Practical Adversarial Multivalid Conformal Prediction

Anonymous Author(s)
Affiliation
Address
email

Abstract

We give a simple, generic conformal prediction method for sequential prediction that achieves target empirical coverage guarantees on adversarial data. It is computationally lightweight — comparable to split conformal prediction — but does not require having a held-out validation set, and so all data can be used for training models from which to derive a conformal score. Furthermore, it gives stronger than marginal coverage guarantees in two ways. First, it gives *threshold calibrated* prediction sets that have correct empirical coverage even conditional on the threshold used to form the prediction set from the conformal score. Second, the user can specify an arbitrary collection of subsets of the feature space — possibly intersecting — and the coverage guarantees will also hold conditional on membership in each of these subsets. We call our algorithm MVP, short for MultiValid Prediction. We give both theory and an extensive set of empirical evaluations.

1 Introduction

Consider the problem of predicting labels $y \in \mathcal{Y}$ given examples $x \in \mathcal{X}$. One popular strategy for expressing uncertainty is to allow the algorithm to produce a *prediction set* $\mathcal{T} \subseteq \mathcal{Y}$ rather than an individual label. We give a simple, practical algorithm for constructing prediction sets in sequential prediction problems over an arbitrary domain $\mathcal{X} \times \mathcal{Y}$, given any data-dependent sequence of *conformal score functions* $s_t : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$. In each round t, an example represented by a feature vector $x_t \in \mathcal{X}$ arrives. We can define an arbitrary conformal score $s_t : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ that can depend on previously observed examples (as well as on f) in arbitrary ways. We produce a round-dependent threshold q_t, which gives us a prediction set $\mathcal{T}_t = \{y \in \mathcal{Y} : s_t(x_t, y) \leq q_t\}$. We then observe the true label y_t; we say our prediction set *covers* y_t if $y_t \in S_t$. Given a coverage target $1 - \delta$, our goal is to produce sets that have correct empirical coverage — i.e. that cover a $1 - \delta$ fraction of the labels (we do not want either over-coverage or under-coverage). We wish to make as few assumptions as possible, so that our method is *robust* to arbitrary and unanticipated distribution shift, and applies to e.g. time series data which are very far from exchangeable. We also want our coverage guarantees to be meaningful not just marginally, but at finer granularities: *conditional* on both the threshold value we choose, and on membership of $x_t \in G$ for a set of groups $G \in \mathcal{G} \subseteq 2^{\mathcal{X}}$ that can be arbitrarily defined and intersecting. Finally, we want our algorithm to have low computational overhead, so that it can be applied as a wrapper on top of arbitrary prediction methods, for both regression and classification. The algorithm we propose achieves these goals and has several desirable properties:

Worst-Case Empirical Coverage: Our method has worst case *adversarial* guarantees. The sequence of examples $\{(x_t, y_t)\}_{t=1}^{T}$ needs not be drawn from an exchangeable distribution as it does for standard conformal prediction methods [Shafer and Vovk, 2008] — instead, it can be chosen by an adaptive adversary. The conformal scores s_t can be arbitrary and can depend on data from previous rounds (e.g. they can be derived from models trained on all past data, and so there is no need to separate data into training and calibration sets as in split conformal prediction [Lei et al., 2018]). Thus, our method can tolerate time series data, and arbitrary and unanticipated distribution shifts.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Calibrated, Multivalid Coverage: Our prediction sets obtain their target empirical coverage level not just marginally, but in a *threshold-calibrated* fashion: for every threshold q, they approach $1-\delta$ coverage on the subsequence of rounds t on which the threshold $q_t = q$[1]. We also promise group-conditional coverage. We can specify an arbitrary collection of *groups* $\mathcal{G}$, each group $G \in \mathcal{G}$ representing an arbitrary feature-space subset: $G \subseteq \mathcal{X}$. These groups can intersect in arbitrary ways. For example, $\mathcal{G}$ can be a collection of demographic groups based on race, age, income, or medical history, and each data point $x \in \mathcal{X}$ could represent a member of an arbitrary subset of these groups. Our method promises that simultaneously for every group $G \in \mathcal{G}$, on the subsequence of rounds t for which $x_t \in G$, our intervals obtain the target empirical coverage rate (again, in a calibrated fashion).

Computationally Lightweight: Our method is computationally lightweight. At each round t, it relies on simple arithmetic calculations involving the historical empirical coverage rates of finitely many candidate thresholds q_t on groups G such that $x_t \in G$. Hence it is comparable in cost to fast split conformal prediction methods [Lei et al., 2018] despite its ability to use all data for model training. We give an implementation of our algorithm and an extensive empirical evaluation. By contrast, prior work by Gupta et al. [2022] that gets comparable theoretical guarantees (in the special case of regression prediction intervals) does not give a practical algorithm: their approach uses the Ellipsoid method with a separation oracle to solve exponentially-sized linear programs at all rounds.

Nearly Statistically Optimal Rates: For each threshold q and group G, we promise $1-\delta$ empirical coverage over the rounds t where $x_t \in G$ and the predicted threshold $q_t = q$. Let $n^{G,q}$ denote the total number of such rounds. Even if the labels y_t were drawn from a known distribution and our coverage probability was exactly $1-\delta$, we would expect our *empirical* coverage on these $n^{G,q}$ rounds to deviate from $1-\delta$ by a $\pm 1/\sqrt{n^{G,q}}$ term. The prior theoretical coverage guarantees for each q and G of Gupta et al. [2022] differ from their target by $\tilde{O}(\sqrt{T}/n^{G,q})$, which is nearly optimal if $n^{G,q} \geq \Omega(T)$ but sub-optimal otherwise. Our algorithm promises empirical coverage rates for each pair (G, q) that differ from the target by an optimal $\tilde{O}(1/\sqrt{n^{q,G}})$ term.

We give an extensive experimental evaluation of our algorithm in a number of settings, and compare to vanilla split conformal prediction [Lei et al., 2018], as well as prior work that handles limited forms of known distribution shift [Tibshirani et al., 2019], produces conservative groupwise coverage [Foygel Barber et al., 2020], and gives adversarial (but uncalibrated) coverage guarantees [Gibbs and Candes, 2021]. In each setting, we show that our algorithm is competitive with previous work "on their turf" (i.e. in settings for which their assumptions are satisfied and we use their evaluation metrics). We then go on to show that our method gives substantial improvements when either the setting or the evaluation metric becomes more difficult — e.g. when the distribution shift is unanticipated, when we measure group-wise rather than just marginal coverage, or when the data comes in adversarial ordering. In some cases we improve on standard techniques even in standard "benign" settings: for example, we improve on split conformal prediction in an online linear regression setting with iid. data when the evaluation metric is just marginal coverage, but the regression function has to be learned from the same stream of data used to calibrate the prediction intervals. This is because split conformal prediction requires using separate splits of the data for training the regression and calibrating the prediction intervals to maintain exchangeability of the conformal scores — but we are able to use all of the data for both tasks.

1.1 Additional Related Work

See Angelopoulos and Bates [2021] for an excellent recent survey of conformal prediction. The weaknesses of these methods that we seek to address — namely, that in the worst case they provide only marginal coverage, and that they rely on strong distributional assumptions (typically *exchangeability*) — have been noted before. For example, Romano et al. [2020a] note that marginal coverage guarantees are undesirable and give group conditional guarantees for *disjoint* groups by calibrating separately on

[1] Calibration is especially important in a distribution free setting, when coverage is measured empirically. If we only asked for the target marginal empirical coverage as Gibbs and Candes [2021] do, rather than for threshold-calibrated prediction sets, one could obtain the right coverage rate by "cheating" in the following uninformative way: predict the full label set $S_t = \mathcal{Y}$ on a $1-\delta$ fraction of rounds (guaranteeing coverage), and the empty set $S_t = \emptyset$ on the remaining δ fraction of rounds (guaranteeing miscoverage of the label). This obtains empirical coverage rate $1-\delta \pm O(1/T)$ marginally, but not conditional on the prediction sets chosen.

each group. This fails when the groups can intersect. Foygel Barber et al. [2020] provide guarantees that are valid conditional on membership in intersecting subgroups $\mathcal{G}$. They take a conservative approach, by computing prediction sets separately for each group and then taking the union of all these sets over the groups of a new individual. As a result, their prediction sets are conservative and do not approach the target coverage level. These results both require exchangeable data. Chernozhukov et al. [2018] obtain approximate marginal coverage guarantees for non-exchangeable time series data coming from a rapidly mixing process. Tibshirani et al. [2019] study conformal prediction under *covariate shift*. They adapt conformal techniques to handle the case when both the point of distribution shift and the likelihood ratio between the training and test distribution are known. Gibbs and Candes [2021] give a method that can guarantee target marginal coverage without any assumptions on the data generating process. In contrast, our prediction sets promise not just marginal coverage, but are "threshold-calibrated" and hold also conditional on membership in arbitrary sub-groups.

Following a recent resurgence in interest in conformal techniques, a number of papers have proposed conformal scores that have desirable properties [Hoff, 2021, Romano et al., 2019, Angelopoulos et al., 2020, Romano et al., 2020b, Park et al., 2019]. Our work is complementary to this line of work: just like traditional methods of conformal prediction, we too take as input arbitrary conformal scores. Thus we can adopt any of these conformal score functions and inherit their properties, while providing the stronger worst-case guarantees of multivalid coverage.

For the special case of prediction intervals, the type of *multi-valid* prediction that we study was first defined in Jung et al. [2021], who gave a way of obtaining it in the *batch* setting for i.i.d. data, via producing *multicalibrated* estimates of label variances and higher moments. Gupta et al. [2022] proved that there exists an online prediction algorithm that gives the sort of multi-valid prediction intervals that we consider in this work. The algorithm we give in this paper is both much more efficient (theirs was not implementable) and has substantially better (optimal) convergence bounds.

Finally, multivalidity is related to subgroup fairness [Kearns et al., 2018, 2019, Kim et al., 2018] that ask for statistical "fairness" constraints to hold across all subgroups defined by some rich class $\mathcal{G}$. In particular, it is closely related to multicalibration [Hébert-Johnson et al., 2018].

2 Online Uncertainty Quantification: Setting and Notation

We let $\mathcal{X}$ denote a feature domain and $\mathcal{Y}$ a label domain. $\mathcal{G} \subseteq 2^{\mathcal{X}}$ denotes a collection of subsets of $\mathcal{X}$, which we call "groups". For $x \in \mathcal{X}$, $\mathcal{G}(x) = \{G \in \mathcal{G} : x \in G\}$ is the set of groups that contain x. For any integer $T > 0$, $[T] = \{1, \ldots, T\}$. The probability simplex over a finite set A is denoted ΔA.

Our uncertainty quantification is based on a bounded conformal score function $s_t : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ which can change in arbitrary ways between rounds $t \in [T]$. Without loss of generality, we assume that the scoring function takes values in the unit interval: $s_t(x, y) \in [0, 1]$ for any $x \in \mathcal{X}, y \in \mathcal{Y}$, and $t \in [T]$. Fix some target coverage rate $1 - \delta$. Ideally, the learner wants to produce prediction sets $\mathcal{T}_t(x_t) = \{y \in \mathcal{Y} : s_t(x_t, y) \leq q_t\}$ that cover the true label y with probability $1 - \delta$ over the randomness of the unknown label distribution: $\Pr_{y|x_t}[y \in \mathcal{T}_t(x_t)] \approx 1 - \delta$. This is equivalent to choosing a conformity threshold q_t such that $\Pr_{y|x_t}[s_t(x_t, y) \leq q_t] \approx 1 - \delta$.

We want to model an adversary that can choose an arbitrary sequence of examples x_t and labels y_t. However, because the adversary may choose the label distribution with knowledge of the conformal score function, we will elide the particulars of the conformal score function and the distribution on labels y_t in our derivation, and instead equivalently imagine the adversary as directly choosing x_t and a distribution over conformal scores s_t conditional on x_t (representing the distribution over conformal scores $s_t(x_t, y_t)$). We may thus without loss view the interaction in the following simplified form:

1. The *adversary* chooses a joint distribution over feature vector $x_t \in \mathcal{X}$ and conformal score $s_t \in [0, 1]$. The learner receives x_t (a realized feature vector), but no information about s_t.
2. The learner produces a conformity threshold q_t.
3. The learner observes the realized conformal score s_t.

For any round $t \in [T]$, we write $\pi_t = (x_t, s_t, q_t)$ to denote the realized outcomes in round t and similarly write $\pi_{t':t}$ for the *transcript* of the interaction between rounds $t' \leq \tau \leq t$: $\pi_{t':t} = ((x_\tau, s_\tau, q_\tau))_{\tau=t'}^{t}$. To denote the extension of a transcript by a single round or the concatenation of two transcripts, we use $\oplus$: for example, we may write $\pi_{1:t} = \pi_{1:t-1} \oplus \pi_t$ and $\pi_{1:T} = \pi_{1:t} \oplus \pi_{t+1:T}$.

We write $\Pi^* = (\mathcal{X} \times [0,1] \times [0,1])^*$ as the domain of all transcripts. Fixing a learner and an adversary induces a probability distribution over transcripts. Our goal is to derive algorithms with guarantees that hold over the transcript randomness, in the worst-case over all possible adversaries.

Given a transcript $\pi_{1:T}$, a group $G \in \mathcal{G}$ and a subset of rounds $S \subseteq [T]$, $G_S = \{t \in S : x_t \in G\}$ denotes the set of rounds in S in which the realized feature vectors in the transcript belonged to G. Overloading this notation, for group $G \in \mathcal{G}$ and period $t \leq T$, we write $G_t = \{t' \in [t] : x_{t'} \in G\}$ for the set of rounds in a transcript $\pi_{1:t}$ in which the feature vectors belonged to the group G.

Given conformity threshold q, we say it *covers* conformal score s if $\mathrm{Cover}(q, s) = 1$, where we define: $\mathrm{Cover}(q, s) = \mathbb{1}[s \leq q]$. To define threshold calibration, we bucket our thresholds using a discretization parameter m. For any m and bucket index $i \in [m-1]$, we write $B_m(i) = \left[\frac{i-1}{m}, \frac{i}{m}\right)$ and $B_m(m) = \left[\frac{n-1}{m}, 1\right]$ so that these buckets evenly partition the unit interval $[0,1]$[2]. Conversely, given a threshold $q \in [0,1]$, we write $B_m^{-1}(q) \in [m]$ for the index of the bucket that w belongs to: $B_m^{-1}(q) = i$, for i such that $q \in B_m(i)$. When clear from the context, we elide the subscript n and write $B(i)$ and $B^{-1}(q)$.

For any $S \subseteq [T]$ and $i \in [n]$, we define $S(i)$ to be the subset of rounds in S in which the learner's threshold falls in bucket i. Formally: $S(i) = \{t \in S : q_t \in B_m(i)\}$.

We can now define our main objective in this paper: threshold calibrated multivalid coverage.

Definition 2.1 (Threshold Calibrated Multivalid Coverage)**.** *Fix a coverage target $1 - \delta$ and a collection of groups $\mathcal{G} \subset 2^{\mathcal{X}}$. Given a transcript $\pi_{1:T}$, a sequence of conformity thresholds $(q_t)_{t=1}^T$ is said to be* (α, m)-multivalid with respect to δ and $\mathcal{G}$ *for some function $\alpha : \mathbb{N} \to \mathbb{R}$, if for every $i \in [m]$ and $G \in \mathcal{G}$, the following holds true:*

$$\left| \frac{1}{|G_T(i)|} \sum_{t \in G_T(i)} \mathrm{Cover}\left(q_t, s_t\right) - (1 - \delta) \right| \leq \alpha(|G_T(i)|)$$

Note that multivalid coverage is defined by a *function* $\alpha(\cdot)$ of the length of the sequence on which empirical coverage is computed. This lets us give fine-grained bounds that scale with the length of this sequence. Throughout this paper we use the following family of functions, parameterized by a constant $\epsilon > 0$:

$$\alpha(n) = \frac{f(n)}{n}, \quad \text{where} \quad f(n) = \sqrt{(n+1)\log^{1+\epsilon}(n+2)}$$

A useful fact is that the series $\frac{1}{f(n)^2}$ is convergent: $\sum_{n=0}^{\infty} \frac{1}{f(n)^2} = K_\epsilon$ where K_ϵ is a constant depending only on our choice of ϵ that will appear in our bounds.

3 Our Algorithm and Analysis

Before we provide the algorithm and its guarantees, we first discuss a needed assumption. Observe that even in the easier setting where the conformal score s is drawn from a fixed, known distribution: $s \sim \mathcal{D}$ — there may not be any threshold $q \in [0,1]$ that satisfies the desired target coverage value, i.e. that guarantees that $|\mathbb{E}_{s \sim \mathcal{D}}[\mathrm{Cover}(q, s) - (1 - \delta)]|$ is small. Consider for example a distribution that places all its mass on a single value s. Then any threshold q covers the s with probability 1 or probability 0, which for $\delta \notin \{0, 1\}$ is bounded away from our target coverage probability. One could randomize the threshold to get the target marginal coverage rate, but this corresponds to the "cheating" strategy we outline in Footnote 1, and in particular would not satisfy our notion of *threshold calibrated* coverage. Of course, if achieving the target coverage is impossible in the easier distributional setting, then it is also impossible in the more challenging online adversarial setting.

With this in mind, just as with many other approaches to conformal prediction that aim to converge to the correct coverage rate (rather than conservatively over-cover), we will need to assume that our target distributions are not too concentrated on any single point. Following Gupta et al. [2022], we define a class of smooth distributions for which achieving (approximately) the target coverage is always possible for some threshold q defined over an appropriately finely discretized range. Our smoothness condition makes sense even for discrete distributions, so we do not need to assume continuity. To denote the uniform grid on $[0,1]$, we write $\mathcal{P}^{rn} = \left\{0, \frac{1}{rm}, \frac{2}{rm}, \ldots, 1\right\}$.

[2] We can handle non-uniform discretizations of the unit interval as well, with no additional complications.

We show that we can achieve (approximately) our target coverage goals in the online adversarial setting when the adversary is constrained to playing smooth distributions, which are distributions that do not put too much probability mass on any sufficiently small sub-interval.

Definition 3.1. *A distribution $Q \in \Delta([0,1])$ is (ρ, rm)-*smooth *if*

$$\Pr_{s\sim Q}[s \in [a,b]] \leq \rho \quad \textit{for any subinterval } [a,b] \subseteq [0,1] \textit{ of length} \leq \tfrac{1}{rm},$$

*A joint distribution $\mathcal{D} \in \Delta(\mathcal{X} \times [0,1])$ is (ρ, rm)-*smooth *if for every $x \in \mathcal{X}$, the marginal conformal score conditional on x, $\mathcal{D}|_x$, is (ρ, rm)-smooth. An adversary is (ρ, rm)-smooth if the joint distribution over (x_t, s_t) is (ρ, rm)-smooth at every round $t \in [T]$.*

Remark 3.1. *For any ρ, the assumption of (ρ, rm)-smoothness becomes more mild as $r \to \infty$. For us, r will be a nuisance parameter that we can take as large as we want — we will not have to pay for it either in our running time or our coverage bounds. We can also* algorithmically enforce *smoothness by perturbing the conformal scores with small amounts of noise from any continuous distribution, and so we should think of smoothness as a mild assumption. Our experiments bear this out.*

We now present the algorithm (MVP — MultiValid Predictor) along with its guarantees and provide the analysis in Section 3.1. It resembles the algorithm for online mean multicalibration given in Gupta et al. [2022], which in turn is a multi-group generalization of the "almost deterministic" calibration algorithm of Foster and Hart [2021].

Algorithm 1: MVP(η, n, r)

for $t = 1, \ldots, T$ **do**

Take as input an arbitrary conformal score $s_t : \mathcal{X} \times \mathcal{Y} \to [0,1]$.

Observe x_t, and for each $i \in [m]$ and $G \in \mathcal{G}(x_t)$, compute:

$$n_{t-1}^{G,i} = \sum_{\tau=1}^{t-1} \mathbb{1}[q_\tau \in B(i),\ x_\tau \in G] \qquad \text{Definition 3.2}$$

$$V_{t-1}^{G,i} = \sum_{\tau=1}^{t-1} \mathbb{1}[x_\tau \in G, q_\tau \in B_m(i)] \cdot (\mathrm{Cover}(q_\tau, s_\tau) - (1-\delta)) \qquad \text{Definition 3.3}$$

$$C_{t-1}^{i}(x_t) = \sum_{G \in \mathcal{G}(x_t)} \tfrac{1}{f(n_t^{G,i})} \left(\exp\left(\eta \tfrac{V_t^{G,i}}{f(n_t^{G,i})}\right) - \exp\left(-\eta \tfrac{V_t^{G,i}}{f(n_t^{G,i})}\right)\right). \qquad \text{From Lemma 3.1}$$

if $C_{t-1}^{i}(x_t) > 0$ for all $i \in [m]$ **then**

Choose threshold $q_t = 0$.

else if $C_{t-1}^{i}(x_t) < 0$ for all $i \in [m]$ **then**

Choose threshold $q_t = 1$.

else

Find $i^* \in [m-1]$ such that $C_{t-1}^{i^*}(x_t) \cdot C_{t-1}^{i^*+1}(x_t) \leq 0$.

Define $0 \leq p_t \leq 1$ as follows:[a] $p_t = \left|C_{t-1}^{i^*+1}(x_t)\right| / \left(\left|C_{t-1}^{i^*+1}(x_t)\right| + \left|C_{t-1}^{i^*}(x_t)\right|\right)$.

Choose threshold $q_t = \frac{i^*}{m} - \frac{1}{rm}$ with probability p_t and $q_t = \frac{i^*}{m}$ with probability $1 - p_t$.

Output prediction set $\mathcal{T}_t(x_t) = \{y \in \mathcal{Y} : s_t(x_t, y) \leq q_t\}$.

[a] Using the convention that $0/0 = 0$.

Theorem 3.1. *Set $\eta = \sqrt{\frac{\ln(|\mathcal{G}|m)}{2K_\epsilon|\mathcal{G}|m}}$. Against any (ρ, rm)-smooth adversary and for any adaptively chosen sequence of conformal scores s_t, MVP (Algorithm 1) produces a sequence of $(q_t)_{t=1}^T$ that is $(c_{exp}\alpha(\cdot), m)$-multivalid in expectation over the randomness of $\pi_{1:T}$ with respect to δ and $\mathcal{G}$, where*

$$c_{exp} \leq \sqrt{4K_\epsilon|\mathcal{G}|m\ln(|\mathcal{G}|m)} + \rho T.$$

Remark 3.2. *Since we can take r to be arbitrarily large, for any continuous distribution we can drive the ρT term to zero. Thus this bound establishes nearly statistically optimal convergence rates for constant $|\mathcal{G}|$ and m. Using a simpler analysis analogous to that of Gupta et al. [2022] for mean multicalibration, it is also possible to establish (α, m)-multivalidity with $\alpha(n) = \tilde{O}(\sqrt{T\log(|\mathcal{G}|m)}/n + \rho)$, which is optimal in $|\mathcal{G}|$ and m but has a bad dependence on T. We believe that our sub-optimal dependence on $|\mathcal{G}|$ and m is an artifact of our analysis, and not a property of our algorithm.*

3.1 Analysis

Omitted proofs from this section are presented in full detail in Appendix A. The analysis in this section can be seen as an extension of the surrogate loss argument developed in Gupta et al. [2022] for the problem of mean multicalibration. There are two main novel insights that lead to our algorithm and analysis for multivalid coverage. First, Gupta et al. [2022] were unable to extend their simple multicalibration algorithm to prediction interval multivalidity (and instead analyzed an impractical Ellipsoid-based algorithm). Informally this is because they parameterized prediction intervals with two parameters (the lower and upper endpoint), which eliminated the simple one-dimensional structure they were able to exploit for mean multicalibration. In contrast, our prediction intervals are parameterized by a single parameter q, which allows us to exploit a simple one-dimensional structure.

Second, the bounds in Gupta et al. [2022] uniformly bound the coverage error for each group $G \in \mathcal{G}$ and bucket $i \in [m]$ by $\tilde{O}(\sqrt{T})$, which is optimal only for subsequences that have $n_T^{G,i} = \Omega(T)$. In contrast, we obtain non-uniform bounds that depend on $n_T^{G,i}$ but not on T, and (at least for constant m and $|\mathcal{G}|$) have the optimal $\sqrt{n_T^{G,i}}$ dependence. We achieve this by analyzing a modified surrogate loss, leading to a significant amount of added complexity which accounts for the bulk of our argument.

Sketch of proof of Theorem 3.1. For each group $G \in \mathcal{G}$, bucket $i \in [m]$ time $t \in [T]$, we'd like to bound the coverage error on the subsequence of rounds τ in which $x_\tau \in G$ and $q_\tau \in B_m(i)$ in terms of the length of that sequence. We give the following notation for these sequence lengths:

Definition 3.2 (Group-bucket size)**.** *Given a transcript $\pi_{1:t} = ((x_\tau, s_\tau, q_\tau))_{\tau=1}^t$, we define the size for a group $G \in \mathcal{G}$ and a bucket $i \in [m]$ at time t to be:* $n_t^{G,i}(\pi_{1:t}) = \sum_{\tau=1}^{t} \mathbb{1}[q_\tau \in B(i),\ x_\tau \in G]$.

Similarly, for each $G \in \mathcal{G}$, $i \in [m]$ and time $t \in [T]$, we can define the (un-normalized) coverage error on the sequence corresponding rounds $\tau \leq t$ such that $x_\tau \in G$ and $q_\tau \in [m]$:

Definition 3.3. *Given transcript $\pi_{1:t}$, the coverage error for group $G \in \mathcal{G}$ and bucket $i \in [n]$ at time t is given as:* $V_t^{G,i} = \sum_{\tau=1}^{t} \mathbb{1}[x_\tau \in G, q_\tau \in B_m(i)] \cdot v_\delta(q_\tau, s_\tau)$, *where* $v_\delta(q, s) = \mathrm{Cover}(q, s) - (1-\delta)$.

Note that $V_t^{G,i}$ just records the deviation of the empirical coverage from its target $(1-\delta)$ on the subsequence of rounds τ in which $x_\tau \in G$ and $q_t \in B_m(i)$: it takes a positive value if we have *over-covered* on this subsequence and a negative value if we have under-covered.

Observation 3.1. *Fix a transcript $\pi_{1:T}$. If for all $G \in \mathcal{G}$ and $i \in [m]$, we have for some constant c: $\left|V_T^{G,i}\right| \leq c f(n_T^{G,i})$, then the resulting sequence of thresholds $(q_t)_{t=1}^T$ is $(c\alpha(\cdot), m)$-multivalid with respect to δ and $\mathcal{G}$.*

To bound the maximum of our normalized absolute coverage errors across all groups and buckets, i.e. $\max_{G \in \mathcal{G}, i \in [m]} \frac{|V_T^{G,i}|}{f(n_T^{G,i})}$, we use the following surrogate loss:

Definition 3.4 (Surrogate loss)**.** *Fix a transcript $\pi_{1:t} \in \Pi^*$ and a parameter $\eta \in (0, 1/2)$. Define a surrogate coverage loss function at day t for bucket $i \in [n]$ and group $G \in \mathcal{G}$ as*

$$L_t^{G,i}(\pi_{1:t}) = \exp\left(\eta \frac{V_t^{G,i}}{f(n_s^{G,i})}\right) + \exp\left(-\eta \frac{V_t^{G,i}}{f(n_s^{G,i})}\right),$$

where $V_t^{G,i}$ are implicitly functions of $\pi_{1:t}$. The overall surrogate coverage loss function is defined as $L_t(\pi_{1:t}) = \sum_{G \in \mathcal{G}, i \in [m]} L_t^{G,i}(\pi_{1:t})$.

We first show that the increase in the surrogate loss can be bounded in the following way:

Lemma 3.1. *Fix $\eta \in (0, \frac{1}{2})$ and a transcript $\pi_{1:t-1}$ for some t. For any $\pi_t = (q_t, x_t, s_t)$, we have*

$$L_t(\pi_{1:t-1} \oplus \pi_t) - L_{t-1}(\pi_{1:t-1}) \leq \sum_{(G,i) \in A_t(\pi_t)} \eta v_\delta(q_t, s_t) C_{t-1}^{G,i} + \frac{2\eta^2}{f(n_t^{G,i})^2} L_{t-1}^{G,i}(\pi_{t-1}),$$

where $A_t(\pi_t) = \{(G,i) : G \in \mathcal{G}(x_t), i = B^{-1}(q_t)\}$ contains (G,i) pairs "active" at time $t \in [T]$, and

$$C_t^{G,i} = \frac{1}{f(n_t^{G,i})} \left(\exp\left(\eta \frac{V_t^{G,i}}{f(n_t^{G,i})} \right) - \exp\left(-\eta \frac{V_t^{G,i}}{f(n_t^{G,i})} \right) \right).$$

Now, we show that Algorithm 1 guarantees $\sum_{(G,i) \in A_t(\pi_t)} v_\delta(q_t, s_t) C_{t-1}^{G,i}$ is small in expectation over the randomness of the algorithm.

Lemma 3.2. *Fix any $t \in [T]$, $\eta \in (0, \frac{1}{2})$, transcript $\pi_{1:t-1}$ recording a realization for the first $t-1$ rounds and x_t. At round t, Algorithm 1 chooses a distribution over q_t such that against any (ρ, rm)-smooth distribution over conformal scores s_t, we have:*

$$\mathop{\mathbb{E}}_{(s_t, q_t)} \left[\sum_{(G,i) \in A_t(\pi_t)} v_\delta(q_t, s_t) C_{t-1}^{G,i} \middle| \pi_{1:t-1} \right] \leq \rho L_{t-1}.$$

Carefully telescoping the bounded increase in surrogate loss over each round via Lemma 3.1 and 3.2 and noticing that $1/f(n)^2$ forms a convergent series yields the result of Theorem 3.1. □

4 Experiments

In this section, we summarize our experiments[3] — full details can be found in Appendix B. We evaluate MVP and compare it to more traditional conformal prediction methods on a variety of tasks. In each case, for a fair comparison, both MVP and the compared-to method receive as input the same predictive model and conformal scores.

Exchangeable data First, in Section B.1 we study a synthetic linear regression problem in a simple exchangeable (iid.) setting, and compare to split conformal prediction [Lei et al., 2018]. Given a regression model $f_t(x)$, we use conformal score $s_t(x, y) = |f_t(x) - y|$. The regression function is unknown at the outset, and must be learned during the interaction. As we show (Figure 1), MVP improves over split conformal prediction even in terms of marginal empirical coverage. This is because to keep the conformal scores exchangeable, split conformal prediction must split the data into two sets: one for training the regression model and one for calibrating prediction sets.[4] In contrast, our method does not require exchangeability, so we can both train the regression model and calibrate our prediction sets on the entire dataset. We are thus able to make better use of the data, and our regression function (and hence our conformal score function) becomes more informative faster. Then, we modify our regression problem so that there are 20 overlapping sub-populations, one of which (consisting of half of the data) has higher label noise. We measure groupwise coverage for MVP, for naive split conformal prediction with no knowledge of to-be-covered groups, and the method of Foygel Barber et al. [2020] which guarantees (conservative) groupwise coverage for intersecting groups. We find (Figure 2) that MVP significantly improves on both methods, obtaining correct coverage on all groups and (correctly) obtaining much smaller interval width on the low noise group.

Covariate shift Next, in Section B.2 we study a regression problem in the presence of covariate shift. First we replicate an experiment of Tibshirani et al. [2019], in which a synthetic covariate shift (with known propensity scores and known changepoint) is simulated on a UCI dataset. The method of Tibshirani et al. [2019] reweighs the calibration set using the propensity scores. MVP can also take advantage of propensity scores warm-start MVP on the same portion of the dataset that split conformal prediction uses for calibration, sampled with replacement after being re-weighted by the propensity scores. Both algorithms are then evaluated on the shifted distribution. We find both algorithms perform comparably. We then experiment with unanticipated covariate shift simulated on

[3] Our experiments are fairly lightweight, and can be run e.g. on a standard 12gb RAM Google Colab account.

[4] This is not only a theoretical requirement — split conformal prediction fails badly otherwise.

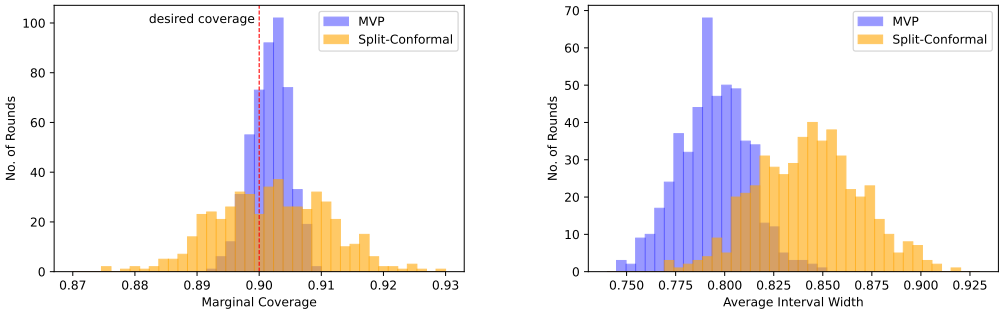


Figure 1: Left: histogram of empirical marginal coverage of MVP vs split conformal over 500 repeated trials. Right: histogram of average interval widths. MVP gets both empirical coverage that is more tightly concentrated around target (0.9), and narrower coverage interval widths.

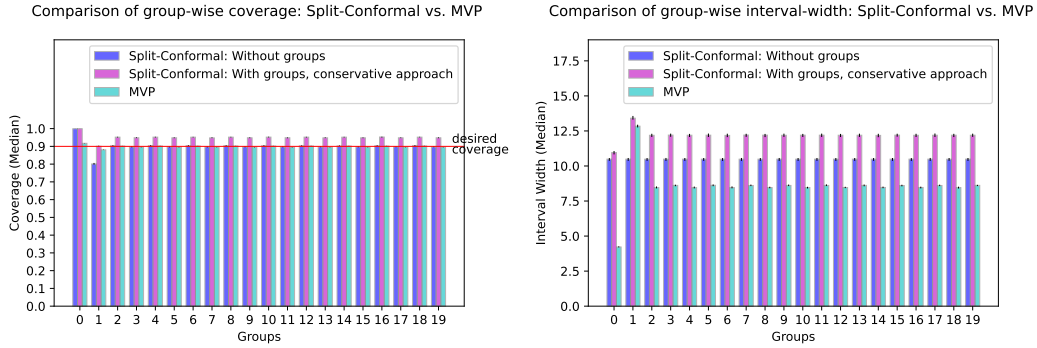


Figure 2: Left: median (over 100 indep. trials) per-group coverage of MVP vs split conformal. Right: median group-conditional interval widths. Compared to split conformal, MVP obtains target coverage on each group, and narrower interval widths.

datasets derived from 2018 U.S. Census data provided from the Folktables package [Ding et al., 2021]. We use the quantile-regression based conformal score proposed by Romano et al. [2019] for both methods. We compare to split conformal prediction calibrated on California data and evaluated on the Pennsylvania data. Similarly, we warm-start MVP on California data and measure its performance on the Pennsylvania data, finding that MVP gets the correct coverage rate and smaller interval widths compared to the split conformal method despite having no knowledge of the distribution shift.

Time series In Section B.3 we evaluate MVP on time series data — 20 years of stock returns, in a volatility prediction task. We compare MVP to the Adaptive Conformal Inference (ACI) method of Gibbs and Candes [2021], which guarantees marginal (but not threshold- or group- calibrated) coverage for adaptively chosen data. When evaluated in terms of marginal coverage, we find that MVP and ACI perform comparably, with ACI's coverage slightly closer to the target, but with MVP's sequence of predicted thresholds exhibiting more stability (Figure 9). We then complicate the experiment to exhibit the two advantages of MVP (groupwise and threshold-calibrated coverage). First, we define 20 intersecting groups of trading days: a period-1 sequence (all the days), a period-2 sequence (the even days), ..., a period-20 sequence (every 20th day). We add perturbations to the stock returns, such that each day receives an amount of noise commensurate with how many groups it belongs to, and find that MVP obtains the correct group-wise coverage, whereas ACI fails to: as shown in Figure 4a, ACI undercovers on most groups.[5] As our next experiment, we produce a fully adversarial sequence by presenting examples to ACI and MVP not in time order but in *sorted order by their conformal scores* (see Figure 4b). By construction, this sequence would cause split conformal prediction methods to have 0 coverage, while both ACI and MVP guarantee correct marginal coverage on it. However, we find (see Figure 11c) that ACI achieves its correct marginal coverage by violating

[5]To balance out this undercoverage, ACI strongly overcovers on points belonging to none of groups 2..20.

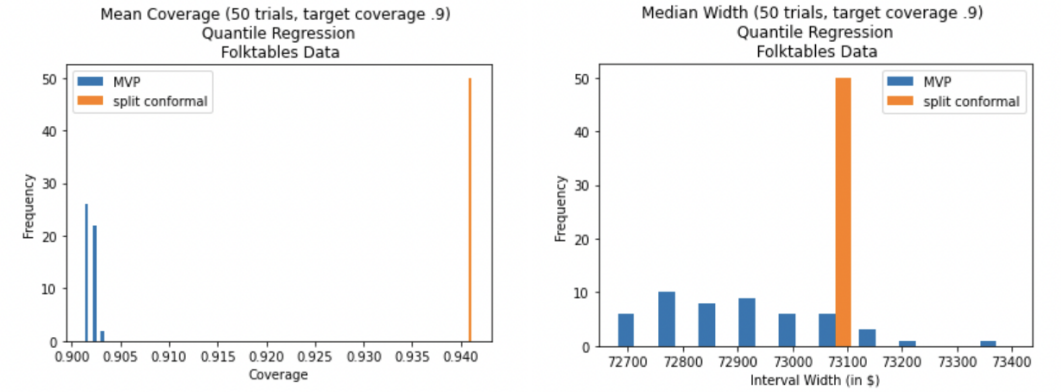
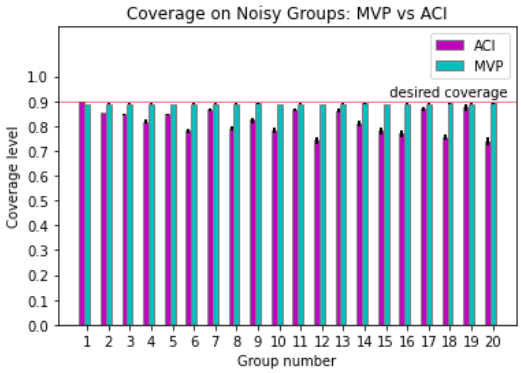
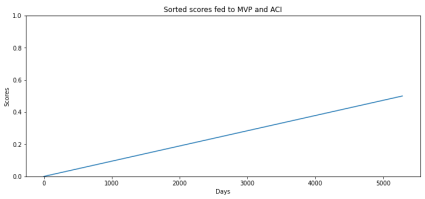


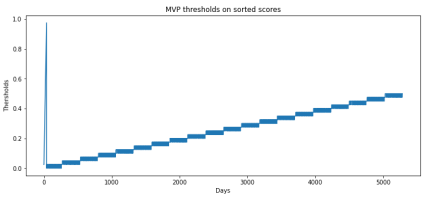
Figure 3: Both MVP and split conformal prediction are initialized on California data and evaluated on Pennsylvania data. On the left-hand is a histogram of the coverage for MVP and split conformal prediction over 50 trials; the right-hand figure is a histogram of the prediction interval width.



(a) MVP and ACI median coverage (over 20 indep. trials) on groups 1-20 on noisy data (group j consists of days t such that $t \equiv 0 \mod j$). MVP closely matches desired coverage level on all groups, whereas ACI significantly undercovers (within up to 10-20% from the target).



(b) Synthetic sorted scores sequence



(c) MVP thresholds on the sorted scores

Figure 4: MVP vs ACI comparison on time-series data

threshold calibration: namely, by predicting predominantly the trivial coverage interval (all of $[0, 1]$) and occasionally — short, under-covering intervals.[6] This strategy both yields thresholds that are uninformative about the input sequence of scores, and produces close-to-maximum average interval widths regardless of the actual magnitude of the input scores. In contrast, MVP, by virtue of its threshold calibration, outputs coverage thresholds that correctly track the input sequence of conformal scores (Figure 4c), and hence produces prediction intervals with the correct widths.

Classification: ImageNet Finally, in Section B.4 we compare MVP to the work of Angelopoulos et al. [2020] on a large-scale ImageNet classification task. We find that MVP obtains comparable coverage rates and prediction set sizes, despite the fact that the setting is favorable to Angelopoulos et al. [2020] — i.e. the data is i.i.d. and we measure only marginal coverage.

Potential societal impact of our work: When the underlying dataset consists of e.g. labeled individuals, our conformal prediction approach achieves target coverage guarantees not only marginally, but on arbitrary collections of user-specified population groups. Thus, we believe that when used responsibly, our method can serve as a key tool for advancing the interests of protected subpopulations, resulting in significant added positive societal impact.

[6] Cf. the "cheating" marginal coverage strategy discussed in Footnote 1.

References

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

Anastasios Nikolas Angelopoulos, Stephen Bates, Michael Jordan, and Jitendra Malik. Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations*, 2020.

Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327, 1986.

Victor Chernozhukov, Kaspar Wüthrich, and Zhu Yinchu. Exact and robust conformal inference methods for predictive machine learning with dependent data. In *Conference On Learning Theory*, pages 732–749, 2018.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in Neural Information Processing Systems*, 34, 2021.

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

Dean P Foster and Sergiu Hart. Forecast hedging and calibration. *Journal of Political Economy*, 129 (12):3447–3490, 2021.

Rina Foygel Barber, Emmanuel J Candès, Aaditya Ramdas, and Ryan J Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 2020.

Isaac Gibbs and Emmanuel Candes. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34, 2021.

Varun Gupta, Christopher Jung, Georgy Noarov, Mallesh M. Pai, and Aaron Roth. Online Multivalid Learning: Means, Moments, and Prediction Intervals. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, pages 82:1–82:24, 2022.

Úrsula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pages 1939–1948, 2018.

Peter Hoff. Bayes-optimal prediction with frequentist coverage control. *arXiv preprint arXiv:2105.14045*, 2021.

Christopher Jung, Changhwa Lee, Mallesh M Pai, Aaron Roth, and Rakesh Vohra. Moment multicalibration for uncertainty estimation. In *Conference on Learning Theory*. PMLR, 2021.

Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning*, pages 2564–2572, 2018.

Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. An empirical study of rich subgroup fairness for machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 100–109, 2019.

Michael Kim, Omer Reingold, and Guy Rothblum. Fairness through computationally-bounded awareness. In *Advances in Neural Information Processing Systems*, pages 4842–4852, 2018.

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

Sangdon Park, Osbert Bastani, Nikolai Matni, and Insup Lee. Pac confidence sets for deep neural networks via calibrated prediction. In *International Conference on Learning Representations*, 2019.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. *Advances in neural information processing systems*, 32, 2019.

Yaniv Romano, Rina Foygel Barber, Chiara Sabatti, and Emmanuel Candès. With malice toward none: Assessing uncertainty via equalized coverage. *Harvard Data Science Review*, 2020a.

Yaniv Romano, Matteo Sesia, and Emmanuel Candes. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems*, 33:3581–3591, 2020b.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *Journal of Machine Learning Research*, 9(Mar):371–421, 2008.

Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in Neural Information Processing Systems*, 32:2530–2540, 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Please see the dedicated paragraph on societal impact at the end of the main part of the paper — we believe that responsible use of our method will lead to significant added *positive* societal impact.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Described in detail where appropriate in Appendix B.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] The supplement will contain our code for the experiments.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] We used and cited open access data.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] None of our data contains such content or information.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Proofs from Subsection 3.1

Proof of Lemma 3.1. Fix $\pi_t = (x_t, s_t, q_t)$. For simplicity, we write $L_t = L_t(\pi_{1:t-1} \oplus \pi_t)$ and $L_t^{G,i} = L_t^{G,i}(\pi_{1:t-1} \oplus \pi_t)$ in the remainder of this proof. For any $(G, i) \notin A_t(\pi_t)$, we have that

$$\begin{aligned} V_t^{G,i} &= V_{t-1}^{G,i} \\ n_t^{G,i} &= n_{t-1}^{G,i}. \end{aligned}$$

and hence $L_t^{G,i} = L_{t-1}^{G,i}$.

On the other hand, for any $(G, i) \in A_t(\pi_t)$, we have

$$\begin{aligned} V_t^{G,i} &= V_{t-1}^{G,i} + v_\delta(q_t, s_t) \\ n_t^{G,i} &= n_{t-1}^{G,i} + 1. \end{aligned}$$

Then, we can bound the change in loss for that group-bucket pair $(G, i) \in A_t(\pi_t)$ as follows:

$$\begin{aligned}
&L_t^{G,i} - L_{t-1}^{G,i} \\
&= \left(\exp\left(\eta \frac{V_t^{G,i}}{f(n_t^{G,i})}\right) + \exp\left(-\eta \frac{V_t^{G,i}}{f(n_t^{G,i})}\right)\right) - \left(\exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\right) \\
&= \left(\exp\left(\eta \frac{V_{t-1}^{G,i} + v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i} + 1)}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i} + v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i} + 1)}\right)\right) - \\
&\quad \left(\exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\right) \\
&\overset{(1)}{\leq} \left(\exp\left(\eta \frac{V_{t-1}^{G,i} + v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i} + v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})}\right)\right) - \\
&\quad \left(\exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\right) \\
&= \exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\left(\exp\left(\eta \cdot \frac{v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})}\right) - 1\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\left(\exp\left(-\eta \cdot \frac{v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})}\right) - 1\right) \\
&\overset{(2)}{\leq} \exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\left(\eta \cdot \frac{v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})} + \frac{2\eta^2}{f(n_{t-1}^{G,i})^2}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\left(-\eta \cdot \frac{v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})} + \frac{2\eta^2}{f(n_{t-1}^{G,i})^2}\right) \\
&= \eta \cdot \frac{v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})}\left(\exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right) - \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\right) + \\
&\quad \frac{2\eta^2}{f(n_{t-1}^{G,i})^2}\left(\exp\left(\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right) + \exp\left(-\eta \frac{V_{t-1}^{G,i}}{f(n_{t-1}^{G,i})}\right)\right) \\
&= \eta v_\delta(q_t, s_t) C_{t-1}^{G,i} + \frac{2\eta^2}{f(n_{t-1}^{G,i})^2} L_{t-1}^{G,i}
\end{aligned}$$

The first inequality (1) holds due to $e^x + e^{-x}$ being monotone with respect to $|x|$, inequality (2) follows from the fact that for $0 \leq |x| \leq \frac{1}{2}$, $\exp(x) \leq 1 + x + 2x^2$, and $\left|\eta \cdot \frac{v_\delta(q_t, s_t)}{f(n_{t-1}^{G,i})}\right| \leq \frac{1}{2}$ because of the way we set $\eta \in (0, 1/2)$ and the fact that $|v_\delta(q_t, s_t)| \in [0, 1]$ and $f(n_{t-1}^{G,i}) \geq 1$.

Therefore, we have the desired bound

$$L_t - L_{t-1} = \sum_{(G,i) \in A_t(\pi_t)} L_t^{G,i} - L_{t-1}^{G,i} \leq \sum_{(G,i) \in A_t(\pi_t)} \eta v_\delta(q_{t+1}, s_{t+1}) C_{t-1}^{G,i} + \frac{2\eta^2}{f(n_t^{G,i})^2} L_t^{G,i}. \quad \square$$

Proof of Lemma 3.2. For simplicity, suppose we write

$$u(q,s) = v_\delta(q,s) \sum_{(G,i)\in A_t(\pi_t)} C_t^{G,i} = v_\delta(q,s) C_t^q$$

where we overload the notation to write

$$C_t^q = C_t^{B^{-1}(q)} = \sum_{G\in\mathcal{G}(x_t)} C_t^{G,B^{-1}(q)}.$$

Case (i) $C_{t-1}^i < 0$ for all $i \in [n]$: With $q_t = 1$, we have

$$\mathop{\mathbb{E}}_{q_t\sim Q^L, s_t\sim Q^A}[u(q_t,s_t)|x_t] = C_{t-1}^1(x_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}[v_\delta(1,s_t)|x_t] < 0$$

as $v_\delta(1,d_t) = 1-(1-\delta) > 0$.

Case (ii) $C_{t-1}^i > 0$ for all $i \in [n]$: With $q_t = 0$, we have

$$\mathop{\mathbb{E}}_{q_t\sim Q^L, s_t\sim Q^A}[u(q_t,s_t)|x_t] = C_{t-1}^0(x_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}[v_\delta(0,s_t)|x_t] < \rho C_{t-1}^0(x_t) < \rho L_{t-1}.$$

as we have

$$\mathop{\mathbb{E}}_{s_t\sim Q^A}[\mathrm{Cover}(0,s_t)|x_t] - (1-\delta) \le \mathop{\mathbb{E}}_{s_t\sim Q^A}[\mathrm{Cover}(0,s_t)|x_t] = \mathop{\Pr}_{s_t\sim Q^A}[s_t = 0|x_t] \le \rho$$

Case (iii) there exists $i^* \in [n-1]$ such that $C_{t-1}^{i^*} \cdot C_{t-1}^{i^*+1} \le 0$: First, consider the case where $C_{t-1}^{i^*} \ge 0$ and $C_{t-1}^{i^*+1} \le 0$. Then, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q_t\sim Q^L, s_t\sim Q^A}[u(q_t,s_t)|x_t] \\
&= p_t \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[u\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] + (1-p_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[u\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right] \\
&= p_t C_{t-1}^{i^*}(x_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] + (1-p_t) C_{t-1}^{i^*+1} \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right] \\
&\le p_t C_{t-1}^{i^*}\left(\mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right]\right) + (1-p_t) C_{t-1}^{i^*+1} \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right] \\
&= \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right]\left(p_t C_{t-1}^{i^*} + (1-p_t) C_{t-1}^{i^*+1}\right) \\
&= 0.
\end{aligned}$$

The first inequality follows from the fact that $\mathrm{Cover}(\frac{i^*}{n} - \frac{1}{rn}, s) \le \mathrm{Cover}(\frac{i^*}{n}, s)$ for any s. Now, consider the other case where $C_{t-1}^{i^*} \le 0$ and $C_{t-1}^{i^*+1} \ge 0$.

$$\begin{aligned}
&\mathop{\mathbb{E}}_{q_t\sim Q^L, s_t\sim Q^A}[u(q_t,s_t)|x_t] \\
&= p_t \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[u\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] + (1-p_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[u\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right] \\
&= p_t C_{t-1}^{i^*}(x_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] + (1-p_t) C_{t-1}^{i^*+1} \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right] \\
&\le p_t C_{t-1}^{i^*}(x_t) \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] + (1-p_t) C_{t-1}^{i^*+1}\left(\mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] + \rho\right) \\
&= \rho(1-p_t) C_{t-1}^{i^*+1} + \mathop{\mathbb{E}}_{s_t\sim Q^A}\left[v_\delta\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right]\left(p_t C_{t-1}^{i^*} + (1-p_t) C_{t-1}^{i^*+1}\right) \\
&= \rho L_{t-1}.
\end{aligned}$$

The first inequality follows from the fact that

$$\begin{aligned}
&\Pr_{s_t\sim Q^A}\left[\text{Cover}\left(\frac{i^*}{n}, s_t\right)\middle| x_t\right] - \Pr_{s_t\sim Q^A}\left[\text{Cover}\left(\frac{i^*}{n} - \frac{1}{rn}, s_t\right)\middle| x_t\right] \\
&\leq \Pr_{s_t\sim Q^A}\left[s_t \in \left[\frac{i^*}{n} - \frac{1}{rn}, \frac{i^*}{n}\right]\middle| x_t\right] \\
&\leq \rho.
\end{aligned}$$

□

Proof of Theorem 3.1. Fix any round $t \in [T]$ and transcript $\pi_{1:t-1}$. For simplicity, we write $L_t = L_t(\pi_{1:t})$. Then, we can use Lemma 3.1 to prove the following lemma.

Lemma A.1. *Fix any transcript $\pi_{1:T}$. Then, for any round $t \in [T]$, we have*

$$L_t \leq L_{t-1}\left(1 + \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{(G,i)\in A_t(\pi_t)} \frac{2\eta^2}{f(n_t^{G,i})^2}\right).$$

Proof. Fix transcript $\pi_{1:T}$. Then at any round t, we have

$$\begin{aligned}
&L_t \\
&= L_{t-1} + L_t - L_{t-1} \\
&\leq L_{t-1} + \sum_{(G,i)\in A_t(\pi_t)} \eta v_\delta(q_t, (x_t, s_t)) C_{t-1}^{G,i} + \frac{2\eta^2}{f(n_t^{G,i})^2} L_{t-1}^{G,i} && \text{(Lemma 3.1)} \\
&\leq L_{t-1} + \sum_{(G,i)\in A_t(\pi_t)} \eta v_\delta(q_t, (x_t, s_t)) C_{t-1}^{G,i} + L_{t-1} \sum_{(G,i)\in A_t(\pi_t)} \frac{2\eta^2}{f(n_t^{G,i})^2} && \left(L_{t-1}^{G,i} \leq L_{t-1}\right) \\
&\leq L_{t-1}\left(1 + \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{(G,i)\in A_t(\pi_t)} \frac{2\eta^2}{f(n_t^{G,i})^2}\right).
\end{aligned}$$

□

Applying Lemma A.1 recursively, we get

$$\begin{aligned}
L_T &\leq L_0 \prod_{t=1}^{T}\left(1 + \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{(G,i)\in A_t(\pi_t)} \frac{2\eta^2}{f(n_t^{G,i})^2}\right) \\
&\overset{(3)}{\leq} L_0 \prod_{t=1}^{T} \exp\left(\frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{(G,i)\in A_t(\pi_t)} \frac{2\eta^2}{f(n_t^{G,i})^2}\right) \\
&\leq L_0 \exp\left(\sum_{t=1}^{T} \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{t=1}^{T} \sum_{(G,i)\in A_t(\pi_t)} \frac{2\eta^2}{f(n_t^{G,i})^2}\right) \\
&\leq L_0 \exp\left(\sum_{t=1}^{T} \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{G\in\mathcal{G}, i\in[m]} \sum_{n=1}^{n_T^{G,i}} \frac{2\eta^2}{f(n)^2}\right) \\
&\leq L_0 \exp\left(\sum_{t=1}^{T} \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \sum_{G\in\mathcal{G}, i\in[m]} \sum_{n=1}^{\infty} \frac{2\eta^2}{f(n)^2}\right) \\
&\leq L_0 \exp\left(\sum_{t=1}^{T} \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + 2\eta^2 K_\epsilon |\mathcal{G}| m\right)
\end{aligned}$$

$$= 2|\mathcal{G}|m \exp\left(\sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + 2\eta^2 K_\epsilon |\mathcal{G}|m\right)$$

where inequality (3) follows from $1 + x \leq \exp(x)$.

Taking the log of both sides, we have

$$\ln(L_T) \leq \ln(2|\mathcal{G}|m) + \sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + 2\eta^2 K_\epsilon |\mathcal{G}|m$$

for any $\pi_{1:T}$.

By Observation 3.1, it suffices to upper bound $\max_{G\in\mathcal{G},i\in[m]} \frac{|V_T^{G,i}|}{f(n_T^{G,i})}$. We have:

$$\begin{aligned}
\max_{G\in\mathcal{G},i\in[m]} \frac{|V_T^{G,i}|}{f(n_T^{G,i})} &= \frac{1}{\eta} \ln\left(\exp\left(\max_{G\in\mathcal{G},i\in[m]} \frac{\eta|V_T^{G,i}|}{f(n_T^{G,i})}\right)\right) \\
&= \frac{1}{\eta} \ln\left(\max_{G\in\mathcal{G},i\in[m]} \exp\left(\frac{\eta|V_T^{G,i}|}{f(n_T^{G,i})}\right)\right) \\
&\leq \frac{1}{\eta} \ln\left(\sum_{G\in\mathcal{G},i\in[m]} \exp\left(\frac{\eta|V_T^{G,i}|}{f(n_T^{G,i})}\right)\right) \\
&\leq \frac{1}{\eta} \ln\left(\sum_{G\in\mathcal{G},i\in[m]} \exp\left(\frac{\eta V_T^{G,i}}{f(n_T^{G,i})}\right) + \exp\left(\frac{-\eta V_T^{G,i}}{f(n_T^{G,i})}\right)\right) \\
&= \frac{\ln(L_T)}{\eta} \\
&\leq \frac{1}{\eta}\left(\ln(2|\mathcal{G}|m) + \sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + 2\eta^2 K_\epsilon |\mathcal{G}|m\right).
\end{aligned}$$

Taking expectation over $\pi_{1:T}$ on both sides, we get

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\pi_{1:T}}\left[\max_{G\in\mathcal{G},i\in[m]} \frac{|V_T^{G,i}|}{f(n_T^{G,i})}\right] \\
&\leq \mathop{\mathbb{E}}_{\pi_{1:T}}\left[\frac{1}{\eta}\left(\ln(2|\mathcal{G}|m) + \sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + 2\eta^2 K_\epsilon |\mathcal{G}|m\right)\right] \\
&\leq \frac{1}{\eta}\left(\ln(2|\mathcal{G}|m) + 2\eta^2 K_\epsilon |\mathcal{G}|m + \mathop{\mathbb{E}}_{\pi_{1:T}}\left[\sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i}\right]\right).
\end{aligned}$$

Let us focus only on the third term:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\pi_{1:T}}\left[\sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i}\right] \\
&= \mathop{\mathbb{E}}_{\pi_{1:T-1}}\left[\mathop{\mathbb{E}}_{\pi_T}\left[\sum_{t=1}^{T} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} \middle| \pi_{1:T-1}\right]\right] \\
&= \mathop{\mathbb{E}}_{\pi_{1:T-1}}\left[\sum_{t=1}^{T-1} \frac{\eta v_\delta(q_t,(x_t,s_t))}{L_{t-1}} \sum_{(G,i)\in A_t(\pi_t)} C_{t-1}^{G,i} + \frac{\eta}{L_{T-1}} \mathop{\mathbb{E}}_{\pi_T}\left[v_\delta(q_T,(x_T,s_T)) \sum_{(G,i)\in A_T(\pi_T)} C_{T-1}^{G,i} \middle| \pi_{1:T-1}\right]\right]
\end{aligned}$$

$$
\begin{aligned}
&\overset{(4)}{\leq} \mathop{\mathbb{E}}_{\pi_{1:T-1}} \left[\sum_{t=1}^{T-1} \frac{\eta v_\delta(q_t, (x_t, s_t))}{L_{t-1}} \sum_{(G,i) \in A_t(\pi_t)} C_{t-1}^{G,i} + \eta\rho \right] \\
&\leq \dots \\
&\leq \eta\rho T
\end{aligned}
$$

where inequality (4) comes from Lemma 3.2.

In other words, we have

$$
\begin{aligned}
\mathop{\mathbb{E}}_{\pi_{1:T}} \left[\max_{G \in \mathcal{G}, i \in [m]} \frac{|V_T^{G,i}|}{f(n_T^{G,i})} \right] &\leq \frac{1}{\eta} \left(\ln(2|\mathcal{G}|m) + 2\eta^2 K_\epsilon |\mathcal{G}|m + \eta\rho T \right) \\
&= \frac{\ln(2|\mathcal{G}|m)}{\eta} + 2\eta|\mathcal{G}|mK_\epsilon + \rho T \\
&\leq \sqrt{4K_\epsilon|\mathcal{G}|m\ln(|\mathcal{G}|m)} + \rho T
\end{aligned}
$$

where the last inequality follows from setting $\eta = \sqrt{\frac{\ln(|\mathcal{G}|m)}{2K_\epsilon|\mathcal{G}|m}}$. Note that $\eta < 1/2$ as $2\ln(|\mathcal{G}|m) < K_\epsilon|\mathcal{G}|m$ because $K_\epsilon \geq 1$.

□

B Experiments

In this section, we evaluate MVP and compare it to more traditional methods of conformal prediction on a variety of tasks. In each comparison, we use the same model and conformal score for MVP and for the methods we compare against — the only difference is the type of the conformal prediction wrapper.

First in Section B.1 we study a synthetic regression problem in a simple exchangeable (i.i.d.) setting, and compare to split conformal prediction [Lei et al., 2018]. We show that even when we measure only marginal empirical coverage, MVP improves over split conformal prediction when the regression function must be learned. This is because to maintain the exchangeability of conformal scores, split conformal prediction must split the data into two sets — one for training the regression function and one for calibrating the prediction sets.[7] In contrast, our method does not require exchangeability, so we can both train the regression model and calibrate our prediction sets on the entire dataset. Then, we modify our regression problem so that there are 20 overlapping sub-populations, and one of the sub-populations (consisting of half of the data points) has higher label noise. We measure group-wise coverage for MVP, for naive split conformal prediction that has no knowledge of the groups to be covered, and the method of Foygel Barber et al. [2020] which guarantees (conservative) group-wise coverage for intersecting groups. We find that MVP significantly improves on both methods.

Next, in Section B.2 we study a regression problem in the presence of covariate shift. First we replicate an experiment of Tibshirani et al. [2019], in which a synthetic covariate shift (with known propensity scores and known changepoint) is simulated on a UCI dataset. The method of Tibshirani et al. [2019] reweights the calibration set using the propensity scores. MVP can also take advantage of propensity scores when they are known: we give MVP a "warm start" from the same portion of the dataset that split conformal prediction uses for calibration, sampled with replacement after being re-weighted by the propensity scores. Both algorithms are then evaluated on the shifted distribution. We find both algorithms perform comparably. We then experiment with unknown and unanticipated covariate shift simulated on datasets derived by U.S. Census data provided from the Folktables package Ding et al. [2021]: We compare to split conformal prediction calibrated on the California data (this time without re-weighting) and evaluated on the Pennsylvania data. Similarly, we again give MVP a warm start on the California data (again without reweighting), and then measure its performance on 2018 Pennsylvania Census data. We find that MVP obtains the correct coverage rate and smaller interval widths compared to the split conformal method despite having no knowledge of the distribution shift.

[7] This is not only a theoretical requirement — split conformal prediction fails badly otherwise.

In Section B.3 we evaluate MVP on time series data — 20 years of stock returns, in a volatility prediction task. We compare MVP to the Adaptive Conformal Inference (ACI) method of Gibbs and Candes [2021], which guarantees marginal (but not threshold calibrated) coverage for adaptively chosen data. When evaluated in terms of marginal coverage, we find that MVP and ACI perform comparably: ACI obtains average coverage slightly closer to the target, whereas MVP predicts a less oscillating sequence of thresholds. We then complicate the experiment to exhibit the two advantages of MVP (groupwise coverage and threshold calibrated coverage). First we define 20 intersecting groups defined as the trading days that are multiples of $1, 2, \ldots, 20$. We add perturbations to the stock returns that differ across these groups, and find that MVP continues to produce the correct group-wise coverage, whereas ACI fails to. Next, we produce a fully adversarial sequence by presenting examples to the algorithms not in time order but in *sorted order by their conformal scores*. By construction, this sequence would cause split conformal prediction methods to have 0 coverage, but both ACI and MVP are required to obtain the correct marginal coverage on this sequence. However, we find that given this sequence, ACI reduces to a strategy that, similar to the uninformative "cheating" strategy mentioned in Footnote 1, predicts the trivial coverage interval (all of $[0, 1]$) on most days — which guarantees marginal but not threshold calibrated coverage, and does not produce non-trivial average interval widths. In contrast, MVP, by virtue of its threshold calibration condition, produces a sequence of coverage thresholds that correctly track the sequence of conformal scores of the true labels in the data, and hence produces prediction intervals with the correct widths.

Finally in Section B.4 we compare MVP to the work of Angelopoulos et al. [2020] on a large-scale image-net classification task. We find that MVP obtains comparable coverage rates and prediction set sizes, despite the fact that the setting is favorable to Angelopoulos et al. [2020] — i.e. the data is i.i.d. and we measure only marginal coverage.

B.1 Exchangeable Data

B.1.1 Basic Experimental Setup and Marginal Coverage

We simulate a synthetic linear regression problem in which the regression model must be trained in tandem with the conformal predictor. The feature domain consists of 10 binary features and 290 continuous features. For any input x, the binary features are drawn from a uniform distribution and each continuous feature is drawn from a normal distribution $\mathcal{N}(0, \sigma_x^2)$. Each example's label is governed by an ordinary least squares model:

$$y = \langle \theta, x \rangle + \mathcal{N}(0, \sigma_y^2)$$

for some fixed vector $\theta \in \mathbb{R}^{300}$ unknown to the learner.

We run both MVP and split conformal prediction [Lei et al., 2018] using the conformal score $s_t(x, y) = |f_t(x) - y|$. When running MVP, we train f_t using least squares regression on all points $(x_{t'}, y_{t'})$ for $t' < t$. For split conformal prediction, we divide points evenly between a calibration set and a training set (points from odd time steps go into the calibration set, points from even time steps go into the training set), and f_t is trained using least squares regression on all points in the training set at time $t - 1$. (We also tried training f_t on all points, but this causes split conformal prediction to fail catastrophically).

Results We set $\sigma_x^2 = 0.1$, $\sigma_y^2 = 0.2$ and run 500 independent trials of our experiment, each for $T = 2000$ steps. θ is independently selected for each trial. The results are shown in Figure 5. MVP simultaneously obtains empirical coverage that is more tightly concentrated around its target and obtains narrower coverage intervals compared to split conformal prediction. Despite the fact that we are in a setting that is extremely favorable to split conformal prediction (i.i.d. data and marginal coverage evaluation), MVP has the advantage that it can use a regression function f_t trained on *all* past data, without the need to set aside a calibration set. This is needed for split conformal prediction to maintain the exchangeability of the conformal scores.

B.1.2 Multi-Group Coverage

We now compare the coverage of MVP to that of split conformal prediction not just marginally, but group-wise. We use the same feature generation process and conformal score as for our marginal coverage experiment described in Section B.1.1. Recall that the first 10 of the 300 features in our

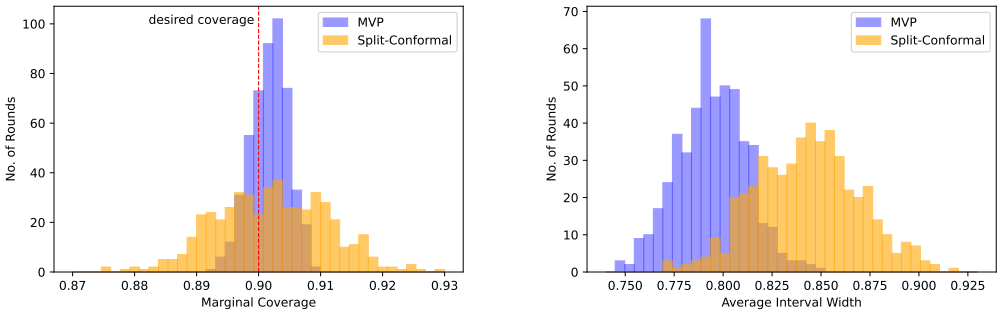


Figure 5: The plot on the left is a histogram of the empirical marginal coverage of MVP and split conformal prediction over 500 repeated trials; the right hand plot is similarly a histogram of the average interval width for both methods. We see that MVP gets both empirical coverage that is more tightly concentrated around its target (0.9) and narrower coverage interval width.

data domain are binary, which we now use to define 20 (intersecting) groups defined by the value of each of the 10 binary features. Labels are still generated according to an ordinary least squares model, but now the noise rate depends on the groups that each datapoint is a member of. Specifically:

$$y = \langle \theta, x \rangle + \mathcal{N}\left(0, \sigma^2 + \sum_{i=1}^{10} \sigma_i^2 x_i\right)$$

for some fixed vector $\theta \in \mathbb{R}^{300}$, and for fixed values of σ_i, each associated with one of the binary features indicating groups.

We run MVP parameterized to promise multi-valid coverage for the set of 20 intersecting groups defined by the first 10 binary features of the input: For each $i \in \{0, 1, \cdots, 19\}$, we define $G_i = \{x \in \mathcal{X} \mid x_{\lceil (i+1)/2 \rceil} \equiv_2 i\}$ and let $\mathcal{G} = \{G_i \mid 0 \leq i \leq 19\}$. At each time-step t, we train a regression model f_t on all past data.

We compare to two benchmark conformal prediction methods. First, we compare to naive split conformal prediction (which ignores the group structure), just as in Section B.1.1. This method offers no guarantees about group-wise coverage. Second, we compare to the method of Foygel Barber et al. [2020] which separately computes a calibration threshold for each of the 20 groups marginally, and on each example x_t, uses the most conservative (i.e. largest) threshold associated with any of the groups for which x_t is a member. This method guarantees coverage *at least* the target coverage level, but does not guarantee coverage approaching the target. Note there are 2^{10} different subsets of groups that each example might be a member of, and so the method of Romano et al. [2020a] which separately calibrates on *disjoint* groupings of the data cannot be run without having roughly 1000-fold more data. For both conformal prediction methods we equally split the data between a training set used for training the regression model f_t and a calibration set used for calibrating the prediction intervals. We run MVP with $m = 40$ calibration buckets.

Results We run 100 independent trials of our experiment, each for $T = 20,000$ data points. Our results are plotted in Figure 6. We set $\sigma_1^2 = 3.0$ and $\sigma_2^2 = \ldots = \sigma_{10}^2 = 0.1$ so that G_0 is a "low noise" group and G_1 is a "high noise" group. We keep the values of the σ_i fixed across all trials, but each is run with an independently drawn θ. As expected, we find that naive split conformal prediction fails to meet its coverage target, over-covering on the low noise group and under-covering on the high noise group, and uses a uniform interval width. In contrast, both MVP and the conservative method of Foygel Barber et al. [2020] use different average interval widths for different groups. The conservative method of Foygel Barber et al. [2020] always gets at least the target coverage, but significantly over-covers on every group except for the high noise group. In contrast MVP obtains the target coverage on every group. MVP also has lower average interval width on every group compared to Foygel Barber et al. [2020], and (correctly) produces significantly narrower intervals on the low noise group.

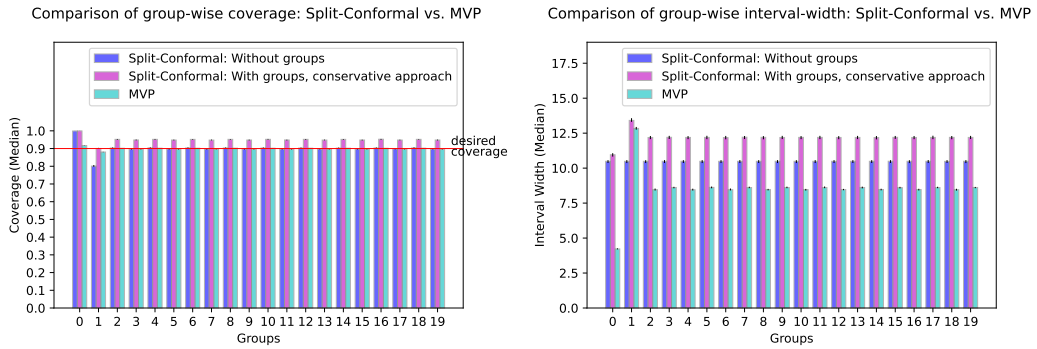


Figure 6: On the left we plot the median coverage (for each trial) conditional on membership in each of our 20 groups, as averaged over 100 independent trials. On the right we plot the median interval width conditional on membership in each of the 20 groups. Compared to the split conformal prediction methods, we see that MVP obtains the target coverage level on each group (neither under nor over covering), and obtains narrower interval widths. The error bars represent 25th and 75th quantiles, and they are not easily visible in this figure as they are quite narrow: for conformal with groups, both bar endpoints are within ± 0.0039 from the median, for conformal without groups, within ± 0.0054 from the median, and for MVP, within ± 0.0021 from the median.

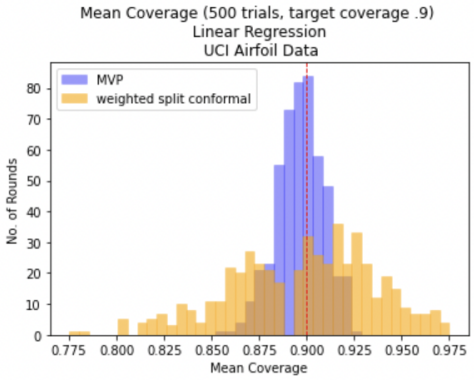
B.2 Covariate Shift

B.2.1 Known Covariate Shift with UCI Airfoil Data

We first study the setting of known covariate shift considered by Tibshirani et al. [2019] (which introduced the weighted split conformal prediction method that we use as our point of comparison) and replicate their design. Following Tibshirani et al. [2019], we use the airfoil dataset from the UCI Machine Learning Repository [Dua and Graff, 2017] which consists of data of NACA 0012 airfoils. The dataset contains $N = 1503$ total instances of $d = 5$ features (frequency in Hz, angle of attack in degrees, chord length in meters, free-stream velocity in meters per second, and suction side displacement thickness in meters) The target feature for prediction is scaled sound pressure in decibels. In this setting, the data available for calibration is drawn from a different distribution from the data that is used for evaluation, but the distributions differ only in their relative weighting of feature vectors, and the relative weightings (likelihood ratios) are known.

Weighted split conformal prediction uses the likelihood ratios between the training and evaluation distributions to find weighted quantiles of the conformal scores on the evaluation data distribution. We note that MVP can also make use of these likelihood ratios when they are known. We do so by "warm starting" MVP by running it on the data that weighted split conformal prediction uses for calibration, but re-sampled with replacement using rejection sampling according to the known likelihood ratios.

Following the protocol in Tibshirani et al. [2019], for both methods, we use 25% of the data to train the underlying linear regression model that will be given to both MVP and weighted split conformal prediction. (It is necessary to use a separate split of the data for the method of Tibshirani et al. [2019], but for our method we could have shared data between training and calibration, which would give us an advantage of the sort we demonstrated in Section B.1. We do not do this in this experiment to disentangle different aspects of the comparison between our techniques). The weighted split conformal prediction algorithm is then given a calibration dataset of 25% of the data to compute the residual quantiles and finally samples with replacement 50% of the remaining points for the evaluation dataset, with probabilities proportional to $w(x) = \exp(x^T \beta)$, where $\beta = (-1, 0, 0, 0, 1)$. This final fold simulates a synthetic covariate shift and in our comparison the weighted split conformal method that has oracle access to the shift likelihood ratios. When running MVP, we use this 25% of the dataset in a comparable way: we sample the calibration fold of the remaining data with replacement with probilities proportional to $w(x)$ and use it to run MVP as a warm start (i.e. the predictions that MVP makes on this fold are not recorded in the metrics we report). This uses the known conformal scores in a similar way to how they are used in weighted split conformal prediction. MVP is then evaluated on an evaluation dataset obtained the same way as for weighted split conformal,



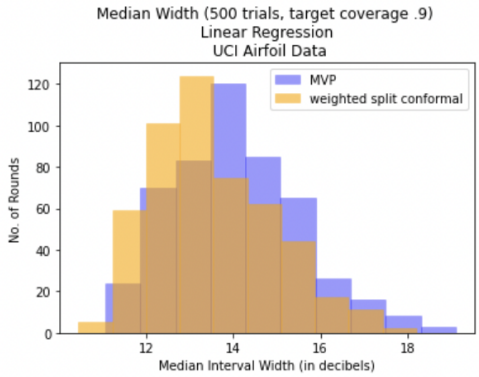


Figure 7: The left-hand figure shows a histogram of the coverage rate of MVP and weighted split conformal prediction over 500 trials; the right-hand figure is a histogram of the median prediction interval widths over the same 500 trials.

by sampling 50% of the remaining data with probabilities proportional to $w(x)$. We run MVP with $m = 40$ threshold-calibration buckets.

Figure 7 shows a histogram of the coverage rate and median prediction interval width of both methods over 500 trials of the experiment, where each trial indicates a different train-test split of the data and a different sampling of the shifted data for the algorithm. We see that MVP obtains coverage that is more tightly concentrated around its target (0.9) compared to weighted split conformal prediction and comparable interval widths. We note that this is even without letting MVP train its regression model on the calibration dataset.

B.2.2 Unknown Shift with Folktables Data

Next we evaluate split conformal prediction and MVP on real data exhibiting distribution shift, in which the distribution changepoint and propensity scores are unknown (and so cannot be used to weight the calibration set as in our earlier experiment). Here we use a dataset derived from the 2018 Census American Community Survey Public Use Microdata provided by the Folktables package [Ding et al., 2021]. The dataset consists of $N = 263973$ (CA: 195665, PA: 68308) instances each comprising of $d = 9$ features. We subsample the dataset ($.4$ of CA, $.2$ of PA), thus using $N = 91927$ overall. The features of the data are Census demographic attributes and the target prediction variable is income. We follow Ding et al. [2021] in investigating covariate shift that results from using data derived from different states related to the same task.

As in all of our experiments, MVP is trained using $m = 40$ buckets for calibration. For both MVP and split conformal prediction we use the quantile-regression based conformal score from Romano et al. [2019], using a quantile regression model trained on half of the available California data. We then use the remaining California data as the calibration dataset, used to “warm start" MVP and compute the residual quantiles for split conformal prediction. Finally, we evaluate MVP and split conformal on the Pennsylvania data and report a histogram of the empirical coverage and interval widths for both methods over 50 trials in Figure 8. MVP comes very close to its coverage target (0.9), whereas split conformal prediction significantly over-covers. Similarly, MVP obtains narrower average prediction interval widths. Here the empirical coverage for both methods is much more tightly concentrated than it is for the UCI Airfoil dataset: this is because the dataset we are using in this experiment is roughly 60 times larger.

B.3 Time Series Data

B.3.1 Basic Experimental Setup and Marginal Coverage

In this set of experiments, we run MVP on stock market data and compare our performance to the Adaptive Conformal Inference (ACI) algorithm of Gibbs and Candes [2021], a recent method that guarantees *marginal* coverage for adversarially chosen data. In contrast to MVP, ACI promises only

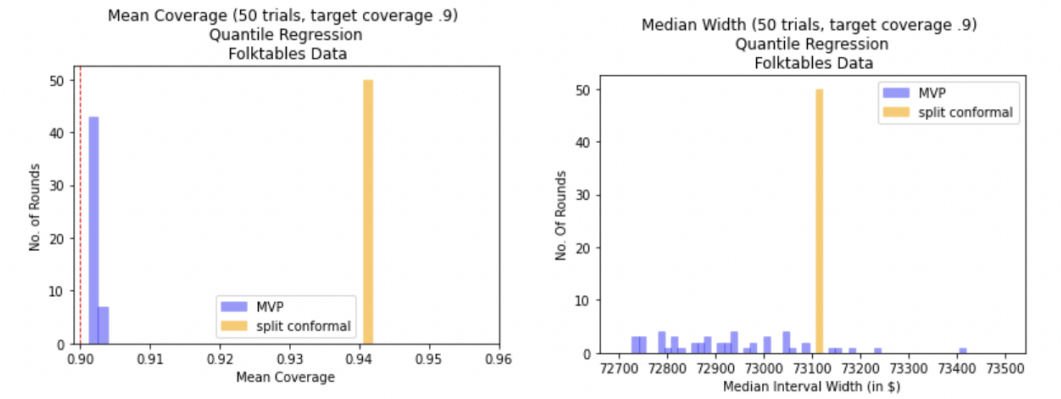


Figure 8: The left-hand figure shows a histogram of the coverage for MVP and split conformal prediction over 50 trials; the right-hand figure shows a histogram of the prediction interval width.

marginal coverage (in particular, its guarantees are not threshold calibrated), and so we expect its convergence to be faster but that its thresholds will fluctuate more; our experiments bear this out.

To directly compare to ACI, we use the same dataset and model construction as in Gibbs and Candes [2021]. Specifically, we start with WSJ daily open price data[8] for AMD stocks in 2000-2020 (corresponding to $T = 5283$ price points $p_1, \ldots, p_T$). We calculate daily returns r_t as $r_t = \frac{p_t - p_{t-1}}{p_{t-1}}$ for every day t. Based on the returns, we then calculate the (realized) daily volatility as $v_t = r_t^2$ for $t \in [T]$. For our prediction task we train a model to estimate daily volatility levels v_t. Following Gibbs and Candes [2021], we use a standard sequential prediction model called GARCH [Bollerslev, 1986]; every day, GARCH makes volatility prediction σ_t, and autoregressively updates the model once it sees the realized volatility v_t. The conformal score we use on day t is the *normalized regression score* $s_t(t, v) = |v_t - \sigma_t|/\sigma_t$. Here σ_t is the prediction that the GARCH model makes at round t, and possible realizations of the volatility v_t play the role of the label. We run MVP and ACI, for miscoverage target $\delta = 0.1$, on the (rescaled) scores[9] $\tilde{s}_1, \ldots, \tilde{s}_T$ of the GARCH model trained to predict AMD stock volatility. In all our experiments with ACI, we set the ACI hyperparameters as follows: $\gamma = 0.005$ (step size), lookback = 100, offset = 10. Figure 9 shows the sequences of conformity thresholds for MVP and ACI. In general we find that even when we only measure marginal prediction, MVP performs comparably to ACI. Both methods obtain coverage close to the target rate of 0.9, where ACI consistently gets a bit closer to the target rate. We visually observe that MVP makes more stable predictions compared to ACI, locally converging to a small stable set of threshold values (and moving over to the next stable set of thresholds once the scores have drifted sufficiently far), whereas ACI uses continuously fluctuating threshold values (this is expected, since it is not aiming for threshold calibrated coverage).

B.3.2 Multigroup Coverage

Next, we augment the experimental setup to investigate multigroup coverage. We define a set of groups based on whether the index of the trading day is divisible by $1, \ldots, 20$: Define $x_t = t$, and let $\mathcal{G} = \{G_1, \ldots, G_{20}\}$, where G_i is defined as the set of all t such that $t \equiv 0 \mod i$. In other words, G_1 consists of the set of all time steps, G_2 consists of even time steps, G_3 consists of time-steps that are multiples of 3, and so on. As these sub-groups mutually intersect, it is not possible to run a separate copy of ACI on each one.

To provide sub-group variability, we artificially introduce varying levels of group-specific additive noise into the stock return data: for each $i \in [20]$, we add noise sampled from $\mathcal{N}(0, \hat{\sigma}_{\text{ret}})$ to the stock

[8] Available at `https://www.wsj.com/market-data`

[9] MVP assumes that the input scores $s_t \in [0, 1]$, but in this set of experiments we can only guarantee that the normalized regression score $s_t = |v_t - \sigma_t|/\sigma_t \in [0, \infty)$. Due to this, we feed MVP (and ACI, for consistency) modified scores $\tilde{s}_t = \frac{s_t}{1+s_t} \in [0, 1]$. This type of rescaling works more generally in any setting where MVP's input scores belong to $[0, \infty)$ and need to be rescaled to be in $[0, 1]$: indeed, observe that the mapping $x \rightarrowtail \frac{x}{1+x}$ is a monotonic continuous bijection from $[0, \infty)$ to $[0, 1)$.

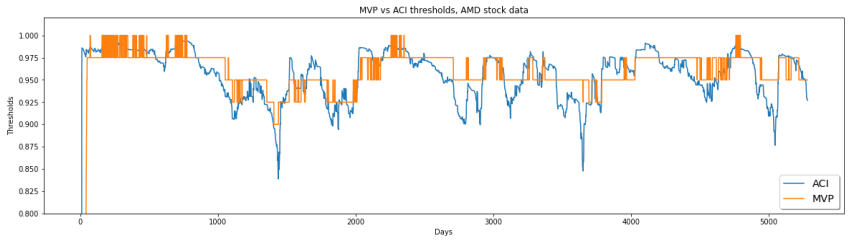


Figure 9: A single trajectory of ACI and MVP thresholds plotted together; for convenience, threshold values are only displayed once MVP and ACI thresholds have risen above 0.8. One can see that MVP and ACI trajectories have somewhat similar shapes, but MVP exhibits a more stable behavior.

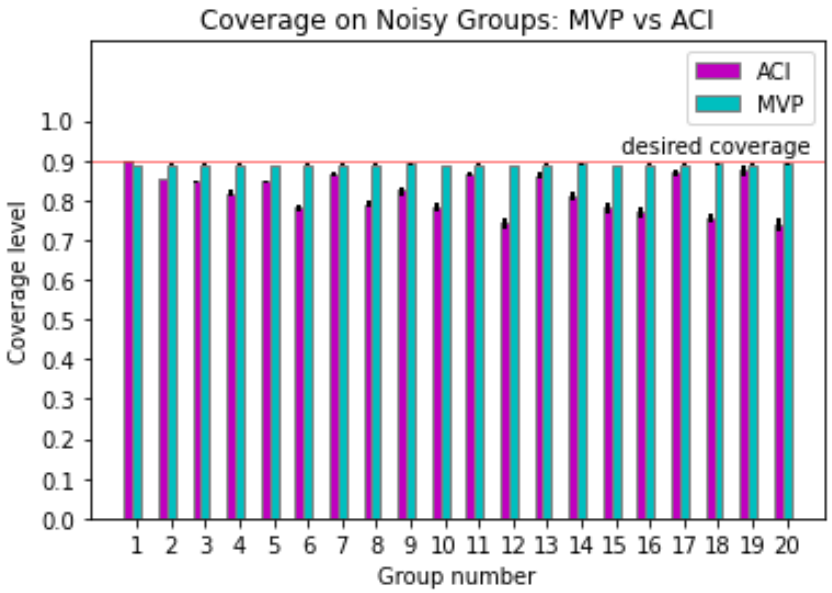
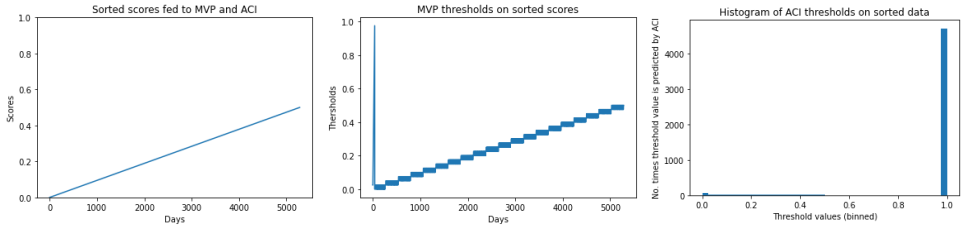


Figure 10: MVP and ACI median coverage (over 20 indep. trials) on groups 1-20 on noisy data (group j consists of days t such that $t \equiv 0 \mod j$). MVP closely matches desired coverage level on all groups, whereas ACI significantly undercovers (within 10-20% from the target). In interpreting the plot, note that Group 1 consists of all of the rounds (and so represents overall marginal coverage), and that each group j consists of a $1/j$ fraction of the data, so the groups become increasingly small from left to right. Note the very small (barely visible) error bars (spanning 25th to 75th quantile coverage): For ACI, the largest error quantile width across groups is 0.0303, whereas for MVP it is even smaller: 0.007.

return r_t on all days t that fall into group G_i, where $\hat{\sigma}_{\text{ret}}$ is the empirical standard deviation of the returns sequence. This noise is additive: so the returns on a day that falls into multiple groups are perturbed by the sum of the group-specific perturbations.

We now run ACI and MVP on the scores produced by GARCH when it is trained on this noisy data. MVP is given the set of groups $\mathcal{G}$. Figure 10 shows a plot of the median coverage rates (over 20 independent trials) for both ACI and MVP on each of the 20 groups. As expected, MVP achieves close to its target coverage on each group, whereas ACI — although getting very close to its target marginal coverage (see group 1) undercovers on most other groups, sometimes significantly as a result of the extra added noise.



(a) (b) (c)

Figure 11: MVP and ACI behavior on a sequence of sorted scores.

B.3.3 Adversarial Ordering

Finally, we present an experiment which tests MVP and ACI on a fully adversarial sequence of conformal scores: a sequence that linearly grows from 0.0 to 0.5 in $T = 5283$ equal steps[10], as shown in Figure 11a. In this ordering, the next score is always larger than the algorithm has ever seen before — hence traditional conformal prediction methods that rely on the exchangeability assumption would obtain 0 coverage on this sequence.

As expected, ACI struggles when it sees scores that are always increasing. ACI and MVP are both guaranteed to approach the target marginal coverage, but this sequence serves to elucidate the difference between simple marginal coverage and threshold calibrated coverage. As can be seen from the histogram in Figure 11c, ACI produces a sequence of thresholds that resembles the uninformative "cheating" strategy outlined in Footnote 1; upon closer inspection, it turns out that ACI predicts the trivial prediction interval (all of $[0, 1]$) on most rounds, and periodically tries to predict lower threshold values (on which it miscovers and is forced back into predicting the full interval). These prediction intervals are not threshold calibrated. In contrast, MVP's sequence of predicted thresholds have to be threshold-calibrated hence (as shown in Figure 11b) they closely track the actual score sequence, resulting in much more informative coverage intervals.

Beyond recognizing that ACI's thresholds, as opposed to MVP's, are uninformative in this setting, we can also see a concrete drawback of ACI's strategy by looking at its average prediction set width. Namely, suppose that the linearly increasing sequence of scores represent *regression scores* $s_t(y_t, \hat{y}_t) = |y_t - \hat{y}_t|$ in a simple regression problem. Then, each threshold q_t generated by ACI or MVP will produce an interval of width $2q_t$. In this case, the average width attained by MVP will be 0.526, whereas the average width attained by ACI will be 1.839. What is more, note that MVP's thresholds closely track the magnitude of the presented sequence of scores, while ACI's threshold is 1 most of the time no matter what subrange of $[0, 1]$ the observed scores are in. Therefore, if we generate increasing scores from 0 to b (above, we took $b = 0.5$), where b can be set arbitrarily small, we will get examples of adversarial data on which the prediction interval widths of ACI are *arbitrarily worse* than the prediction widths produced by MVP.

B.4 A Classification Task: ImageNet

In this section, we compare the performance of MVP against an existing conformal prediction method for the task of generating prediction sets in image classification. The recent work of Angelopoulos et al. [2020] details and implements an algorithm, *Regularized Adaptive Prediction Sets* (RAPS) which, given a trained image classifier, generates small-sized prediction sets of image labels with marginal coverage guarantees. This is done by defining a modified conformal score which empirically produces smaller and more stable sets compared to previously used scores [Romano et al., 2020b].

Using ResNet-152 as the base image classifier, we use calibration data of size 1000 from ImageNet to train RAPS. This same data is used as a "warm-start" training set for MVP (i.e. MVP predicts sets for this data and uses it to update variables used in the algorithm; MVP's performance over these time-steps is not recorded). MVP is run with $m = 40$ calibration buckets. The results shown in Figure 12 detail the performance of both methods (using the same conformal score) on a held-out validation dataset of size 30,000.

[10] That is, the sequence is $\{\frac{0.5 \cdot i}{T-1}\}_{t=0}^{T-1}$, where $T = 5283$ is chosen to be the same as in the above experiments

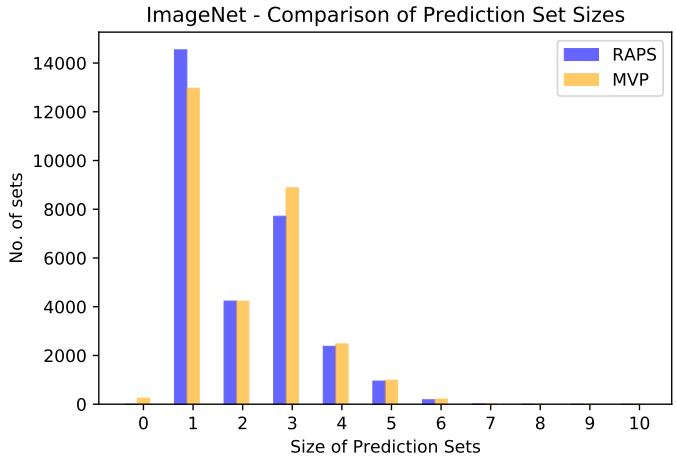


Figure 12: A bar graph showing the size of prediction-sets generated by MVP and RAPS over a dataset of 30,000 images. MVP achieves prediction-set sizes on par with RAPS.

Results The marginal coverage achieved by RAPS across all $T = 30000$ images is 0.90523, and the marginal coverage achieved by MVP is 0.902133. The average prediction-set size for RAPS and MVP are 2.0506 and 2.13986 respectively, and the distribution across prediction-set sizes is similar for both methods. Once again we achieve competitive performance with "traditional" state of the art conformal prediction methods, even in a setting favorable to them (i.e. a setting with i.i.d. data in which only marginal coverage is measured).