ..., n$$

$$(X_{n+1}, Y_{n+1}) \overset{\text{i.i.d.}}{\sim} \widetilde{P}_X \times P_{Y|X}, \textit{independently}. \tag{4}$$

Feedback covariate shift (FCS) as described by Fannjiang et al. (2022), however, can be understood as a generalization of SCS where the marginal distribution of the test inputs depend on the realization of the training data $Z_{1:n}$:

$$(X_i, Y_i) \overset{\text{i.i.d.}}{\sim} P_X \times P_{Y|X}, i = 1, ..., n$$

$$(X_{n+1}, Y_{n+1}) \overset{\text{i.i.d.}}{\sim} \widetilde{P}_{X;Z_{1:n}} \times P_{Y|X}. \tag{5}$$

Feedback covariate shift thus describes the example experimental design scenario described in the introduction, where a learned model on the training data of protein sequences, $Z_{1:n}$, influences $\widetilde{P}_{X;Z_{1:n}}$, a distribution over possible designed protein sequences predicted to have high fitness.

2.5. Conformal Prediction for Feedback Covariate Shift

The main result of Fannjiang et al. (2022) extends full conformal prediction to achieve valid coverage of the form (1) under feedback covariate shift. However, for an arbitrary, potentially nonlinear black-box predictor, full conformal prediction for FCS requires the impractical computational demand of retraining the model-training algorithm $\mathcal{A}$ a total of $(n+1) \cdot |\mathcal{Y}|$ times for each unique test point Z_{n+1} (where in regression, often $\mathcal{Y} = \mathbb{R}$ so $\mathcal{Y}$ must be approximated by a fine grid of values). Only in special cases such as linear models can full conformal-FCS's computational demand be reduced to retraining $n + 1$ times per test point.

As an alternative to full conformal for FCS, Fannjiang et al. (2022) also demonstrate that split conformal prediction weighted with data-dependent likelihood ratio weights, proposed in Tibshirani et al. (2019) for SCS, also maintains valid coverage under FCS without requiring model retraining. Weighted split conformal prediction maintains its coverage guarantee under FCS unlike weighted full conformal prediction (also proposed by Tibshirani et al. (2019) for SCS) because due to sample splitting, the test distribution depends on split conformal's training data proper but not on its holdout calibration set, which returns FCS to the setting of SCS. However, the use of weighted split conformal prediction under FCS is statistically inefficient due to its sample splitting requirement, which (as we will demonstrate) results in reduced model performance and overly wide (and thus less informative) predictive intervals relative to our proposed methods.

2.6. JAW: Jackknife+ Under Standard Covariate Shift

The **ja**ckknife+ **w**eighted with data-dependent likelihood ratio weights, or JAW, proposed in Prinster et al. (2022), relaxes the jackknife+'s assumption of data exchangeability to allow for SCS while achieving the same finite-sample coverage guarantee (2). Like jackknife+, JAW under SCS (hereon JAW-SCS) requires retraining n leave-one-out models. Prinster et al. (2022) also propose approximating these leave-one-out models using higher-order influence functions to avoid model retraining, but the resulting JAW **a**pproximation method (JAWA) imposes additional regularity assumptions on the data and predictor, and the coverage guarantee for JAWA under SCS is asymptotic rather than finite-sample like JAW-SCS. In this work, we generalize the JAW-SCS and propose two computational relaxations of JAW under either SCS or FCS.

3. JAW-FCS: Jackknife+ Weighted for FCS

Let training data $Z_{1:n} = \{Z_1, ..., Z_n\}$ and a test point Z_{n+1} be generated from feedback covariate shift (5), and denote $w(x; D) = \mathrm{d}\widetilde{P}_{X;D}(x)/\mathrm{d}P_X(x)$ as a likelihood ratio function for the data that depends on D for $D \subseteq Z_{1:n}$. Then, for each $j \in \{1, ..., n+1\}$, let $Z_{-j} = Z_{1:n} \backslash Z_j$ denote the training data with point j removed (where $Z_{-(n+1)} = Z_{1:n}$), and define the normalized weight function

$$\tilde{w}_{n+1,j}(x) = \frac{w(x; Z_{-j})w(X_j; Z_{-j})}{\sum_{j'=1}^{n+1}\left[w(x; Z_{-j'})w(X_{j'}; Z_{-j'})\right]}. \tag{6}$$

Given X_{n+1} as an argument, $\tilde{w}_{n+1,j}(X_{n+1})$ can be thought of as a weight applied to the training point X_j that is carefully normalized with respect to the other training data and the test point X_{n+1}. To condense notation slightly, for $j = n + 1$ we will also write $\tilde{w}_{n+1,n+1}(x)$ as $\tilde{w}_{(n+1)^2}(x)$.

We then define the predictive interval for JAW under feedback covariate shift, JAW-FCS, as follows:

$$\begin{aligned}
&\widehat{C}^{\text{JAW-FCS}}_{n,\alpha}(x) = \\
&\Big[Q_\alpha\Big(\sum_{j=1}^{n} \tilde{w}_{n+1,j}(x)\delta_{\widehat{\mu}_{-j}(x)-R_j^{LOO}} + \tilde{w}_{(n+1)^2}(x)\delta_{\infty}\Big), \\
&Q_{1-\alpha}\Big(\sum_{j=1}^{n} \tilde{w}_{n+1,j}(x)\delta_{\widehat{\mu}_{-j}(x)+R_j^{LOO}} + \tilde{w}_{(n+1)^2}(x)\delta_{-\infty}\Big)\Big],
\end{aligned} \tag{7}$$

where δ_v denotes a point mass at v and $Q_\beta(\cdot)$ denotes the level β empirical quantile function. The following theorem presents the finite-sample coverage guarantee for the JAW-FCS interval (7), which relaxes the assumption of standard covariate shift in Prinster et al. (2022) to allow for feedback covariate shift. We defer the proof for the theorem to Appendix A.2.

Theorem 3.1. *Suppose data are generated under feedback covariate shift* (5) *and assume* $\widetilde{P}_{X;D}$ *is absolutely continuous with respect to* P_X *for all possible values of* D*. Then, for any miscoverage level,* $\alpha \in (0, 1)$*, the JAW-FCS predictive interval in* (7) *satisfies*

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}^{\textit{JAW-FCS}}_{n,\alpha}(X_{n+1})\} \geq 1 - 2\alpha. \tag{8}$$

Remark 3.2. The JAW-FCS prediction interval is statistically efficient in that it does not require sample splitting to form a holdout dataset, and it is computationally efficient relative to full conformal for FCS (Fannjiang et al., 2022) in general. Whereas full conformal for FCS requires rerunning the model-fitting algorithm $\mathcal{A}$ a total of $(n+1)\cdot|\mathcal{Y}|\cdot m$ times m test points for arbitrary $\widehat{\mu}$, JAW-FCS requires rerunning $\mathcal{A}$ a total of n times for an arbitrary $\widehat{\mu}$ and for any number of unique test points m (Table 1).

4. Further Computational Relaxations

4.1. Relaxation of JAW with K Leave-One-Out Models

While the calculation of the JAW-FCS prediction interval (7) is in general computationally efficient relative to full conformal for FCS (Fannjiang et al., 2022), the retraining requirements of JAW-FCS can be relaxed even further to enable rerunning the model-training algorithm $\mathcal{A}$ only $K \leq n$ times. As our first proposed computational relaxation, we demonstrate that using only $K \leq n$ of the leave-one-out models required by the full JAW-FCS method still achieves the same coverage guarantee, though at the price of wider or more variable predictive intervals. We call this first computational relaxation JAW-KLOO.

The K training points used for leave-one-out retraining in JAW-KLOO can be selected either deterministically (e.g., selecting the K points with the largest normalized weight) or randomly (e.g., via random uniform sampling or sampling with probabilities proportional to normalized weights). In practice, the deterministic JAW-KLOO method pays the price of larger predictive intervals than JAW-FCS, whereas the random JAW-KLOO method's expense largely takes the form of higher coverage variance (corresponding to less reliable coverage) than JAW-FCS. In this work, we focus on a deterministic variant of JAW-KLOO where the K points with largest normalized weight values are selected for the leave-one-out models.

Let $S_{\text{LOO}} \subseteq \{1, ..., n\}$ denote a subset of the training data where $|S_{\text{LOO}}| = K$. Then, we define the JAW-KLOO prediction interval similarly as the JAW-FCS interval, except only using leave-one-out models for points $j : j \in S_{\text{LOO}}$:

$$\tilde{w}^{K\text{LOO}}_{n+1,j}(x) = \frac{w(x; Z_{-j})w(X_j; Z_{-j})}{\sum_{j' \in S_{\text{LOO}}} \left[w(x; Z_{-j'})w(X_{j'}; Z_{-j'})\right]}. \tag{9}$$

where to condense notation slightly we also write $\tilde{w}^{w}_{(n+1)^2}(X_{n+1})$ to denote $\tilde{w}_{n+1,n+1}(X_{n+1})$. Then, we define the JAW-KLOO prediction interval as follows

$$\begin{aligned}&\widehat{C}^{\text{JAW-}K\text{LOO}}_{n,\alpha}(x) = \\ &\Big[Q_\alpha\Big(\sum_{j \in S_{\text{LOO}}} \tilde{w}^{K\text{LOO}}_{n+1,j}(x)\delta_{\widehat{\mu}_{-j}(x) - R^{LOO}_j} + \tilde{w}^{K\text{LOO}}_{(n+1)^2}(x)\delta_{\infty}\Big), \\ &Q_{1-\alpha}\Big(\sum_{j \in S_{\text{LOO}}} \tilde{w}^{K\text{LOO}}_{n+1,j}(x)\delta_{\widehat{\mu}_{-j}(x) + R^{LOO}_j} + \tilde{w}^{K\text{LOO}}_{(n+1)^2}(x)\delta_{-\infty}\Big)\Big]\end{aligned} \tag{10}$$

The JAW-KLOO model then satisfies the same coverage guarantee as the full JAW-FCS model, which we state formally in the following theorem (proof in Appendix A.3).

Theorem 4.1. *Suppose data are generated under feedback covariate shift* (5) *and assume $\widetilde{P}_{X;D}$ is absolutely continuous with respect to P_X for all possible values of D. Then, for any miscoverage level, $\alpha \in (0, 1)$, the JAW-KLOO predictive interval in* (10) *satisfies*

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

4.2. K-fold Weighted Cross-Valdation+

We now propose an alternative computational relaxation of JAW-FCS that relies on a weighted K-fold cross validation resampling procedure with K leave-$\frac{n}{K}$-out models (in contrast to JAW-KLOO that uses K leave-*one*-out models). In particular, this second computational relaxation generalizes the K-fold cross validation+ (CV+) method of Barber et al. (2021) to allow for feedback or standard covariate shift—we call this method WCV+ or K-fold WCV+ when referring to a specific K.

Prior to defining the WCV+FCS predictive interval, we first need to generalize the normalized weights defined in (6) using likelihood ratio functions w that depend on all the training data aside from a specific fold (rather than depending on leave-one-out subsets Z_{-j} as in (6)). For $k(j) \in \{1, ..., K\}$ denoting the index of the cross validation fold for point j, let $Z_{-S_{k(j)}} = Z_{1:n} \backslash S_{k(j)}$. Then, we define

$$\tilde{w}^{CV}_{n+1,j}(x) = \frac{w(x; Z_{-S_{k(j)}})w(X_j; Z_{-S_{k(j)}})}{\sum_{j'=1}^{n+1} \left[w(x; Z_{-S_{k(j')}})w(X_{j'}; Z_{-S_{k(j')}})\right]}. \tag{12}$$

for training data $j \in \{1, ..., n\}$. We can now define the K-fold WCV+FCS (WCV+ for short) predictive interval as follows:

$$\begin{aligned}&\widehat{C}^{\text{WCV+}}_{n,K,\alpha}(x) = \\ &\Big[Q_\alpha\Big(\sum_{j=1}^{n} \tilde{w}^{CV}_{n+1,j}(x)\delta_{\widehat{\mu}_{-S_{k(i)}}(x) - R^{CV}_i} + \tilde{w}^{CV}_{(n+1)^2}(x)\delta_{-\infty}\Big), \\ &Q_{1-\alpha}\Big(\sum_{j=1}^{n} \tilde{w}^{CV}_{n+1,j}(x)\delta_{\widehat{\mu}_{-S_{k(i)}}(x) - R^{CV}_i} + \tilde{w}^{CV}_{(n+1)^2}(x)\delta_{\infty}\Big), \Big],\end{aligned} \tag{13}$$

where the $\tilde{w}^{CV}_{n+1,j}(x)$ are defined in (12), and where $\tilde{w}^{CV}_{(n+1)^2}(x) = \tilde{w}^{CV}_{n+1,n+1}(x)$. We now present the coverage guarantee for WCV+, which is weaker than the JAW-FCS guarantee by a term that that approximately represents the average normalized weight as in (12) across the folds. We defer the proof to Appendix A.4.

Theorem 4.2. *Suppose data are generated under feedback covariate shift* (5) *and assume $\widetilde{P}_{X;D}$ is absolutely continuous with respect to P_X for all possible values of D. Then, for any miscoverage level, $\alpha \in (0, 1)$, the K-fold WCV+ predictive interval in* (13) *satisfies*

$$\begin{aligned}&\mathbb{P}\{Y_{n+1} \in \widehat{C}^{WCV+FCS}_{n,K,\alpha}(X_{n+1})\} \geq \\ &\qquad 1 - 2\alpha - \mathbb{E}_i\Big[\sum_{j \in S_{k(i)}, j \neq i} \tilde{w}^{CV}_{ij}(X_{n+1})\Big].\end{aligned} \tag{14}$$

Remark 4.3. This generalization also extends CV+ to standard covariate shift as a special case, which to the best of our knowledge is also a novel contribution (the first computational relaxation of the JAW-SCS method in Prinster et al. (2022) that avoids sample splitting and maintains a finite-sample guarantee).

5. Experiments

To demonstrate the practical performance of our JAW-FCS method and its computational relaxations in a real-world, feedback covariate shift scenario, we focus on the protein design problem initially presented in the introduction. In this setting, a common ML-assisted protein engineering goal is to design a novel protein sequence that is predicted to have high "fitness" with regard to some desired functional utility (Yang et al., 2019; Sinai & Kelsic, 2020; Wu et al., 2021), such as strong expression of fluorescence. We also evaluate the two computational relaxations in the standard covariate shift settings.

5.1. Protein Design Experiments under FCS

Datasets In protein engineering, the labels for designed sequences are usually unknown, so we follow Fannjiang et al. (2022) by using the workaround offered by combinatorially complete protein datasets (Wu et al., 2019; Wittmann et al., 2021; Poelwijk et al., 2019; Brookes et al., 2022), where each sequence was moreover measured for both a "red" and a "blue" wavelength fluorescence, thus resulting in two combinatorially complete datasets corresponding to the two distinct fitness functions.

Creation of Feedback Covariate Shift In line with Fannjiang et al. (2022) and other biomolecular engineering papers (Biswas et al., 2021; Zhu et al., 2021), design or "test point" protein sequences were sampled from a distribution over sequences from the fluorescent proteins dataset (Poelwijk et al., 2019) with log-likelihood proportional to the regression model's prediction. That is, design sequences were sampled from $\widetilde{P}_{X;Z_{1:n}}(X_{n+1}) \propto \exp(\lambda \cdot \widehat{\mu}(X_{n+1}))$, where the hyperparameter $\lambda \geq 0$ is the "inverse temperature". Thus, this design procedure corresponds to the creation of feedback covariate shift with larger values of λ corresponding to larger shift magnitudes. Artificial measurement noise was also added to the sampling procedure as in Fannjiang et al. (2022) to simulate a real experimental scenario where measuring the same sequence several time often results in different observed measurements. We used the scikit-learn package (Pedregosa et al., 2011) MLPRegressor method (with L-BFGS solver and the logistic activation function) for the neural network predictor, and we used the package's RandomForestRegressor method (with 20 trees and the absolute error criterion) for the random forest predictor.

Oracle Weights Are Known in the Design Problem The design problem is a special case of feedback covariate shift (FCS) where the distribution of the inputs is usually known or can be reliably simulated, which substantially reduces or removes altogether the challenge of likelihood-ratio weight estimation (Fannjiang et al., 2022). The intuition is that the training data are selected by a known procedure defined by a domain expert (e.g., random substitutions to a known wild-type sequence as in Brookes et al. (2019); Biswas et al. (2021); Bryant et al. (2021)), and the (shifted) test distribution is a *designed* distribution shift, intentionally selected so that "test" protein sequences that are expected to have high fitness. That is, the biomolecular engineering and design literature commonly uses other optimization procedures (such as training a generative model) where the test/design distribution is also explicitly known (Brookes et al., 2022; Fannjiang & Listgarten, 2020; Popova et al., 2018; Kang & Cho, 2018; Russ et al., 2020; Wu et al., 2020; Hawkins-Hooker et al., 2021; Shin et al., 2021) or that produce an implicit test distribution that can be more easily simulated, and thus estimated (Killoran et al., 2017; Gómez-Bombarelli et al., 2018; Linder et al., 2020; Sinai et al., 2020; Bashir et al., 2021; Bryant et al., 2021), compared to naive density estimation with unknown shifts. Accordingly, for our protein design experiments we evaluate our methods and relevant baselines using oracle weights.

5.1.1. Results of JAW-FCS: Coverage and Interval Width

For the protein design problem, a predictive inference method is reliable if it provides confidence intervals with valid coverage. Beyond this necessary criterion, however, a method is most useful if it moreover enables a protein engineer to identify promising candidate sequences that are predicted, with minimal uncertainty (minimal interval width), to have maximum fitness. Accordingly, we consider a predictive inference method to have ideal statistical properties if it maintains coverage at or above the user-specified target confidence level $1-\alpha$, if the predicted fitness of its designed sequences are as high as possible, and if its interval widths are as small and thus as informative as possible. In Figure 1 we thus plot these three performance criteria (coverage, predicted fitness, and interval width) for JAW-FCS and several baselines across a grid of shift magnitudes $\lambda \in \{0, 1, 2, 3\}$. We provide these plots across both the red and blue fluorescence datasets for both a neural network and a random forest predictor function $\widehat{\mu}$.

For all dataset × predictor conditions in Figure 1, JAW-FCS and weighted split conformal prediction maintain coverage at the target level of $1-\alpha = 0.9$ regardless of data shift magnitude λ, whereas the traditional, exchangeable-data versions of each of the two methods (jackknife+ and split conformal) lose coverage. This result empirically verifies the coverage guarantees for JAW-FCS and weighted split under FCS. However, for all dataset × predictor conditions, the $\widehat{\mu}$ predictor corresponding to the JAW-FCS method proposes designed sequences with higher mean predicted fitness than the $\widehat{\mu}$ predictor corresponding to weighted split method, and the JAW-FCS prediction intervals are generally smaller and thus more informative than the weighted split intervals. The superior predicted fitness and interval width values for JAW-FCS relative to weighted split conformal largely reflect the former's greater statistical efficiency relative to the latter. That is, by avoiding sample splitting, JAW-FCS maintains a

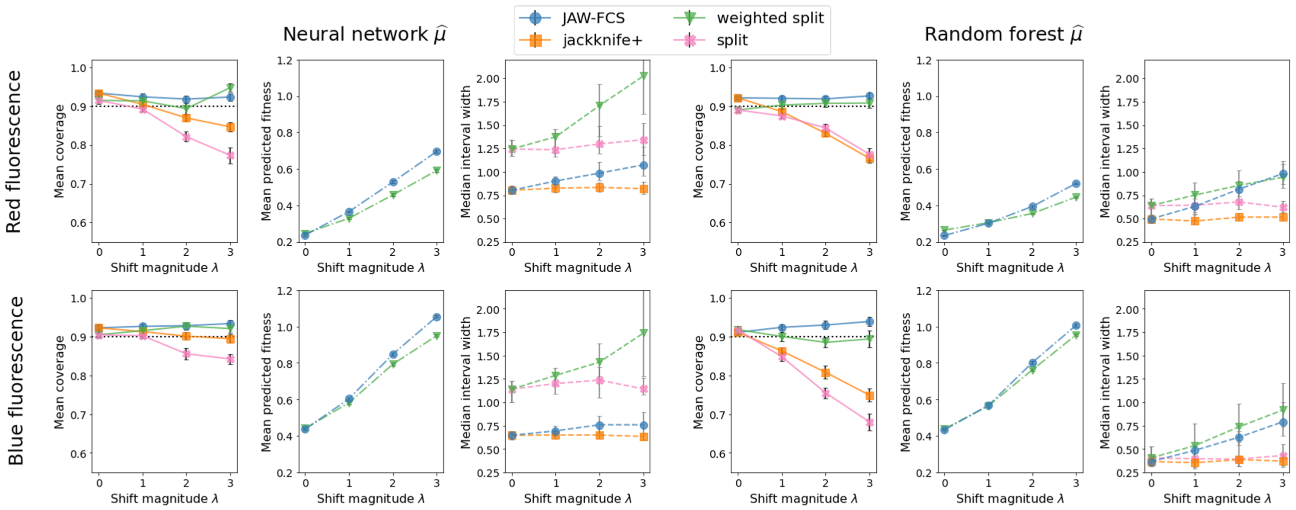

Figure 1. Mean prediction interval coverage, mean predicted fitness (i.e., mean predicted fluorescence for test point "designed" protein sequence), and median interval width for our proposed JAW-FCS method (blue circles) and its baselines, across both red fluorescence and blue fluorescence datasets with both neural network and random forest predictor functions $\widehat{\mu}$. The baselines are jackknife+ (orange squares), weighted split conformal (green triangles), and traditional split conformal (pink Xs); the predicted fitness values for JAW-FCS and jackknife+ are identical, as are those for weighted and traditional split conformal. All values are for 20 repeated experiments with a distinct, 192-sample training dataset per experiment, and each experiment consisted of 200 test points, so that 4000 test points used to calculate each plotted value. Black error bars for predicted fitness and coverage represent standard error, and gray bars for median interval width represent upper and lower quartiles. On the coverage plot, the target coverage level of $1 - \alpha = 0.9$ is shown with a black dotted line. JAW-FCS maintains coverage at the target level regardless of shift magnitudes λ, with higher mean predicted fitness and smaller, more informative prediction intervals than weighted split conformal, the only baseline that also maintains target coverage.

$\widehat{\mu}$ predictor that is trained on more data and therefore more "competent" than that of weighted split, and JAW-FCS is also able to efficiently use all of its labeled data for the construction of more precise predictive intervals.

5.1.2. Results for Protein Design with JAW-FCS Computational Relaxations

We now turn to evaluating the performance of WCV+FCS and JAW-KLOO, our two proposed computational relaxations of JAW-FCS, relative to the full JAW-FCS method for a range of computational budgets K. In Figure 2, we compare the methods' coverage and interval width across a grid of $K \in \{16, 24, 32, 48, 96, 192\}$, where K refers to the number of times that the predictor is retrained, across the same four protein dataset × predictor conditions. We also include standard CV+ as an additional baseline. We omit the predicted fitness values because all these approaches avoid sample splitting and thus correspond to the same predictor.

In Figure 2, we see that both WCV+ and JAW-KLOO maintain coverage at or above the target $\alpha = 0.9$ level for computational budgets $K \in \{16, 24, 32, 48, 96, 192\}$, whereas standard CV+ loses coverage. Notably, the prediction interval widths for WCV+ are largely comparable to those of the full JAW-FCS method for all evaluated computational budgets K, but for smaller values of K, JAW-KLOO has overly conservative prediction intervals that are much wider than those of the full JAW-FCS model. These results favor the use of WCV+ as a practical computational relaxation of JAW-FCS that empirically appears to avoid loss of coverage or increase in interval widths even for smaller computational budgets K. However, due to the fact that JAW-KLOO achieves the same coverage guarantee as JAW-FCS (while WCV+ has a weaker guarantee), in some cases, JAW-KLOO could offer a conservative computational relaxation to JAW-FCS when there is a severe miscoverage penalty.

5.2. Computational Relaxations Under SCS

Datasets for SCS Experiments For the standard covariate shift setting, we use the same five UCI datasets (Dua & Graff, 2017) used for experiments in Prinster et al. (2022): airfoil self-noise, red wine quality prediction (Cortez et al., 2009), wave energy converters, superconductivity (Hamidieh, 2018), and communities and crime (Redmond & Baveja, 2002), which represent a range of different dimensionalities. We follow the procedure described in Prinster et al. (2022) for the creation of standard covariate shift, and we refer to that work for details. The same predictors as used

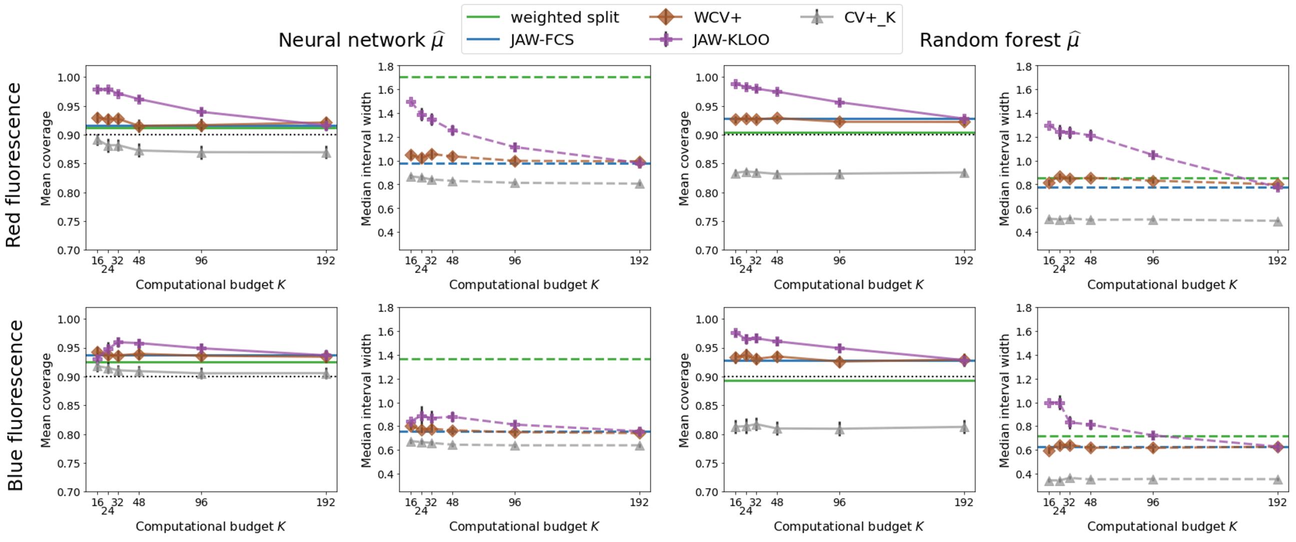

Figure 2. Mean coverages and median interval widths for computational relaxations of JAW-FCS—WCV+ (brown diamond) and JAW-KLOO (violet plus shape) methods—relative to full JAW-FCS (blue line), weighted split conformal (green line), and standard CV+ (gray triangles), for a range of computational budgets K (where K refers to the number of times that the model is retrained). All values are for 20 repeated experiments with a distinct, 192-sample training dataset per experiment, with shift magnitude $\lambda = 2$, and each experiment consisted of 200 test points, so that 4000 test points used to calculate each plotted value. Black error bars for predicted fitness and coverage represent standard error, and gray bars for median interval width represent upper and lower quartiles. On the coverage plot, the target coverage level of $1 - \alpha = 0.9$ is shown with a black dotted line. Both WCV+ and JAW-KLOO maintain coverage above the target level of $1 - \alpha = 0.9$ for all tested computational budget values K. While the predictive intervals for JAW-KLOO are overly wide for smaller values of K, the predictive intervals for WCV+ are comparable to those of JAW-FCS for all values of K.

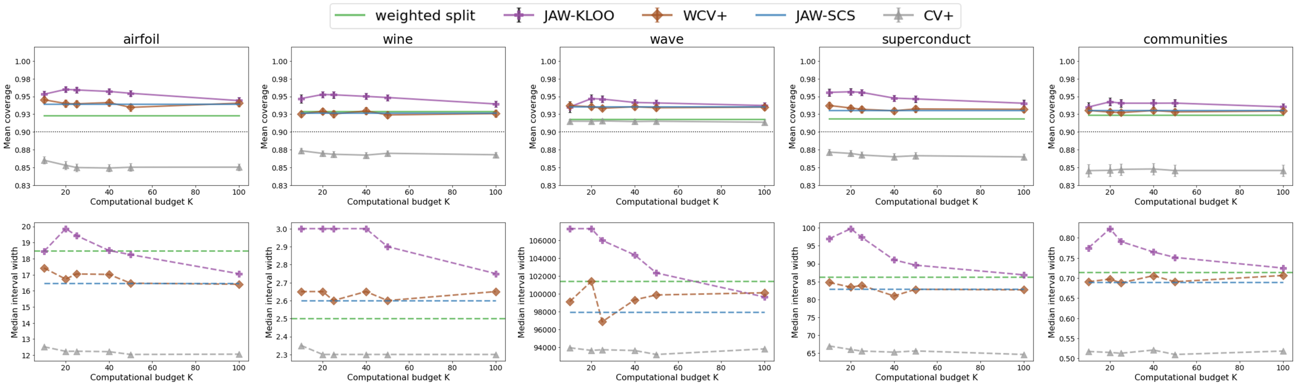

Figure 3. Mean coverage (top row) and median interval widths (bottom row) for proposed JAW-KLOO (violet plus shape) and WCV+ (brown diamond) methods under standard covariate shift (SCS), compared to JAW-SCS (blue line), weighted split (green line), and standard CV+ (gray triangles) baselines, for a range of computational budgets K. Results for five UCI datasets with a random forest predictor are reported, across 75 repeated experiments each corresponding to different 200-sample training datasets, with 200 test points per experiment. JAW-KLOO and WCV+ maintain coverage at the target level of $1 - \alpha = 0.9$ across all datasets and all tested computational budget values K, and WCV+ maintains comparably informative interval widths relative to JAW-SCS. Supplementary results for the neural network predictor are provided in Appendix B.

in the FCS experiments were used in the SCS experiments.

Figure 3 shows the mean coverage and median interval width results for JAW-KLOO and WCV+ (our two proposed computational relaxations of JAW for FCS or SCS) under SCS compared with several baselines and across five UCI datasets. The baselines are weighted split conformal prediction, JAW-SCS (JAW under standard covariate shift as in Prinster et al. (2022)), and CV+. WCV+ and JAW-KLOO maintain coverage at the target level of 0.9 for all datasets and predictor functions μ, along with the other methods with coverage guarantees under covariate shift (JAW-SCS and weighted split conformal prediction). These results demonstrate that computationally relaxing JAW in the form of either w-CV+ or JAW-KLOO does not come at the cost of lowering the empirical coverage of the methods below the target coverage (thus experimentally validating the coverage guarantees for both computational relaxations). Moreover, across datasets, the predictive intervals for WCV+ have comparable width to the JAW-SCS method, which suggests that WCV+ is a practical computational relaxation of JAW under standard covariate shift that avoids sacrificing either coverage validity or predictive interval sharpness. These findings are similar to the results under FCS, which provides further evidence for the utility of our proposed methods under either standard or feedback covariate shift settings.

5.3. Experiments with Estimated FCS Weights

Active Learning Exploration with Probabilistic Bounds While in Section 5.1 the input distributions (and thus the FCS weights) are known for the protein design task, in other settings of feedback covariate shift the weights may require some estimation. For instance, take high-stakes or resource-constrained exploration in active learning, where predictive intervals for query (test) points can help (probabilistically) bound risks or costs associated with the "annotation" procedure (e.g., invasive medical diagnostic procedures or expensive lab experiments). In each active learning iteration, the training data are updated with newly labeled "query" points based on a systematic querying strategy, which will usually result in an early-iteration training distribution differing drastically from the distribution at a later iteration. Meanwhile, the distribution of the query (test) points can also change as the model becomes more "informed". However, with common active learning setups such as pool-based active learning with uncertainty sampling, the query or test distribution shift is a direct function an attribute of the model predictions (e.g., least confidence, prediction entropy) (Nguyen et al., 2022; Brochu et al., 2010; Settles, 2009), which means that the test distribution is known. Therefore, FCS likelihood-ratio weight estimation in active learning often reduces to density estimation of only the training data.

To emperically evaluate our proposed predictive inference methods with estimated rather than oracle weights, we implemented a pool-based active learning task with the NASA Airfoil Self-Noise Dataset (Dua & Graff, 2017). We used kernel density estimation (with a Gaussian kernel) to estimate the density of the labeled training data, and query/test data were sampled from a larger pool of unlabeled points with likelihoods proportional to the predicted variance at each point from a Gaussian process (with a dot product kernel and added white noise), which is a common practice (Yue et al., 2020). Figure 4 firstly demonstrates that our proposed (JAW-FCS, JAW-KLOO, and WCV+) methods can achieve predictive interval coverage above the target level of $1 - \alpha = 0.9$ even with estimated rather than oracle weights. Moreover, in Figure 4 our methods generally maintain smaller (and thus more informative) interval widths than the weighted split conformal prediction baseline as well as achieve smaller validation-set mean squared error.

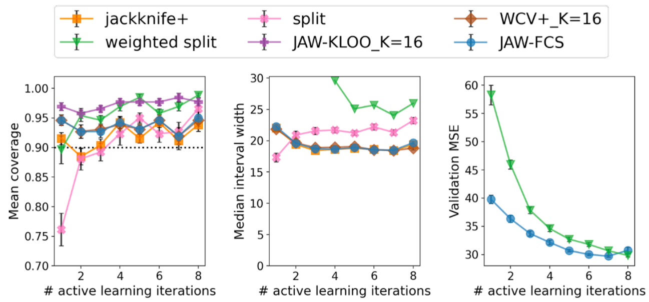

Figure 4. Mean coverage (left), median interval width (middle), and validation-set mean squared error (right) on the active learning predictive inference experiments on the NASA Airfoil Self-Noise Dataset (Dua & Graff, 2017). 30 experimental replicates were performed each with a unique random seed and 8 active learning query iterations, where 16 points were queried in each iteration. These results firstly demonstrate that our proposed methods (JAW-FCS, JAW-KLOO, and WCV+) can achieve target coverage ($1 - \alpha = 0.9$) even with estimated rather than oracle weights, while moreover maintaining smaller (more informative) interval widths and lower validation-set MSE.

6. Conclusion

In this paper, we propose several computationally and statistically efficient distribution-free predictive inference methods under both standard and feedback covariate shift. We provide rigorous coverage guarantees and validate them in real-world feedback covariate shift problems including protein design and standard covariate shift scenarios with real-world benchmark datasets. Our methods achieve a substantial speedup in computational demands relative to full conformal-FCS as well as a considerable improvement to predictor performance and interval sharpness relative to weighted split conformal prediction, without losing coverage. We moreover demonstrate that our proposed methods achieve target performance even with estimated rather than oracle weights on an active learning task, although the fact that our guarantees assume oracle weights is a limitation of our work. Promising future directions include examining weaker guarantees that account for error in weight estimation and generalizations to other distribution shifts that occur due to sequential decision-making under feedback loops.

Software and Data

Anonymous Google Drive link to code for experiments: https://drive.google.com/drive/folders/19DPEh22G0JmUP0m6huHN3k7PhV67fDll?usp=share_link

References

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

Barber, R. F., Candes, E. J., Ramdas, A., and Tibshirani, R. J. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486–507, 2021.

Barber, R. F., Candes, E. J., Ramdas, A., and Tibshirani, R. J. Conformal prediction beyond exchangeability. *arXiv preprint arXiv:2202.13415*, 2022.

Bashir, A., Yang, Q., Wang, J., Hoyer, S., Chou, W., McLean, C., Davis, G., Gong, Q., Armstrong, Z., Jang, J., et al. Machine learning guided aptamer refinement and discovery. *Nature Communications*, 12(1):2366, 2021.

Biswas, S., Khimulya, G., Alley, E. C., Esvelt, K. M., and Church, G. M. Low-n protein engineering with data-efficient deep learning. *Nature methods*, 18(4):389–396, 2021.

Brochu, E., Cora, V. M., and De Freitas, N. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

Brookes, D., Park, H., and Listgarten, J. Conditioning by adaptive sampling for robust design. In *International conference on machine learning*, pp. 773–782. PMLR, 2019.

Brookes, D. H., Aghazadeh, A., and Listgarten, J. On the sparsity of fitness functions and implications for learning. *Proceedings of the National Academy of Sciences*, 119 (1):e2109649118, 2022.

Bryant, D. H., Bashir, A., Sinai, S., Jain, N. K., Ogden, P. J., Riley, P. F., Church, G. M., Colwell, L. J., and Kelsic, E. D. Deep diversification of an aav capsid protein by machine learning. *Nature Biotechnology*, 39(6):691–696, 2021.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4): 547–553, 2009.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Fannjiang, C. and Listgarten, J. Autofocused oracles for model-based design. *Advances in Neural Information Processing Systems*, 33:12945–12956, 2020.

Fannjiang, C., Bates, S., Angelopoulos, A. N., Listgarten, J., and Jordan, M. I. Conformal prediction under feedback covariate shift for biomolecular design. *Proceedings of the National Academy of Sciences*, 119(43):e2204569119, 2022.

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

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

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

Hawkins-Hooker, A., Depardieu, F., Baur, S., Couairon, G., Chen, A., and Bikard, D. Generating functional protein variants with variational autoencoders. *PLoS computational biology*, 17(2):e1008736, 2021.

Kang, S. and Cho, K. Conditional molecular design with deep generative models. *Journal of chemical information and modeling*, 59(1):43–52, 2018.

Killoran, N., Lee, L. J., Delong, A., Duvenaud, D., and Frey, B. J. Generating and designing dna with deep generative models. *arXiv preprint arXiv:1712.06148*, 2017.

Landau, H. 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.

Linder, J., Bogard, N., Rosenberg, A. B., and Seelig, G. A generative neural network for maximizing fitness and diversity of synthetic dna and protein sequences. *Cell systems*, 11(1):49–62, 2020.

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

Nguyen, V.-L., Shaker, M. H., and Hüllermeier, E. How to measure uncertainty in uncertainty sampling for active learning. *Machine Learning*, 111(1):89–122, 2022.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Poelwijk, F. J., Socolich, M., and Ranganathan, R. Learning the pattern of epistasis linking genotype and phenotype in a protein. *Nature communications*, 10(1):1–11, 2019.

Popova, M., Isayev, O., and Tropsha, A. Deep reinforcement learning for de novo drug design. *Science advances*, 4(7): eaap7885, 2018.

Prinster, D., Liu, A., and Saria, S. Jaws: Auditing predictive uncertainty under covariate shift. *Thirty-sixth Conference on Neural Information Processing Systems*, pp. arXiv:2207.10716, 2022.

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

Russ, W. P., Figliuzzi, M., Stocker, C., Barrat-Charlaix, P., Socolich, M., Kast, P., Hilvert, D., Monasson, R., Cocco, S., Weigt, M., et al. An evolution-based model for designing chorismate mutase enzymes. *Science*, 369 (6502):440–445, 2020.

Settles, B. Active learning literature survey. 2009.

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

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

Shin, J.-E., Riesselman, A. J., Kollasch, A. W., McMahon, C., Simon, E., Sander, C., Manglik, A., Kruse, A. C., and Marks, D. S. Protein design and variant prediction using autoregressive generative models. *Nature communications*, 12(1):2403, 2021.

Sinai, S. and Kelsic, E. D. A primer on model-guided exploration of fitness landscapes for biological sequence design. *arXiv preprint arXiv:2010.10614*, 2020.

Sinai, S., Wang, R., Whatley, A., Slocum, S., Locane, E., and Kelsic, E. D. Adalead: A simple and robust adaptive greedy search algorithm for sequence design. *arXiv preprint arXiv:2010.02141*, 2020.

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

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

Sugiyama, M., Krauledat, M., and Müller, K.-R. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(5), 2007.

Tibshirani, R. J., Foygel Barber, R., Candes, E., and Ramdas, A. Conformal prediction under covariate shift. *Advances in neural information processing systems*, 32, 2019.

Vovk, V. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1):9–28, 2015.

Vovk, V., Gammerman, A., and Shafer, G. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

Wittmann, B. J., Yue, Y., and Arnold, F. H. Informed training set design enables efficient machine learning-assisted directed protein evolution. *Cell systems*, 12(11):1026–1045, 2021.

Wu, Z., Kan, S. J., Lewis, R. D., Wittmann, B. J., and Arnold, F. H. Machine learning-assisted directed protein evolution with combinatorial libraries. *Proceedings of the National Academy of Sciences*, 116(18):8852–8858, 2019.

Wu, Z., Yang, K. K., Liszka, M. J., Lee, A., Batzilla, A., Wernick, D., Weiner, D. P., and Arnold, F. H. Signal peptides generated by attention-based neural networks. *ACS Synthetic Biology*, 9(8):2154–2161, 2020.

Wu, Z., Johnston, K. E., Arnold, F. H., and Yang, K. K. Protein sequence design with deep generative models. *Current opinion in chemical biology*, 65:18–27, 2021.

Yang, K. K., Wu, Z., and Arnold, F. H. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, 16(8):687–694, 2019.

Yue, X., Wen, Y., Hunt, J. H., and Shi, J. Active learning for gaussian process considering uncertainties with application to shape control of composite fuselage. *IEEE Transactions on Automation Science and Engineering*, 18 (1):36–46, 2020.

Zhu, D., Brookes, D. H., Busia, A., Carneiro, A., Fannjiang, C., Popova, G., Shin, D., Donohue, K. C., Chang, E. F., Nowakowski, T. J., et al. Optimal trade-off control in machine learning-based library design, with application to adeno-associated virus (aav) for gene therapy. *bioRxiv*, pp. 2021–11, 2021.

A. Proofs for theoretical results.

A.1. Preliminaries

Data from feedback covariate shift (FCS) as in (5) are a special case of what Fannjiang et al. (2022) call *pseudo-exchangeable* random variables.

Definition A.1. *Random variables $V_1, ..., V_{n+1}$ are* pseudo-exchangeable *with factor functions $w_1, ..., w_{n+1}$ and core function h if the density, f, of their joint distribution can be factorized as*

$$f(v_1, ..., v_{n+1}) = \prod_{i=1}^{n+1} w_i(v_i; v_{-i}) \cdot h(v_1, ..., v_{n+1}) \tag{15}$$

where $v_{-i} = v_{1:(n+1)} \backslash v_i$, each $w_i(\cdot; v_{-i})$ is a function that depends on the multiset v_{-i} (that is, on the values in v_{-i} but not on their ordering), and h is a function that does not depend on the ordering of its $n+1$ inputs.

The proofs for our theoretical results leverage the observation that any subsequence of pseudo-exchangeable random variables is itself pseudo-exchangeable, which we state formally in the following lemma.

Lemma A.2. *Let $(V_1, ..., V_{n+1})$ be a sequence of pseudo-exchangeable random variables with factor functions $w_1, ..., w_{n+1}$. For any $J = \{j_1, ..., j_m\} \subseteq \{1, ..., n+1\}$, the subsequence $(V_{j_1}, ..., V_{j_m})$ is pseudo-exchangeable.*

Proof. Let $J = \{j_1, ..., j_m\} \subseteq \{1, ..., n+1\}$ denote an arbitrary set of indices so that $(V_{j_1}, ..., V_{j_m})$ is a subsequence of $(V_1, ..., V_{n+1})$, and let $J^C = \{j'_1, ..., j'_{n+1-m}\}$. Then, we can integrate (15) over all $v_{j'}$ such that $j' \in J^C$:

$$\int_{v_{j'_1}} \cdots \int_{v_{j'_{n+1-m}}} f(v_1, ..., v_{n+1}) \, dv_{j'_1} \dots dv_{j'_{n+1-m}} = \int_{v_{j'_1}} \cdots \int_{v_{j'_{n-m}}} \prod_{i=1}^{n+1} w_i(v_i; v_{-i}) \cdot h(v_1, ..., v_n) \, dv_{j'_1} \dots dv_{j'_{n+1-m}} \tag{16}$$

such that the right-hand side of (16) no longer depends on any specific value of $v_{j'}$ for $j' \in J^C$. That is, letting $v_J = \{v_j : j \in J\}$, we can write (16) as

$$f_J(v_{j_1}, ..., v_{j_m}) = \prod_{j \in J} w_j(v_j; v_J \backslash v_j) \cdot g_J(v_{j_1}, ..., v_{j_m}), \tag{17}$$

for some weight functions $w_j(\cdot; v_J \backslash v_j)$ and some core function g_J that does not depend on the ordering of its inputs. Therefore, the subsequence $(V_{j_1}, ..., V_{j_m})$ is pseudo-exchangeable. □

A.2. Proof for JAW-FCS coverage under feedback covariate shift

We first restate the JAW-FCS coverage guarantee before proceeding with the proof.

Theorem 3.1 *Suppose data are generated under feedback covariate shift (5) and assume $\widetilde{P}_{X;D}$ is absolutely continuous with respect to P_X for all possible values of D. Then, for any miscoverage level, $\alpha \in (0, 1)$, the JAW-FCS predictive interval in (7) satisfies*

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}_{n,\alpha}^{\text{JAW-FCS}}(X_{n+1})\} \geq 1 - 2\alpha. \tag{18}$$

Our proof technique generalizes the proof for JAW-SCS presented in Prinster et al. (2022) from standard covariate shift to feedback covariate shift. Since the proof for JAW-SCS in Prinster et al. (2022) is itself a generalization of the jackknife+ coverage proof (for exchangeable data) in Barber et al. (2021), the proof we present here is thus also a generalization of the jackknife+ coverage proof.

We use (a) - (e) 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 both Theorem 1 in Prinster et al. (2022) and 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 techniques in both Prinster et al. (2022) and Barber et al. (2021):

(c) We first introduce a generalized version of the weights defined in (6):

$$\tilde{w}_{i,j}(X_{n+1}) = \frac{w(X_i; z_{-\{i,j\}})w(X_j; z_{-\{i,j\}})}{\sum_{j'=1}^{n+1}\left[w(X_i; z_{-\{i,j'\}})w(X_{j'}; z_{-\{i,j'\}})\right]} \tag{19}$$

Note that letting $i = n+1$ in (19) yields the weights defined in (6). Hereon, for simplicity we refer to the terms $\tilde{w}_{i,j}(X_{n+1})$ as defined in (19) as "normalized weights"—these quantities reduce to the normalized likelihood ratio weights used in Prinster et al. (2022) under standard covariate shift, where $w(X_i; z_{i,j'}) = w(X_i) = w(X_i; z_{i,j})$ for all $j, j' \in \{1, ..., n+1\}$.

(d) We define a weighted comparison matrix that we call $\hat{A}^w \in \mathbb{R}^{(n+1)\times(n+1)}$. In order to describe $\hat{A}^w$, let us first define A as an unweighted comparison matrix (as in Barber et al. (2021)) with entries $A_{ij} = \mathbb{1}\{R_{ij} > R_{ji}\}$—i.e.,indicators for the event that, when i and j are excluded from the regression fitting, i has larger residual than j. Also, define $\hat{W}$ as the weight matrix with entries $\hat{W}_{ij} = w(X_i; z_{-\{i,j\}})$ (we note that Prinster et al. (2022) defined a similar weight matrix W, except under the special case of standard covariate shift with likelihood ratio weight functions that are not determined by the training data like in feedback covariate shift). Then, define $\hat{A}^w = \hat{W} \odot A \odot \hat{W}^\top$ where $\odot$ denotes pointwise multiplication, so that $\hat{A}^w$ has entries $\hat{A}^w_{ij} = w(X_i; z_{-\{i,j\}}) \cdot w(X_j; z_{-\{i,j\}}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\}$. For any $i, j \in \{1, ..., n+1\}$, note that $\hat{A}^w_{ij} > 0$ implies $\hat{A}^w_{ji} = 0$ for any $i, j \in \{1, ..., n+1\}$. (Moreover, note that for exchangeable data, $w(X_i; z_{-\{i,j\}}) = w(X_j; z_{-\{i,j\}}) = 1$ for all $i, j \in \{1, ..., n+1\}$ and the weighted comparison matrix $\hat{A}^w$ becomes equivalent to the unweighted comparison matrix A described in Barber et al. (2021).)

(e) Next, as in Prinster et al. (2022) and 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, as in the standard covariate shift setting (Prinster et al., 2022), for feedback covariate shift we need to account for the fact that the informativeness of the comparison $\mathbb{1}\{R_{ij} > R_{ji}\}$ depends on relative weight or likelihood of the point j in the test distribution relative to the training distribution. 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 $\mathcal{S}(\hat{A}^w) \subseteq \{1, ..., n+1\}$ in the following three equivalent ways that each serve a different illustrative purpose:

$$\mathcal{S}(\hat{A}^w) = \left\{i \in [n+1] \ : \ \sum_{j=1}^{n+1}\left(\tilde{w}_{i,j}(X_{n+1}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\}\right) \geq 1 - \alpha\right\}$$

$$\mathcal{S}(\hat{A}^w) = \left\{i \in [n+1] \ : \ \sum_{j=1}^{n+1}\left(w(X_i; z_{-\{i,j\}})w(X_j; z_{-\{i,j\}}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\}\right) \geq (1-\alpha)\sum_{j'=1}^{n+1}\left[w(X_i; z_{-\{i,j'\}})w(X_{j'}; z_{-\{i,j'\}})\right]\right\}$$

$$\mathcal{S}(\hat{A}^w) = \left\{i \in [n+1] \ : \ \sum_{j=1}^{n+1}\hat{A}^w_{ij} \geq (1-\alpha)\sum_{j=1}^{n+1}\hat{W}^2_{ij}\right\}$$

The first definition represents our intuition of $\mathcal{S}(\hat{A}^w)$ as a set of "strange" points, which we have described (where $\mathbb{1}\{R_{ij} > R_{ji}\}$ for a sufficiently large total normalized weight $\tilde{w}_{i,j}(X_{n+1})$ (defined in (6)) of other points j). That is, in the first definition it is relatively straightforward to see how $\mathcal{S}(\hat{A}^w) \subseteq \{1, ..., n+1\}$ 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 $\tilde{w}_{j,i}(X_{n+1})$ of all such points j is sufficiently large (at least $1-\alpha$). On the other hand, the second and third definitions represents how the set of strange points can be computed from the weighted comparison matrix $\hat{A}^w$, where the third line is a condensed version of the second. That is, in the second line the sum on the left side of the inequality $w(X_i; z_{-\{i,j\}})w(X_j; z_{-\{i,j\}}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\}$ is the sum over the ith row in $\hat{A}^w = \hat{W} \odot \hat{A}^w \odot \hat{W}^\top$, while the sum on the right side of the inequality is the sum over the ith row in $\hat{W} \odot \hat{W}^\top$. The fact that the set of strange points can be computed directly from the weighted comparison matrix $\hat{A}^w$ is important to the second main step of our proof. (In the absence of covariate shift—i.e., for exchangeable data—when $w_k = 1$ for all $k \in \{1, ..., n+1\}$, these definitions are equivalent to the set of strange points in the jackknife+ coverage proof in Barber et al. (2021).)

We now proceed to the main steps of our proof, which generalize the corresponding proof steps in Prinster et al. (2022) and Barber et al. (2021) to allow for feedback covariate shift:

- Step 1: Establish that $\mathbb{E}\big[\sum_{j\in\mathcal{S}(\hat{A}^w)} \tilde{w}_{i,j}(X_{n+1})\big] \leq 2\alpha$. That is, $\sum_{j\in\mathcal{S}(\hat{A}^w)} \tilde{w}_{i,j}(X_{n+1})$, the total normalized weight of strange points in any row i of any comparison matrix $\hat{A}^w$, is in expectation no more than 2α.
- Step 2: Using the fact that the datapoints are pseudo-exchangeable, show that the probability that the test point $n+1$ is strange (i.e., $n+1 \in \mathcal{S}(\hat{A}^w)$) 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 expected total normalized weight of the strange points. This proof step follows and generalizes the corresponding proof step for Theorem 1 in Prinster et al. (2022) as well as for Theorem 1 in Barber et al. (2021), which rely 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 $\hat{A}^w_{ij} > 0$, 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.

The analogous proof step in Barber et al. (2021) derives a bound on the *number* of strange points from a bound on the *number of pairs* of strange points, and Prinster et al. (2022) extends this step to standard covariate shift by deriving 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. This idea in Prinster et al. (2022) 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, 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. We thus follow the approach of Prinster et al. (2022), except with more general weights that allow for feedback covariate shift (whereas the proof in Prinster et al. (2022) is limited to standard covariate shift).

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

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

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 $i \in \mathcal{S}(\hat{A}^w)$ we construct a rectangle L_i with width equal point i's normalized weight, $\tilde{w}_{i,i}(X_{n+1})$, and length equal to the largest total normalized weight that the points that i loses to could have, $\alpha - \tilde{w}_{i,i}(X_{n+1})$. In addition, suppose that we also construct a second rectangle L'_i for each strange point $i \in \mathcal{S}(\hat{A}^w)$ with width equal to $\tilde{w}_{i,i}(X_{n+1})$—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, $\frac{1}{2}\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i}\ \tilde{w}_{j,i}(X_{n+1})$.

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., $\tilde{w}_{i,j}(X_{n+1})$) with an upper bound on the total normalized weight that the points i loses to could have (i.e., $\alpha - \tilde{w}_{i,j}(X_{n+1})$). Thus, the area of L_i is by construction an upper bound on the product of point i's normalized weight (i.e., $\tilde{w}_{i,j}(X_{n+1})$) 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 $\tilde{w}_{i,j}(X_{n+1})$ and length $\tilde{w}_{j,i}(X_{n+1})$. 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 $\tilde{w}_{i,j}(X_{n+1})$ and length $\frac{1}{2}\tilde{w}_{j,i}(X_{n+1})$ (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 $\tilde{w}_{j,i}(X_{n+1})$ and length $\frac{1}{2}\tilde{w}_{i,j}(X_{n+1})$ contained in L'_j. Thus, the combined area of L'_{ji} and L'_{ij} is $\text{Area}(L'_{ij}) + \text{Area}(L'_{ji}) = \tilde{w}_{i,j}(X_{n+1}) \cdot \tilde{w}_{j,i}(X_{n+1})$, 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 $s = |S(\hat{A}^w)|$ 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 $\sum_{i,j\in\mathcal{S}(\hat{A}^w),\ i\neq j} \text{Area}(L'_{ij}) = \sum_{i,j\in\mathcal{S}(\hat{A}^w),\ i\neq j} \text{Area}(L_{ij}) \leq \sum_{i\in\mathcal{S}(\hat{A}^w),\ i \text{ loses to } j} \text{Area}(L_{ij})$. Moreover, by construction $\sum_{i,j\in\mathcal{S}(\hat{A}^w),\ i\neq j} \text{Area}(L'_{ij}) = \sum_{i\in\mathcal{S}(\hat{A}^w)} \text{Area}(L'_i)$ and $\sum_{i\in\mathcal{S}(\hat{A}^w),\ i \text{ loses to } j} \text{Area}(L_{ij}) \leq \sum_{i\in\mathcal{S}(\hat{A}^w)} \text{Area}(L_i)$. 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:

$$\sum_{i\in\mathcal{S}(\hat{A}^w)} \text{Area}(L'_i) \leq \sum_{i\in\mathcal{S}(\hat{A}^w)} \text{Area}(L_i)$$

$$\sum_{i\in\mathcal{S}(\hat{A}^w)} \left(\tilde{w}_{i,i}(X_{n+1}) \cdot \frac{1}{2} \sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\right) \leq \sum_{i\in\mathcal{S}(\hat{A}^w)} \left(\tilde{w}_{i,i}(X_{n+1}) \cdot \left(\alpha - \tilde{w}_{i,i}(X_{n+1})\right)\right) \tag{20}$$

Recall that we defined $\tilde{w}_{i,j}(X_{n+1}) = \frac{w(X_i; z_{-\{i,j\}}) w(X_j; z_{-\{i,j\}})}{\sum_{j'=1}^{n+1}\left[w(X_i; z_{-\{i,j'\}}) w(X_{j'}; z_{-\{i,j'\}})\right]}\ \forall\, i, j \in \{1, ..., n+1\}$, so in the uniform weighted case where $w(X_i; z_{-\{i,j\}}) = w(X_j; z_{-\{i,j\}}) = 1\ \forall\, i \in \{1, ..., n+1\}$ then the denominator of $\tilde{w}_{i,j}(X_{n+1})$ is equal to $n+1$, 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.

Note that in (20), the left-hand side and the right hand side have an identical outer summation and first term inside the summation—that is they both take the form $\sum_{i\in\mathcal{S}(\hat{A}^w)}\left(\tilde{w}_{i,i}(X_{n+1}) \cdot (\text{second term})\right)$. This allows us to take the expectation with respect to X_i of the second term on both sides while maintaining the inequality, where for condensed notation we denote $\mathbb{E}_i[\cdot] = \mathbb{E}_{X_i\sim P_{X_i}}[\cdot]$ to denote the expectation with respect to the random variable with index i:

$$\sum_{i\in\mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1})\cdot \mathbb{E}_i\Big[\frac{1}{2}\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\Big]\Big) \le \sum_{i\in\mathcal{S}(\hat{A}^w)}\Big(\tilde{w}_{i,i}(X_{n+1})\cdot\mathbb{E}_i\big[\alpha-\tilde{w}_{i,i}(X_{n+1})\big]\Big),$$

We can then proceed to solve for an upper bound on $\mathbb{E}_i\big[\sum_{j\in\mathcal{S}(\hat{A}^w)}\ \tilde{w}_{i,j}(X_{n+1})\big]$, the expected weight of strange points

$$\begin{aligned}
\sum_{i\in\mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1})\cdot \frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\Big]\Big) &\le \sum_{i\in\mathcal{S}(\hat{A}^w)}\Big(\tilde{w}_{i,i}(X_{n+1})\cdot\mathbb{E}_i\big[\alpha-\tilde{w}_{i,i}(X_{n+1})\big]\Big)\\
\frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\Big]\cdot\sum_{i\in\mathcal{S}(\hat{A}^w)}\Big(\tilde{w}_{i,i}(X_{n+1})\Big) &\le \mathbb{E}_i\big[\alpha-\tilde{w}_{i,i}(X_{n+1})\big]\sum_{i\in\mathcal{S}(\hat{A}^w)}\Big(\tilde{w}_{i,i}(X_{n+1})\Big)\\
\frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\Big] &\le \mathbb{E}_i\big[\alpha-\tilde{w}_{i,i}(X_{n+1})\big]\\
\frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\Big] &\le \alpha-\mathbb{E}_i\big[\tilde{w}_{i,i}(X_{n+1})\big]\\
\frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \tilde{w}_{i,j}(X_{n+1})\Big]+\frac{1}{2}\mathbb{E}_i\big[\tilde{w}_{i,i}(X_{n+1})\big] &\le \alpha-\frac{1}{2}\mathbb{E}_i\big[\tilde{w}_{i,i}(X_{n+1})\big]\\
\frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i} \big[\tilde{w}_{i,j}(X_{n+1})\big]+\tilde{w}_{i,i}(X_{n+1})\Big] &\le \alpha-\frac{1}{2}\mathbb{E}_i\big[\tilde{w}_{i,i}(X_{n+1})\big]\\
\frac{1}{2}\cdot\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)} \tilde{w}_{i,j}(X_{n+1})\Big] &\le \alpha-\frac{1}{2}\mathbb{E}_i\big[\tilde{w}_{i,i}(X_{n+1})\big]\\
\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)} \tilde{w}_{i,j}(X_{n+1})\Big] &\le 2\alpha-\mathbb{E}_i\big[\tilde{w}_{i,i}(X_{n+1})\big]\\
\mathbb{E}_i\Big[\sum_{j\in\mathcal{S}(\hat{A}^w)} \tilde{w}_{i,j}(X_{n+1})\Big] &\le 2\alpha
\end{aligned}\tag{21}$$

Step 2: Pseudo-exchangeability of the datapoints. We now leverage the pseudo-exchangeability of the data to show that, since the total weight of the strange points is in expectation at most 2α, that a test point has at most 2α probability of being strange. We organize this step into the following pieces:

- Step 2.1: Argue that $\hat{A}^w \overset{\mathrm{d}}{=} P_\pi\hat{A}^w P_\pi^\top$ for any $(n+1)\times(n+1)$ permutation matrix P_π.
- Step 2.2: Argue that $\mathbb{P}\{n+1\in\mathcal{S}(\hat{A}^w)\}=\mathbb{P}\{j\in\mathcal{S}(\hat{A}^w)\}$ for all $j\in\{1,...,n+1\}$.
- Step 2.3: Use the fact that the total weight of the strange points is at most 2α (from Step 1) to show that $\mathbb{P}\{n+1\in\mathcal{S}(\hat{A}^w)\}\le 2\alpha$.

Beginning with Step 2.1, for a permutation π of $\{1,...,n+1\}$, let P_π denote the corresponding permutation matrix—that is, $\pi(i')=i \iff P_\pi(i',i)=1$, which corresponds to the ith row in A becoming the i'th row in $P_\pi A$. Then, observe that $\hat{A}^w_{ii}=0$ for all $i\in\{1,...,n+1\}$, since $\mathbb{1}\{R_{ii}>R_{ii}\}=0$ for all i. So, since an entry in the diagonal of $\hat{A}^w$ will always be mapped to another location in the diagonal of $P_\pi\hat{A}^wP_\pi^\top$, then, deterministically, the diagonal entries of both $\hat{A}^w$ and $P_\pi\hat{A}^wP_\pi^\top$ will be all zeros. So, to prove $\hat{A}^w \overset{\mathrm{d}}{=} P_\pi\hat{A}^wP_\pi^\top$ it is sufficient to prove that $\hat{A}^w$ and $P_\pi\hat{A}^wP_\pi^\top$ are equivalent in distribution in their off diagonal entries.

Recall that $\hat{W}\odot A$ has entries $(\hat{W}\odot A)_{ij}=w(X_i;z_{-\{i,j\}})\cdot\mathbb{1}\{R_{ij}>R_{ji}\}$ (equivalent to A with each ith row weighted by $w(X_i;z_{-\{i,j\}})$) and that $A\odot\hat{W}^\top$ has entries $(A\odot\hat{W}^\top)_{ij}=w(X_j;z_{-\{i,j\}})\cdot\mathbb{1}\{R_{ij}>R_{ji}\}$ (equivalent to A with each jth column weighted by $w(X_j; z_{-\{i,j\}})$). Moreover, note that $P_\pi(\hat{W} \odot A)$—which results from permuting the rows of $\hat{W} \odot A$—does not change the column membership of any entry in $\hat{W} \odot A$. In particular, $P_\pi(\hat{W} \odot A)$ has entries $(P_\pi(\hat{W} \odot A))_{ij} = (\hat{W} \odot A)_{\pi(i)j}$. Similarly, $(A \odot \hat{W}^\top)P_\pi^\top$ does not change the row membership of any entry in $A \odot \hat{W}$, such that $(A \odot \hat{W}^\top)P_\pi^\top$ has entries $(A \odot \hat{W}^\top)P_\pi^\top{}_{ji} = (A \odot \hat{W}^\top)_{j\pi(i)}$. So, to show that $\hat{A}^w$ and $P_\pi \hat{A}^w P_\pi^\top$ are equivalent in distribution in their off diagonal entries, it is sufficient to show each jth column in $\hat{W} \odot A$ is equivalent in distribution to the jth column in $P_\pi(\hat{W} \odot A)$ and that aside from the initial diagonal entries in $A \odot \hat{W}^\top$, each ith row in $A \odot \hat{W}^\top$ is equivalent in distribution to the corresponding ith row in $(A \odot \hat{W}^\top)P_\pi^\top$.

To show $P_\pi(\hat{W} \odot A) \overset{\mathrm{d}}{=} \hat{W} \odot A$ aside from the initial diagonal entries of $\hat{W} \odot A$, we draw on and adapt ideas from the proof for Lemma 3 in Tibshirani et al. (2019). For simplicity we assume that the pairs (R_{ij}, R_{ji}) are distinct almost surely (the result holds in the general case as well, but the notation is more cumbersome). Using condensed notation for the data as $\{Z_1, ..., Z_{n+1}\} = \{(X_1, Y_1), ..., (X_{n+1}, Y_{n+1})\}$, denote by E_z the event that $\{Z_1, ..., Z_{n+1}\} = \{z_1, ..., z_{n+1}\}$, and let f denote the density function of the joint sample $Z_1, ..., Z_{n+1}$. To do so, we begin by conditioning on E_z and then inspecting the probability of the joint event $R_{n+1,j} = r_{ij}, R_{j,n+1} = r_{ji}$ for each $i \in \{1, ..., n+1\}$ in each jth column, which occurs when $Z_{n+1} = z_i$:

$$\begin{aligned}\mathbb{P}\{R_{n+1,j} = r_{ij}, R_{j,n+1} = r_{ji} \mid E_z\} &= \mathbb{P}\{Z_{n+1} = z_i \mid E_z\} \\ &= \frac{\sum_{\pi:\pi(n+1)=i} f(z_{\pi(1)}, ..., z_{\pi(n+1)})}{\sum_\pi f(z_{\pi(1)}, ..., z_{\pi(n+1)})},\end{aligned}$$

where the second line above follows by the same reasoning as in the proof for Lemma 3 in Tibshirani et al. (2019). Then, recalling that data from feedback covariate shift (5) are pseudo-exchangeable with weight functions $w_1 = ... = w_n = 1$ and $w_{n+1} = w = \frac{d\widetilde{P}_{X;D}}{dP_X}$, this becomes

$$\mathbb{P}\{R_{n+1,j} = r_{i,j}, R_{j,n+1} = r_{j,i} \mid E_z\} = \frac{\sum_{\pi:\pi(n+1)=i} w(x_{\pi(n+1)}; z_{-\{\pi(n+1),j\}}) h(z_{\pi(1)}, ..., z_{\pi(n+1)})}{\sum_\pi w(x_{\pi(n+1)}; z_{-\{\pi(n+1),j\}}) h(z_{\pi(1)}, ..., z_{\pi(n+1)})}$$

where the core function h does not depend on the order of its inputs, so we have

$$\begin{aligned}\mathbb{P}\{R_{n+1,j} = r_{i,j}, R_{j,n+1} = r_{j,i} \mid E_z\} &= \frac{\sum_{\pi:\pi(n+1)=i} w(x_{\pi(n+1)}; z_{-\{\pi(n+1),j\}}) h(z_1, ..., z_{n+1})}{\sum_\pi w(x_{\pi(n+1)}; z_{-\{\pi(n+1),j\}}) h(z_1, ..., z_{n+1})} \\ &= \frac{\sum_{\pi:\pi(n+1)=i} w(x_{\pi(n+1)}; z_{-\{\pi(n+1),j\}})}{\sum_\pi w(x_{\pi(n+1)}; z_{-\{\pi(n+1),j\}})} \\ &= \frac{w(x_i; z_{-\{i,j\}})}{\sum_{i'=1}^{n+1} w(x_{i'}; z_{-\{i',j\}})},\end{aligned}$$

which is equivalent to the jth column of $\hat{W}$ divided by a normalization constant. We can then rewrite this probability statement as

$$(R_{n+1,j}, R_{j,n+1}) \mid E_z \sim \sum_{i=1}^{n+1} \frac{w(x_i; z_{-\{i,j\}})}{\sum_{i'=1}^{n+1} w(x_{i'}; z_{-\{i',j\}})} \delta_{(r_{ij}, r_{ji})}.$$

Due to the conditioning on E_z, this is equivalent to

$$(R_{n+1,j}, R_{j,n+1}) \mid E_z \sim \sum_{i=1}^{n+1} \frac{w(X_i; Z_{-\{i,j\}})}{\sum_{i'=1}^{n+1} w(X_{i'}; Z_{-\{i',j\}})} \delta_{(R_{ij}, R_{ji})},$$

and since this statement holds for any $\{Z_1, ..., Z_{n+1}\} = \{z_1, ..., z_{n+1}\}$, marginalization yields

$$(R_{n+1,j}, R_{j,n+1}) \sim \sum_{i=1}^{n+1} \frac{w(X_i; Z_{-\{i,j\}})}{\sum_{i'=1}^{n+1} w(X_{i'}; Z_{-\{i',j\}})} \delta_{(R_{ij}, R_{ji})}.$$

More generally, substituting in any index $i' \in \{1, ..., n+1\}$ in for $n+1$ in the argument above yields

$$(R_{i',j}, R_{j,i'}) \sim \sum_{i=1}^{n+1} \frac{w(X_i; Z_{-\{i,j\}})}{\sum_{i'=1}^{n+1} w(X_{i'}; Z_{-\{i',j\}})} \delta_{(R_{ij}, R_{ji})}, \tag{22}$$

where the only difference is on the left-hand side.

Statement (22) tells us that within each jth column, draws of $(R_{i',j}, R_{j,i'})$ from this discrete distribution resemble the analogous draw $(R_{n+1,j}, R_{j,n+1})$ for the test point. That is, the distribution of $(R_{i',j}, R_{j,i'})$ in (22) is irrespective of the index i' and so these draws "look exchangeable". Thus, the distribution of the off diagonal entries in the jth column of $\hat{W} \odot A$ do not depend on the ordering of the elements. By a similar argument, the distribution of the off diagonal entries in the ith row of $A \odot \hat{W}$ do not depend on the ordering of the elements, and therefore $P_\pi \hat{A}^w P_\pi^\top \overset{\text{d}}{=} \hat{A}^w$ for any $(n+1) \times (n+1)$ permutation matrix P_π, the desired result for Step 2.1.

Because $P_\pi \hat{A}^w P_\pi^\top \overset{\text{d}}{=} \hat{A}^w$ from Step 2.1, this implies $\mathbb{P}\{j \in \mathcal{S}(P_\pi \hat{A}^w P_\pi^\top)\} = \mathbb{P}\{j \in \mathcal{S}(\hat{A}^w)\}$. Now, let P_π denote a specific permutation matrix that maps $n+1$ to j, that is where $P_\pi(j, n+1) = 1$. Then, deterministically, $n+1 \in \mathcal{S}(\hat{A}^w) \iff j \in \mathcal{S}(\Pi \hat{A}^w \Pi^\top)$, so we have

$$\mathbb{P}\{n+1 \in \mathcal{S}(\hat{A}^w)\} = \mathbb{P}\{j \in \mathcal{S}(P_\pi \hat{A}^w P_\pi^\top)\} = \mathbb{P}\{j \in \mathcal{S}(\hat{A}^w)\}$$

for all $j = 1, ..., n+1$. That is, an arbitrary training point j is equally likely to be strange as the test point $n+1$, which concludes Step 2.2.

Then, we begin Step 2.3 by multiplying by $\mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big]$ to obtain

$$\mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] \cdot \mathbb{P}\{n+1 \in \mathcal{S}(\hat{A}^w)\} = \mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] \cdot \mathbb{P}\{j \in \mathcal{S}(\hat{A}^w)\}$$

And summing over j, we have

$$\begin{aligned}
\sum_{j=1}^{n+1} \mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] \cdot \mathbb{P}\{n+1 \in \mathcal{S}(\hat{A}^w)\} &= \sum_{j=1}^{n+1} \mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] \cdot \mathbb{P}\{j \in \mathcal{S}(\hat{A}^w)\} \\
\mathbb{P}\{n+1 \in \mathcal{S}(\hat{A}^w)\} \cdot \sum_{j=1}^{n+1} \mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] &= \sum_{j=1}^{n+1} \mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] \cdot \mathbb{P}\{j \in \mathcal{S}(\hat{A}^w)\} \\
\mathbb{P}\{n+1 \in \mathcal{S}(\hat{A}^w)\} &= \sum_{j=1}^{n+1} \mathbb{E}_i\big[\tilde{w}_{i,j}(X_{n+1})\big] \cdot \mathbb{P}\{j \in \mathcal{S}(\hat{A}^w)\} \\
&= \mathbb{E}\Bigg[\sum_{j \in \mathcal{S}(\hat{A}^w)} \tilde{w}_{i,j}(X_{n+1})\Bigg] \\
&\leq 2\alpha
\end{aligned}$$

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

$$\begin{aligned} &Y_{n+1} > Q_{1-\alpha}\Big\{\sum_{j=1}^{n}\big[\tilde{w}_{n+1,j}(X_{n+1})\delta_{\widehat{\mu}_{-j}(X_{n+1})+R_j^{LOO}}\big] + \tilde{w}_{(n+1)^2}(X_{n+1})\delta_{\infty}\Big\} \\ &\qquad\implies \sum_{j=1}^{n}\tilde{w}_{n+1,j}(X_{n+1})\cdot \mathbb{1}\big\{Y_{n+1} > \widehat{\mu}_{-j}(X_{n+1}) + R_j^{LOO}\big\} \geq 1-\alpha \end{aligned}$$

or otherwise

$$\begin{aligned} &Y_{n+1} < Q_{\alpha}\Big\{\sum_{j=1}^{n}\big[\tilde{w}_{n+1,j}(X_{n+1})\delta_{\widehat{\mu}_{-j}(X_{n+1})+R_j^{LOO}}\big] + \tilde{w}_{(n+1)^2}(X_{n+1})\delta_{-\infty}\Big\} \\ &\qquad\implies \sum_{j=1}^{n}\tilde{w}_{n+1,j}(X_{n+1})\cdot \mathbb{1}\big\{Y_{n+1} < \widehat{\mu}_{-j}(X_{n+1}) - R_j^{LOO}\big\} \geq 1-\alpha \end{aligned}$$

And we can write the union of these two events as

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

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

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

A.3. Proof for JAW-KLOO coverage under feedback covariate shift

We first restate the theorem before proceeding with the proof.

Theorem 4.1 *Suppose data are generated under feedback covariate shift* (5) *and assume $\widetilde{P}_{X;D}$ is absolutely continuous with respect to P_X for all possible values of D. Then, for any miscoverage level, $\alpha \in (0,1)$, the JAW-KLOO predictive interval in* (10) *satisfies*

$$\mathbb{P}\{Y_{n+1} \in \widehat{C}^{\text{JAW-}K\text{LOO}}_{n,\alpha}(X_{n+1})\} \geq 1-2\alpha. \tag{23}$$

Theorem 4.1 follows from from Lemma A.2 and Theorem 3.1. With training data $Z_1, ..., Z_n$ and test point Z_{n+1} generated under feedback covariate shift (5), let $S_{\text{LOO}} \subseteq \{1, ..., n\}$ denote a subset of the training data where we retrain a leave-one-out model $\widehat{\mu}_{-j}$ for each $j \in S_{\text{LOO}}$. Then, by Lemma A.2 the random variables $\{Z_j : j \in S_{\text{LOO}}\} \cup \{Z_{n+1}\}$ are pseudo-exchangeable, generated under FCS. Assuming that the model-fitting algorithm $\mathcal{A}$ treats the data symmetrically, every leave-one-out model $\widehat{\mu}_{-j}$ for $j \in S_{\text{LOO}}$ is trained on the subset of data $\{Z_{j'} : j' \notin S_{\text{LOO}}\}$, so for the purpose of JAW-KLOO training on the data $\{Z_{j'} : j' \notin S_{\text{LOO}}\}$ can be considered a subroutine of the model-fitting algorithm that is invariant to the ordering of the remaining data for the fitting of each $\widehat{\mu}_{-j}$. Thus, treating JAW-KLOO as an instance of JAW-FCS where the former is given a smaller dataset S_{LOO} and a different model-fitting algorithm $\mathcal{A}_{-S_{\text{LOO}}}$ that depends on the data $\{1, ..., n\}\backslash S_{\text{LOO}}$ but that still treats data symmetrically, the JAW-KLOO coverage guarantee follows from the guarantee for JAW-FCS given in Theorem 3.1.

A.4. Proof of WCV+ coverage under feedback covariate shift

The proof for weighted cross validation+ under feedback covariate shift (WCV+FCS) coverage follows a similar structure as the proof for JAW-FCS coverage presented in Appendix A.2, so here we focus on the key differences.

The first two setup steps are identical to the corresponding setup steps in the proof for CV+ coverage, or Theorem 4, in Barber et al. (2021):

(a) We now suppose the hypothetical scenario where we have access to $n/K - 1$ additional test points, for a total of $m = n/K$ test points $\{(X_{n+1}, Y_{n+1}), ..., (X_{n+m}, Y_{n+m})\}$. We then partition the training data into sets $S_1, ..., S_K$ with m datapoints each and define $S_{K+1} = \{n+1, ..., n+m\}$ as the set of test points. For any pair of distinct partition indices $k, k' \in \{1, ..., K+1\}$ such that $k \neq k'$ we define $\tilde{\mu}_{-(S_k, S_{k'})}$ as the regression model fit with training and test data except with S_k and $S_{k'}$ removed (i.e., fit with $\{1, ..., n+m\} \backslash \{S_k \cup S_{k'}\}$).

(b) We then define the matrix of residuals $R \in \mathbb{R}^{(n+m)\times(n+m)}$ as follows, where $k(i)$ denotes the index of the partition that contains point i (so that for $i, j \in \{1, ..., n+m\}$ where $i \neq j$, $k(i) = k(j) \iff i, j \in S_k$):

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

In the next three setup steps we introduce changes to the proof for Theorem 4 in Barber et al. (2021) that are analogous to setup steps (c-e) in our proof for JAW-FCS coverage.

(c) As in the proof for JAW-FCS coverage, we define a more general version of the normalized FCS weights introduced in (12), as follows:

$$\tilde{w}^{CV}_{i,j}(X_{n+1}) = \frac{w(X_i; Z_{-\{S_{k(i)}, S_{k(j)}\}}) w(X_j; Z_{-\{S_{k(i)}, S_{k(j)}\}})}{\sum_{j' \in \{i\} \cup \{1, ..., n+m\} \backslash S_{k(i)}} \left[w(X_i; Z_{-\{S_{k(i)}, S_{k(j')}\}}) w(X_{j'}; Z_{-\{S_{k(i)}, S_{k(j')}\}}) \right]}. \tag{24}$$

where letting $i = n+1$ yields the weights introduced in the main paper for WCV+ (12).

(d) We define the weighted comparison matrix $A^w \in \{0, 1\}^{(n+m)\times(n+m)}$ analogously as with the JAW-FCS proof. That is, with A as the unweighted comparison matrix with entries $A_{ij} = \mathbb{1}\{R_{ij} > R_{ji}\}$, also define $\hat{W}^{CV}$ as the weight matrix with entries $\hat{W}^{CV}_{ij} = w(X_i; z_{-\{S_{k(i)}, S_{k(j)}\}})$. Then, define $\hat{A}^{wCV} = \hat{W}^{CV} \odot A \odot \hat{W}^{CV\top}$, with entries $\hat{A}^{wCV}_{ij} = w(X_i; z_{-\{S_{k(i)}, S_{k(j)}\}})$.

(e) We then define strange points analogously as in the JAW-FCS proof in several equivalent ways, to aid with intuition for different later steps in the proof

$$\begin{aligned} \mathcal{S}(\hat{A}^w) =& \Big\{ i \in [n+m] \ : \ \sum_{j=1}^{n+m} \Big(\tilde{w}^{CV}_{i,j}(X_{n+1}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\} \Big) \geq 1 - \alpha \Big\} \\ \mathcal{S}(\hat{A}^w) =& \Big\{ i \in [n+m] \ : \ \sum_{j=1}^{n+m} \Big(w(X_i; z_{-\{S_{k(i)}, S_{k(j)}\}}) w(X_j; z_{-\{S_{k(i)}, S_{k(j)}\}}) \cdot \mathbb{1}\{R_{ij} > R_{ji}\} \Big) \\ & \geq (1-\alpha) \sum_{j' \in \{i\} \cup \{1, ..., n+m\} \backslash S_{k(i)}} \left[w(X_i; z_{-\{S_{k(i)}, S_{k(j')}\}}) w(X_{j'}; z_{-\{S_{k(i)}, S_{k(j')}\}}) \right] \Big\} \\ \mathcal{S}(\hat{A}^w) =& \Big\{ i \in [n+m] \ : \ \sum_{j=1}^{n+m} \hat{A}^w_{ij} \geq (1-\alpha) \sum_{j' \in \{i\} \cup \{1, ..., n+m\} \backslash S_{k(i)}} (\hat{W}^{CV}_{ij'})^2 \Big\}. \end{aligned} \tag{25}$$

Each of the three above equivalent definitions of strange points for WCV+ serve a different illustrative purpose. The first definition illustrating the intuition of strange point being a point with abnormally large residuals relative to a sufficient weighted fraction of the other points. The second and third lines describe how the set of strange points can be computed from the weighted comparison matrix $\hat{A}^{wCV}$. With some abuse of notation to improve conciseness, hereon we will denote $\{i, -S_{k(i)}\} = \{i\} \cup \{\{1, ..., n+m\} \backslash S_{k(i)}\}$

Step 1: Bounding the expected normalized weight of strange points

Step 1 begins similarly as Step 1 in our JAW-FCS coverage proof, except we need to make several adjustments to account for the fact that points in the same fold do not play against each other in the "tournament". Additionally, this proof step generalizes the analogous step for exchangeable cross-validation+ coverage proof in Barber et al. (2021) by extending to covariate shift and by accounting for issues that arise when different cross-validation folds or a batch of test points have different likelihood ratio weights.

To describe the tournament setup for WCV+, for each pair of points $\{i, j\}$ in $\{1, ..., n+m\}$ where $i \neq j$, we here say that i "wins" its game against point j if $\hat{A}^w_{ij} > 0$ (note that this implies $k(i) \neq k(j)$ or that i and j play each other); we say that i loses its game with j if $k(i) \neq k(j)$ and $\hat{A}^w_{ij} = 0$; and if $k(i) = k(j)$ then i and j do not play each other in the tournament so neither point wins nor loses a game against the other.

To begin this proof step, we leverage the first definition of strange points given in (25). The definition (25) says that a strange point i wins its tournament games against other points that together amount to at least a total normalized weight of $1-\alpha$ (normalized as in (24)), which in turn implies that if i is a strange point, that i *loses* to other points that have total normalized weight *at most* $\alpha - \tilde{w}^w_{i,i}(X_i; z_{-S_{k(i)}})$ (since by our definition i does not play against nor lose to itself), which we can write as follows:

$$\begin{array}{c}\text{Total normalized weight of points}\\ \text{that } i \text{ plays against and loses to}\end{array} = \sum_{j=1}^{n+m} \left[\tilde{w}^{CV}_{ij}(X_{n+1}) \cdot \mathbb{1}\{R_{ij} \leq R_{ji}\} \right] \quad \leq \quad \alpha - \tilde{w}^{CV}_{ii},$$

Whereas the JAW-FCS coverage proof derives the inequality (20) by leveraging the observation that (in JAW-FCS's leave-one-out construction) a pair of strange points is also a pair of points where one point is strange and the other is a point that loses to the strange point, the same assumption is not true for the WCV+. In particular, in WCV+ a pair of strange points $\{i, j\}$ s.t. $i, j \in \mathcal{S}(A^w)$ might consist of points in the same fold, that is where $k(i) = k(j)$, which implies that i and j do not play against each other in the tournament, and thus there is no loser. We can thus define two types of pairs of strange points $\{i, j\}$ s.t. $i, j \in \mathcal{S}(A^w)$: one type where the strange points in the pair play against each other in the tournament (i.e., where $k(i) \neq k(j)$), and another type where the strange points in a pair do not play against each other in the tournament (i.e., where $k(i) = k(j)$). We can then proceed by obtaining a bound corresponding to each of the two types of strange point pairs before combining the two bounds.

By essentially the same arguments as in the JAW-FCS coverage proof to obtain (20), we can obtain an inequality that appears similar to (20) except with the summation over pairs of strange points (on the left) restricted only to points that play each other in the tournament as follows:

$$\sum_{i \in \mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1}) \cdot \frac{1}{2} \sum_{j \in \mathcal{S}(\hat{A}^w) \setminus i,\ k(i) \neq k(j)} \tilde{w}_{i,j}(X_{n+1}) \Big) \leq \sum_{i \in \mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1}) \cdot \big(\alpha - \tilde{w}_{i,i}(X_{n+1}) \big) \Big). \tag{26}$$

Taking the expectation with respect to i of $\sum_{j \in \mathcal{S}(\hat{A}^w) \setminus i,\ k(i) \neq k(j)} \tilde{w}_{i,j}(X_{n+1})$ on the left-hand side and of $\big(\alpha - \tilde{w}_{i,i}(X_{n+1})\big)$ on the right-hand side and factoring out the expectations, we obtain

$$\sum_{i \in \mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1}) \cdot \frac{1}{2} \mathbb{E}_i \Big[\sum_{j \in \mathcal{S}(\hat{A}^w) \setminus i,\ k(i) \neq k(j)} \tilde{w}_{i,j}(X_{n+1}) \Big] \Big) \leq \sum_{i \in \mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1}) \cdot \mathbb{E}_i[\alpha - \tilde{w}_{i,i}(X_{n+1})] \Big)$$

$$\frac{1}{2} \mathbb{E}_i \Big[\sum_{j \in \mathcal{S}(\hat{A}^w) \setminus i,\ k(i) \neq k(j)} \tilde{w}_{i,j}(X_{n+1}) \Big] \sum_{i \in \mathcal{S}(\hat{A}^w)} \tilde{w}_{i,i}(X_{n+1}) \leq \mathbb{E}_i[\alpha - \tilde{w}_{i,i}(X_{n+1})] \sum_{i \in \mathcal{S}(\hat{A}^w)} \Big(\tilde{w}_{i,i}(X_{n+1}) \Big) \tag{27}$$

However, we also want a similar inequality for pairs of strange points that are in the same fold and thus do not play each other in the tournament. Accordingly, we examine a term analogous to the left-hand side of (26) but for pairs of points in the same fold—i.e., the following term is the sum, over all strange point pairs in the same fold and over all $K+1$ folds, of the product of $\tilde{w}_{ii}^{CV}(X_{n+1})$ with the weights $\tilde{w}_{ij}^{CV}(X_{n+1})$ of all the strange points j in the same fold as i:

$$\frac{1}{2}\sum_{k=1}^{K+1}\sum_{i\in S_{k(i)}\cap\mathcal{S}(A^w)}\tilde{w}_{ii}^{CV}(X_{n+1})\sum_{j\in S_{k(i)}\cap\mathcal{S}(A^w),j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big). \tag{28}$$

Then, taking the expectation of the summation term $\sum_{j\in S_{k(i)}\cap\mathcal{S}(A^w),j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)$ with respect to X_i (where as in the JAW-FCS proof we use condensed notation $\mathbb{E}_i[\cdot]=\mathbb{E}_{X_i\sim P_{X_i}}[\cdot]$ to denote the expectation with respect to the random variable with index i):

$$\begin{aligned}
&\frac{1}{2}\sum_{k=1}^{K+1}\sum_{i\in S_{k(i)}\cap\mathcal{S}(A^w)}\tilde{w}_{ii}^{CV}(X_{n+1})\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)}\cap\mathcal{S}(A^w),j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\\
&\quad=\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)}\cap\mathcal{S}(A^w),j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\sum_{k=1}^{K+1}\sum_{i\in S_{k(i)}\cap\mathcal{S}(A^w)}\tilde{w}_{ii}^{CV}(X_{n+1})\\
&\quad=\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)}\cap\mathcal{S}(A^w),j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\sum_{i\in\mathcal{S}(A^w)}\tilde{w}_{ii}^{CV}(X_{n+1})\\
&\quad\leq\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\sum_{i\in\mathcal{S}(A^w)}\tilde{w}_{ii}^{CV}(X_{n+1})
\end{aligned}\tag{29}$$

While (27) provides a bound corresponding to pairs of strange points in different folds, (29) provides a bound corresponding to pairs of points in the same fold. We can then combine the two results to obtain

$$\begin{aligned}
\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i}\tilde{w}_{i,j}(X_{n+1})\Bigg]\sum_{i\in\mathcal{S}(\hat{A}^w)}\tilde{w}_{i,i}(X_{n+1})&\leq\mathbb{E}_i[\alpha-\tilde{w}_{i,i}(X_{n+1})]\sum_{i\in\mathcal{S}(\hat{A}^w)}\Big(\tilde{w}_{i,i}(X_{n+1})\Big)\\
&\quad+\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\sum_{i\in\mathcal{S}(A^w)}\tilde{w}_{ii}^{CV}(X_{n+1})\\
\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i}\tilde{w}_{i,j}(X_{n+1})\Bigg]&\leq\mathbb{E}_i[\alpha-\tilde{w}_{i,i}(X_{n+1})]+\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\\
\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in\mathcal{S}(\hat{A}^w)\setminus i}\tilde{w}_{i,j}(X_{n+1})\Bigg]+\frac{1}{2}\mathbb{E}_i[\tilde{w}_{i,i}(X_{n+1})]&\leq\alpha-\frac{1}{2}\mathbb{E}_i[\tilde{w}_{i,i}(X_{n+1})]+\frac{1}{2}\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\\
\mathbb{E}_i\Bigg[\sum_{j\in\mathcal{S}(\hat{A}^w)}\tilde{w}_{i,j}(X_{n+1})\Bigg]&\leq2\alpha-\mathbb{E}_i[\tilde{w}_{i,i}(X_{n+1})]+\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]\\
\mathbb{E}_i\Bigg[\sum_{j\in\mathcal{S}(\hat{A}^w)}\tilde{w}_{i,j}(X_{n+1})\Bigg]&\leq2\alpha+\mathbb{E}_i\Bigg[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)\Bigg]
\end{aligned}\tag{30}$$

Step 2: Pseudo exchangeability of the datapoints. We now leverage the pseudo exchangeability of the data to show that, since the expected total weight of the strange points is at most $2\alpha+\mathbb{E}_i[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)]$, that a test point has at most $2\alpha+\mathbb{E}_i[\sum_{j\in S_{k(i)},j\neq i}\Big(\tilde{w}_{ij}^{CV}(X_{n+1})\Big)]$ probability of being strange.

This step proceeds similarly as the analogous step 2 in the JAW-FCS proof, except with a restriction on the permutation matrix P_π to maintain the fold structure of the data. That is, for any $(n+m)\times(n+m)$ permutation matrix P_π where

$i \sim j$ if $k(i) = k(j)$, through a similar argument as in the JAW-FCS proof we can show that $\overset{\mathrm{d}}{=} P_\pi \hat{A}^w P_\pi^\top$ for any such permutation matrix P_π. When combined with (30), the result for this step then follows.

Step 3: Connection to weighted CV+: The last proof step proceeds similarly as the third main step in the JAW-FCS proof. Through the same procedure, we establish that

$$\therefore\ \mathbb{P}\{Y_{n+1} \in \widehat{C}^{\text{JAW}}_{n,\alpha}(X_{n+1})\} \geq 1 - 2\alpha - \mathbb{E}_i\Bigg[\sum_{j \in S_{k(i)}, j \neq i} \Big(\tilde{w}^{CV}_{ij}(X_{n+1})\Big)\Bigg] \tag{31}$$

B. Supplementary experimental results.

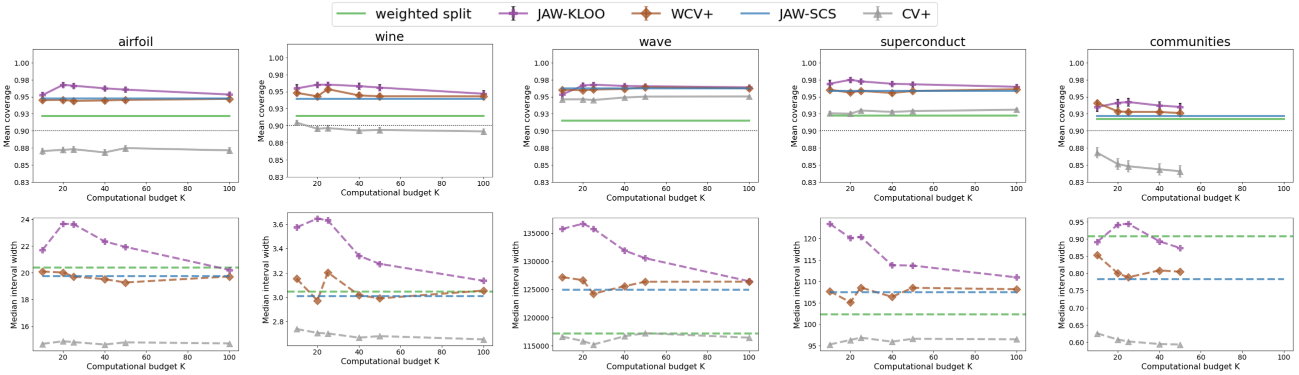

Figure 5. Supplementary SCS mean coverage (top row) and median interval widths (bottom row) results for neural network predictor. Coverage and interval widths for JAW-KLOO (violet plus shape) and WCV+ (brown diamond) methods under standard covariate shift are compared to JAW-SCS (blue line), weighted split (green line), and standard CV+ (gray triangles) baselines, for a range of computational budgets K. Results for five UCI datasets with a random forest predictor are reported, across 75 repeated experiments each corresponding to different 200-sample training datasets, with 200 test points per experiment.