Online Multivalid Learning: Means, Moments, and Prediction Intervals

Varun Gupta[1], Christopher Jung[1], Georgy Noarov[1], Mallesh M. Pai[2], and Aaron Roth[1]

[1]University of Pennsylvania Department of Computer and Information Science
[2]Rice University Department of Economics

January 7, 2021

Abstract

We present a general, efficient technique for providing contextual predictions that are "multivalid" in various senses, against an online sequence of adversarially chosen examples (x, y). This means that the resulting estimates correctly predict various statistics of the labels y not just *marginally* — as averaged over the sequence of examples — but also conditionally on $x \in G$ for any G belonging to an arbitrary intersecting collection of groups $\mathcal{G}$.

We provide three instantiations of this framework. The first is mean prediction, which corresponds to an online algorithm satisfying the notion of multicalibration from Hébert-Johnson et al. [2018]. The second is variance and higher moment prediction, which corresponds to an online algorithm satisfying the notion of mean-conditioned moment multicalibration from Jung et al. [2020]. Finally, we define a new notion of prediction interval multivalidity, and give an algorithm for finding prediction intervals which satisfy it. Because our algorithms handle adversarially chosen examples, they can equally well be used to predict statistics of the residuals of arbitrary point prediction methods, giving rise to very general techniques for quantifying the uncertainty of predictions of black box algorithms, even in an online adversarial setting. When instantiated for prediction intervals, this solves a similar problem as conformal prediction, but in an adversarial environment and with multivalidity guarantees stronger than simple marginal coverage guarantees.

Contents

1 Introduction

Consider the problem of making predictions about the prognoses of patients with an infectious disease at the early stages of a pandemic. To be able to guide the allocation of medical interventions, we may want to predict, from each patient's observable features x, things such as the expected severity of the disease y in two days' time. And since we will be using these predictions to allocate scarce resources, we will want to be able to quantify the uncertainty of our predictions: perhaps by providing estimates of the variance of outcomes, or perhaps by providing prediction intervals at a desired level of confidence.

This is an *online* problem because we must start making predictions before we have much data, and the predictions are needed immediately upon the arrival of a patient. It is also a problem in which the environment is rapidly changing: the distribution of patients changes as the disease spreads through different populations, and the conditional distribution on outcomes given features changes as we learn how to better treat the disease.

How can we approach this problem? The *conformal prediction* literature [Shafer and Vovk, 2008] aims to equip arbitrary regression and classification procedures for making point predictions with prediction intervals that contain the true label with (say) 95% probability. But for the application in our example, conformal prediction has two well-known shortcomings:

Marginal Guarantees: Conformal prediction only gives *marginal* prediction intervals: in other words, it provides guarantees that (e.g.) 95% of the prediction intervals produced over a sequence of predictions cover their labels. But these guarantees are averages over what are typically large, heterogeneous populations, and therefore provide little guidance for making decisions about individuals. For example, it would be entirely consistent with the guarantee of a 95% marginal prediction interval $[\ell_t, u_t]$ if for individuals from some demographic group G making up less than 5% of the population, their labels y_t fall outside of $[\ell_t, u_t]$ 100% of the time.[1] One could run many parallel algorithms for different demographic groups G_i, but then there would be no clear way to interpret the many different predictions one would receive for an individual belonging to several demographic groups at once ($x \in G_i$ for multiple groups G_i); for example, prediction intervals corresponding to different demographic groups could be disjoint. To see that marginal guarantees on their own are extremely weak, consider a batch (distributional) setting in which labelled points are drawn from a fixed distribution $\mathcal{D}$: $(x, y) \sim \mathcal{D}$. Then we could provide valid 95% marginal prediction intervals by entirely ignoring the features and giving a fixed prediction interval of $[\ell, u]$ for every point, where $[\ell, u]$ is such that $\Pr_{(x,y)\sim\mathcal{D}}[y \notin [\ell, u]] = 0.05$.

Distributional Assumptions: The conformal prediction literature almost exclusively assumes that the data is drawn from an *exchangeable* distribution (for example, i.i.d. data satisfies this property), and does not offer any guarantees when the data can quickly change in unanticipated or adversarial ways.

In this paper we give techniques for dealing with both of these problems (and similar issues that arise for the problem of predicting label means and higher moments) by drawing on ideas from the literature on *calibration* Dawid [1982], Foster and Vohra [1998]. Calibration is similar to conformal prediction in that it aims to give point estimates in nonparametric settings that satisfy marginal rather than conditional guarantees (i.e. that agree with the true distribution as averaged over the data rather than conditioned on the features of a particular data point). But calibration is concerned with predicting label expectations, rather than giving prediction intervals. Informally speaking, calibrated predictions satisfy that when averaging over all rounds over which the prediction was (approximately) p, the realized labels average to (approximately) p, for all p. Note that in a distributional setting, if a learner truly was predicting the conditional label expectations conditional on features $p_x = \mathbb{E}_{(x,y)\sim\mathcal{D}}[y|x]$, then the forecasts would be calibrated — but just as with marginal prediction intervals, calibration on its own is a very weak condition in a distributional setting. For example, a learner could achieve calibration simply by making a single, constant prediction of

[1] Even more insidious reversals, albeit not in the context of conformal prediction, have been observed on real world data—see the Wikipedia entry for Simpson's paradox (https://en.wikipedia.org/wiki/Simpson%27s_paradox) for several examples.

$p = \mathbb{E}_{(x,y)\sim\mathcal{D}}[y]$ for every point, and so calibrated predictions need not convey much information. Thus, just like the conformal prediction literature, the calibration literature is primarily focused on the online prediction setting. But from early on, the calibration literature has focused on the *adversarial* setting in which no distributional assumptions need to be made at all Foster and Vohra [1998], Fudenberg and Levine [1999a], Sandroni et al. [2003].

Calibration also suffers from the weaknesses that come with marginal guarantees: namely that calibrated predictions may have little to do with the conditional label expectations for members of structured subpopulations. Hébert-Johnson et al. [2018] proposed an elegant solution to this problem in the batch setting, when predicting expectations, which they termed "multicalibration". Informally speaking, a guarantee of multicalibration is parameterized by a large collection of potentially intersecting subsets of the feature space $\mathcal{G}$ (corresponding e.g. to demographic groups or other categories relevant for the prediction task at hand). Multicalibration asks for predictions that are not just calibrated over the full distribution $\mathcal{P}$, but are also simultaneously calibrated over all of the induced distributions that are obtained by conditioning on membership in a set $G \in \mathcal{G}$. Moreover, Hébert-Johnson et al. [2018] showed how to obtain multicalibrated estimators in the batch, distributional setting with sample complexity that depends only logarithmically on $|\mathcal{G}|$. Jung et al. [2020] showed how to extend the notion of (multi)calibration from expectations to variances and other higher moments — and derived algorithms for obtaining such estimates in the batch setting.

1.1 Our Results and Techniques

In this paper, we give a general method for obtaining different kinds of "multivalid" predictions in an online, adversarial setting. This includes mean estimates that satisfy the notion of mean multicalibration from Hébert-Johnson et al. [2018], moment estimates that satisfy the notion of mean-conditioned moment multicalibration from Jung et al. [2020], and prediction intervals which satisfy a new notion of multivalidity, defined in this paper. The latter asks for tight marginal prediction intervals, which are simultaneously valid over each demographic group $G \in \mathcal{G}$. We give a formal definition in Section 2 (and review the definitions of mean and moment multicalibration), but informally, multivalidity for prediction intervals asks, given a target coverage probability $1-\delta$, that for each group $G \in \mathcal{G}$ there be roughly a $1-\delta$-fraction of points (x_t, y_t) with $x_t \in G$ whose label is contained within the predicted interval $(y_t \in [\overline{\ell}_t, \overline{u}_t))$. In fact, we ask for the stronger calibration-like guarantee, that these marginal coverage guarantees hold even conditional on the prediction interval, which (among other things) rules out the trivial solution to marginal coverage that predicts the full interval with probability $1-\delta$ and an empty interval with probability δ. Because our algorithms handle adversarially selected examples, they can equally well be used to augment arbitrary point prediction procedures which give predictions $f_t(x_t) = \hat{y}_t$, independently of how they are trained: We can simply feed our algorithms for multivalid predictions with the *residuals* $\hat{y}_t - y_t$. For example, we can get variance estimates or prediction intervals for the residuals to endow the *predictions* of f_t with uncertainty estimates. Endowing point predictors with prediction intervals in this way provides an alternative to conformal prediction that gives stronger-than-marginal (multivalid) guarantees, under much weaker assumptions (adversarially chosen examples). In general, for each of our techniques, if we instantiate them with the trivial group structure (i.e. one group, containing all points), then we recover standard (or slightly stronger) marginal guarantees: i.e. simple calibrated predictions and simple marginal prediction intervals.[2] But as we enrich our collection of sets $\mathcal{G}$, our guarantees become correspondingly stronger.

The General Strategy We derive our online algorithms using a general strategy that dates back to Fudenberg and Levine [1999a], who used it to give online algorithms for the problem of simple calibration in a setting without features (see also the argument by Sergiu Hart, communicated in Foster and Vohra [1998] and more recently elaborated on in Hart [2020]). In our context, the general strategy proceeds as follows:

[2] In fact, even with the trivial group structure, our guarantees (with appropriately set parameters) remain stronger than marginal coverage. This is because our prediction intervals remain valid even conditioning on the prediction that we made. For example, a prediction interval $[\ell, u)$ is valid not just as averaged over all rounds t, but also as averaged over all rounds t for which we made that specific prediction: $t : [\overline{\ell}_t, \overline{u}_t) = [\ell, u)$.

1. Define a surrogate loss function, such that if the surrogate loss is small at the end of T rounds, then the learner's predictions satisfy our chosen notion of multivalidity over the empirical distribution of the history of the interaction.

2. Argue that if at each round t, the adversary's chosen distribution over labelled examples were known to the learner, then there would be some prediction that the learner could make that would guarantee that the expected increase in the surrogate loss function at that round would be small. This step is often straightforward, because once we fix a known data distribution $\mathcal{D}$, "true distributional quantities" like conditional label expectations, conditional label variances, conditional label quantiles, etc, generally satisfy our corresponding multivalidity desideratum by design.

3. Appeal to the minimax theorem to conclude that there must therefore exist a randomized prediction strategy for the learner that guarantees that the expected increase in the surrogate loss function is small for *any* choice of the adversary.

On its own, carrying out this strategy for a particular notion of multivalidity proves the *existence* of an algorithm that can obtain the appropriate notion of multivalidity against an adversary; but turning it into an actual (and efficient) algorithm requires the ability to *compute* at each round the equilibrium strategy whose existence is shown in Step 3 above.

We instantiate this general strategy in Section 3 for the case of mean multicalibration, which also serves as a template for our derivation and analysis of algorithms for moment multicalibration in Section 4 and prediction interval multivalidity in Section 5. The framework of our analysis is the same in each case, but the details differ: to carry out Step 2, we must bound the value of a different game, and to carry out Step 3, we must solve for the equilibrium of a different game. In each case, we obtain efficient online algorithms for obtaining high probability α-approximate multivalidity bounds (of different flavors), with α scaling roughly as $\alpha \approx \sqrt{\log |\mathcal{G}|/T}$, over interactions of length T — but see Sections 3.2, 4.2, and 5.2 for exact theorem statements. In all cases, our algorithms have per-round runtime that is linear in $|\mathcal{G}|$, and polynomial in the other parameters of the problem. In fact, both our run-time and our convergence bounds can be improved if each individual appears in only a bounded number of groups. Our algorithms can at each step t be implemented in time linear in the number of groups $G \in \mathcal{G}$ that *contain the current example* x_t. This is linear in $|\mathcal{G}|$ in the worst case, but can be substantially smaller. Similarly, we show in Appendix B that if each individual appears in at most d groups, then the $\log |\mathcal{G}|$ term in our convergence bounds can be replaced with $\log(d)$, which gives informative bounds even if $\mathcal{G}$ is infinitely large. Without assumptions of this sort, running time that is polynomial in $|\mathcal{G}|$ (rather than logarithmic in $|\mathcal{G}|$, as our convergence bounds are) is necessary in the worst case, even for mean multicalibration in the offline setting, as shown by Hébert-Johnson et al. [2018].

Adapting the original approach of Fudenberg and Levine [1999a] runs into several obstacles, stemming from the fact that the *action space* of both the learner and the adversary and the *number of constraints* defining our calibration desideratum are both much larger in our setting. Consider the case of mean prediction — in which the goal is to obtain calibrated predictions. In the featureless setting studied by Fudenberg and Levine [1999a], the action space for the learner corresponds to a discretization of the real unit interval $[0, 1]$, and the action space of the adversary is binary. In our setting, in which data points are endowed with features from a large feature space $\mathcal{X}$, the learner's action space corresponds to the set of all *functions* mapping $\mathcal{X}$ to $[0, 1]$, and the adversary's action space corresponds to the set of all labelled examples $\mathcal{X} \times [0, 1]$. Similarly, for simple calibration, the number of constraints is equal to the chosen discretization granularity of the unit interval $[0, 1]$, whereas in our case, the number of constraints also grows linearly with $|\mathcal{G}|$, the number of groups over which we want to be able to promise guarantees.

Convergence Rates and Sample Complexity The surrogate loss function used by Fudenberg and Levine [1999a] bounds the ℓ_2 calibration error — i.e. the average squared violation of all of the constraints used to define calibration. Because all of the notions of multivalidity that we consider consist of a set of constraints of size scaling linearly with $|\mathcal{G}|$, if we were to attempt to bound the ℓ_2 violation of our multivalidity constraints, we would necessarily obtain convergence bounds that scale polynomially with $|\mathcal{G}|$. Instead we use

a different surrogate loss function — a sign-symmetrized version of an exponential soft-max — that can be used to bound the ℓ_∞ violation of our multivalidity constraints, and allows us to obtain bounds that scale only logarithmically with $|\mathcal{G}|$. For moment multicalibration, we face the further complication of needing to define a potential function bounding a linear surrogate for what is ultimately a nonlinear measure of distributional fidelity. An outline of the specific new ideas needed here can be found in Section 4.1. For interval multivalidity, we face the further complication that tight prediction intervals need not exist even in the distributional setting, for worst-case distributions. An outline of the new ideas we need to overcome this can be found in Section 5.1. Finally, we note that ℓ_∞ violation is consistent with how the existing literature on batch multicalibration [Hébert-Johnson et al., 2018] has quantified approximation guarantees. In fact, by using standard online-to-offline reductions, we are able to derive new, optimal sample complexity bounds for mean and moment multicalibration for the *batch distributional* setting in Appendix A that improve on the sample complexity bounds given in Hébert-Johnson et al. [2018], Jung et al. [2020]. This is because when applied to the batch setting, our online algorithms take only a single pass through the data, and avoid issues related to adaptive data re-use that complicated previous algorithms in the batch setting.

Computation of Equilibrium Strategies To compute equilibria of the large action space games we define, we do not attempt to directly compute or represent the function that we use at each round t to map features to labels. Instead, we represent this function implicitly by "lazily" solving a smaller equilibrium computation problem only after we have observed the adversary's choice of feature vector x (but before we have observed the label y) to compute a distribution over predictions. We show in each of our three settings that this computation is tractable. In the case of mean multicalibration, we are able to analytically derive a simple algorithm for sampling from this equilibrium strategy, presented in Section 3.3. For mean-conditioned k^{th} moment multicalibration we show that the equilibrium can be found using a linear program with polynomially many variables and 2^k+1 constraints. For the most interesting cases, k is a small constant (e.g. for variance, $k=2$, and so the linear program has only 5 constraints). Even when k is large, we show that this linear program has a separation oracle that runs in time $O(k)$, and so it can be solved efficiently via the Ellipsoid algorithm. We show in Appendix C that there always exists an equilibrium for the learner with support over at most $k+1$ many predictions, limiting the extent to which it needs to deploy randomization. Finally, for prediction interval multivalidity, we show in Section 5.3 that we can express the equilibrium computation problem as a linear program. Although the linear program is naively defined by infinitely many constraints, we show that it can ultimately be represented with only finitely many constraints, and that it has an efficient separation oracle, so can be solved in polynomial time using the Ellipsoid algorithm.

Advantages of Conformal Prediction We have thus far emphasized the advantages that our techniques have over conformal prediction — but we also want to highlight the strengths of conformal prediction relative to our work, and directions for future improvement. Conformal prediction aims to obtain marginal coverage with respect to some (unknown) underlying distribution. As a result of the distributional assumption, it is able to obtain coverage (over the randomness of the distribution) at a rate of coverage $1-\delta+O(1/T)$ [Lei et al., 2018]. In contrast, in our setting, there is no underlying distribution. We therefore give guarantees on *empirical coverage* — i.e the fraction of labels that our predicted intervals have covered in the realized sequence of examples. As a result, our coverage bounds necessarily have error terms that tend to 0 at a rate of $O(1/\sqrt{T})$, over sequences of length T. We note that conformal prediction methods also obtain *empirical* coverage on the order of $1-\delta \pm O(1/\sqrt{T})$, as our methods do [Lei et al., 2018]. Conformal prediction methods naturally give one sided coverage error on the distribution (i.e. the coverage probability is always $\geq 1-\delta$), whereas as we present our bounds, our empirical coverage has two sided error. We note that there is a simple but inelegant way to use our techniques to obtain one sided coverage: run our algorithms with coverage parameter $1-\delta' = 1-\delta/2$, and predict trivial coverage intervals until our error bounds are $\leq \delta/2$[3]. Techniques from the conformal prediction literature also can be applied to very general label domains $\mathcal{Y}$, and can be used to produce very general kinds of prediction *sets*. In our paper, we restrict attention to real-valued labels $\mathcal{Y}=[0,1]$ and prediction *intervals*. We do not believe that there are any

[3]Restarting periodically with δ' closer to δ if we want to asymptotically converge to exact coverage

fundamental obstacles to generalizing our techniques to other label domains and prediction sets, and this is an interesting direction for future work. Finally, the conformal prediction literature has developed a number of very simple, practical techniques. In this paper, we give polynomial time algorithms, of varying complexity. Our algorithm for mean multicalibration in Section 3 is very simple to implement, but our algorithm for multivalid interval prediction in Section 5 requires solving a linear program with a separation oracle. Another important direction for future work is reducing the complexity of our techniques, and doing empirical evaluations.

1.2 Additional Related Work

Work on calibrated mean prediction dates back to Dawid [1982]. Foster and Vohra [1998] were the first to show that in the online setting without features, it is possible to obtain asymptotic calibration even against an adversary. Once this initial result was proven, a number of proofs of it were given using different techniques, including Blackwell's approachability theorem [Foster, 1999] and a non-constructive minimax argument (originally communicated verbally by Sergiu Hart, appearing first in Foster and Vohra [1998], and more recently formalized in Hart [2020]). This argument was "non-constructive" because it was a minimax argument over the entire algorithm design space. Fudenberg and Levine [1999a] gave a more tractable per-round minimax argument, which we adapt to our work — although they were satisfied with an existential argument, and do not derive a concrete algorithm. The algorithm we give for online multicalibration is similar to the algorithm given by Foster and Hart [2019] for the simple calibration problem in the special case of a featureless setting and the trivial group structure. Lehrer [2001], Sandroni et al. [2003] (and in a slightly different context, Fudenberg and Levine [1999b]) generalized this literature and showed that it was possible to extend these ideas in order to satisfy dramatically more demanding notions of calibration (e.g. calibration on all computable subsequences of rounds). This line of work primarily gives limit results via non-constructive arguments without establishing rates. There are two notable exceptions. Foster et al. [2011] give a non-constructive argument establishing that it is possible to obtain mean calibration loss $\tilde{O}(\sqrt{\frac{\log K}{T}})$ with respect to a set of K "checking rules" which define subsequences over which the algorithm must be calibrated. These results are derived in a setting without features x, but we believe their techniques could be used to establish the same convergence bounds that we do, for mean multicalibration: $\alpha = \tilde{O}(\sqrt{\frac{\log |\mathcal{G}|}{T}})$. Foster and Kakade [2006] give an efficient algorithm based on ridge-regression which can be used to achieve what we call *mean consistency*[4] on a collection of sets $\mathcal{G}$ with error rates converging as $\alpha = \tilde{O}(\sqrt{\frac{|\mathcal{G}|}{T}})$. Their algorithm is deterministic, which in particular means it cannot be used to achieve the standard notion of calibration, which can only be achieved by randomized algorithms in adversarial environments [Oakes, 1985]. It can be used to achieve what is called "weak calibration" by Kakade and Foster [2004] and "smooth calibration" by Foster and Hart [2018] — a relaxation that can be obtained by deterministic algorithms. In comparison, our algorithm for mean multicalibration achieves the standard notion of calibration with the optimal sample complexity dependence on $\log |\mathcal{G}|$, while simultaneously being explicitly defined and computationally efficient.

There has also been a recent resurgence of interest in calibration in the computer science community, in part motivated by fairness concerns [Kleinberg et al., 2016, Chouldechova, 2017, Pleiss et al., 2017]. It is from this literature that the original proposal for multicalibration arose [Hébert-Johnson et al., 2018], as well as the related notion of multiaccuracy [Hébert-Johnson et al., 2018, Kim et al., 2019]. Shabat et al. [2020] prove uniform convergence bounds for multicalibrated predictors, Dwork et al. [2019] draw connections between multicalibrated predictors and notions of fair rankings, and Dwork et al. [2020] define a notion of outcome indistinguishability related to distribution testing, and show close connections to multicalibration. Jung et al. [2020] extend the notion of mean calibration to variances and higher moments, and give efficient algorithms for learning moment multicalibrated predictors. Jung et al. [2020] also show that their moment predictors can be used to derive *conservative* multivalid prediction intervals, using Chebyshev's inequality and generalizations to higher moments. In general, however, these moment-based inequalities give intervals

[4]This is also what is known as *multi-accuracy* in [Hébert-Johnson et al., 2018, Kim et al., 2019].

that may cover their label much more frequently than the target $1-\delta$ coverage probability, and cannot achieve the kinds of tight multicoverage guarantees that we obtain in this work. All of this work operates in the batch, distributional setting. Recently, Qiao and Valiant [2020] proved lower bounds for simple mean calibration in the online setting, showing that no algorithm can obtain rates better than $O(T^{-0.472})$ against an adversary. At first blush, our upper bounds appear to contradict these lower bounds — but they do not, because we study convergence in the ℓ_∞ sense, whereas they study it in the ℓ_1 sense.

Conformal prediction is motivated similarly to calibration, but aims to produce marginal prediction intervals rather than mean estimates — see Shafer and Vovk [2008] for an overview. The problems that we highlight — namely, that marginal guarantees are weak, and that this literature relies on strong distributional assumptions — have been noted before. For example, Foygel Barber et al. [2020] prove that even in the distributional setting, *conditional* prediction intervals are impossible to provide, and aim instead for a goal that is similar to ours: providing marginal prediction intervals that are valid as averaged over a large number of subgroups $\mathcal{G}$. They take a conservative approach, by using a holdout set to estimate empirical prediction intervals separately for each group, and then taking the union of all of these prediction intervals over the demographic groups of a new individual. The result is that their prediction intervals — unlike ours — do not become tight, even in the limit. Chernozhukov et al. [2018] consider the problem of conformal prediction for time series data, for which the exchangeability assumption may not hold. They show that if the data comes from a rapidly mixing process (so that, in particular, points that are well separated in the sequence are approximately independent) then it is still possible to obtain approximate marginal coverage guarantees. Tibshirani et al. [2019] consider the problem of conformal prediction under *covariate shift*, in which the marginal distribution on features $\mathcal{X}$ differs between the training and test distributions, but the conditional distribution on labels $\mathcal{Y}|\mathcal{X}$ remains the same. They show how to adapt techniques from conformal prediction when the likelihood ratio between the training and test distribution is known. In the distributional setting, Gupta et al. [2020] have proven close relationships between calibration, confidence intervals, and prediction intervals.

Finally, the notion of multicalibration is related to subgroup fairness notions [Kearns et al., 2018, 2019, Kim et al., 2018] that ask for statistical "fairness" constraints of various sorts (beyond calibration) to hold across all subgroups defined by some rich class $\mathcal{G}$. See Chouldechova and Roth [2020] for a survey.

2 Preliminaries

2.1 Notation

We write $\mathcal{X}$ to denote a feature domain and $\mathcal{Y} = [0,1]$ to denote a label domain. We write $\mathcal{G} \subseteq 2^{\mathcal{X}}$ to denote a collection of subsets of $\mathcal{X}$. Given any $x \in \mathcal{X}$, we write $\mathcal{G}(x)$ for the set of groups that contain x, i.e. $\mathcal{G}(x) = \{G \in \mathcal{G} : x \in G\}$. Given an integer T we write $[T]$ to denote the set of integers $[T] = \{1, \ldots, T\}$. In general, we denote our random variables with tildes (e.g. $\tilde{X}$, $\tilde{Y}$) to distinguish them from their realizations (denoted e.g. X, Y). Given a finite set A, we write ΔA for the probability simplex over the elements in A.

2.2 Online Prediction

Online (contextual) prediction proceeds in rounds that we index by $t \in [T]$, for a given finite horizon T. In each round, an interaction between a *learner* and an *adversary* proceeds as follows. In each round t:

1. The *adversary* chooses a joint distribution over feature vectors $x_t \in \mathcal{X}$ and labels $y_t \in \mathcal{Y}$. The learner receives x_t (a realized feature vector), but no information about y_t is revealed.

2. The *learner* chooses a distribution over predictions $p_t \in \mathcal{P}$. (We will consider several different kinds of predictions in this paper, and so are agnostic to the domain of the prediction for now — we use $\mathcal{P}$ as a generic domain).

3. The learner observes y_t (a realized label).

For an index $s \in [T]$, we denote by π_s the *transcript* of the interaction in rounds $t = 1$ through s: $\pi_s = ((x_t, p_t, y_t))_{t=1}^s$. We write Π^* as the domain of all transcripts.

Formally, the adversary is modelled as a probabilistic mapping $\text{Adv} : \Pi^* \to \Delta(\mathcal{X} \times \mathcal{Y})$ from transcripts to distributions over labelled data points, and the learner is modeled as a mapping $\text{Learn} : \Pi^* \to (\mathcal{X} \to \Delta\mathcal{P})$ from transcripts to a probabilistic mapping from feature vectors to distributions over predictions. An adversary may be either unconstrained (free to play any point in $\Delta(\mathcal{X} \times \mathcal{Y})$) or constrained to choose from some specified subset of $\Delta(\mathcal{X} \times \mathcal{Y})$. Fixing both a learner and an adversary induces a probability distribution over transcripts. Our goal is to derive particular learning algorithms, and to prove that various kinds of bounds hold either in expectation, or with high probability over the randomness of the transcript, in the worst case over transcript distributions, where we quantify over all possible adversaries.

Given a transcript π_T, a group $G \in \mathcal{G}$ and a set of rounds $S \subseteq [T]$, we write

$$G_S = \{t \in S : x_t \in G\}.$$

In words, this is the set of rounds in S in which the realized feature vectors in the transcript belonged to G. When it is clear from context, we sometimes overload notation, and for a group $G \in \mathcal{G}$, and a period $s \leq T$, write G_s to denote the set of data points (indexed by their rounds) in a transcript π_s that are members of the group G:

$$G_s = \{t \in [s] : x_t \in G\}.$$

2.2.1 Types of Predictions, and Notions of Validity

We consider three types of predictions in this paper: Mean predictions, pairs of mean and higher moment predictions (e.g. variance), and prediction intervals.

Mean Predictions For mean predictions, the prediction domain will be the unit interval: $\mathcal{P}_{\text{mean}} = [0, 1]$. The learner will select $p_t \equiv \overline{\mu}_t \in \mathcal{P}_{\text{mean}}$ in each round t, with the goal of predicting the conditional label expectation $\mathbb{E}[y_t | x_t]$. For any subset of days $S \subseteq [T]$, we write

$$\mu(S) = \frac{1}{|S|} \sum_{t \in S} y_t, \quad \overline{\mu}(S) = \frac{1}{|S|} \sum_{t \in S} \overline{\mu}_t$$

to denote the true label population mean conditional on $t \in S$ and the average of our mean estimates over days $t \in S$, respectively. We will ask for our predictions to satisfy large numbers of *mean consistency* constraints: that the conditional label averages be (approximately) equal to conditional prediction averages over different sets S.

Definition 2.1 (Mean Consistency)**.** *Given a transcript π_T, we say that the mean predictions $\{\overline{\mu}_t\}_{t=1}^T$ are α-mean consistent on $S \subseteq [T]$, if*

$$|\mu(S) - \overline{\mu}(S)| \leq \alpha \frac{T}{|S|}.$$

Remark 2.1. *Note the scaling with both T and $|S|$. If $S = [T]$, then this condition simply asks for the true label mean and the average prediction to be within α of one another, as averaged over the entire transcript. For smaller sets, the allowable error grows with the inverse of $\frac{|S|}{T}$ — i.e. the measure of S within the uniform distribution over the transcript. Even in a distributional setting, estimates inevitably degrade with the size of the set we are conditioning on, and our formulation corresponds exactly to how mean consistency is defined in Jung et al. [2020]. Our definitions are also consistent with how the literature on online calibration quantifies calibration error with respect to subsequences. Hébert-Johnson et al. [2018] handle this issue slightly differently, by asking for uniform bounds, but in the end proving bounds only for sets S that have sufficient mass γ in the underlying probability distribution. In the batch setting, our formulation can recover bounds that are strictly stronger than those of Hébert-Johnson et al. [2018] after a reparametrization $\alpha \leftarrow \gamma\alpha$.*

Next, we define multicalibration in our setting. Informally, a sequence of mean predictions is *calibrated* if the average realized label y_t on all days for which $\overline{\mu}_t$ is (roughly) p is (roughly) p. The need to consider days in which the prediction was *roughly* p arises from the fact that a learning algorithm will not necessarily ever make the same prediction twice. More generally, by bucketing predictions at a fixed granularity, we can guarantee that the average number of predictions within each bucket grows linearly with T.

To collect mean predictions $\overline{\mu}_t$ that are approximately equal to p for each p, we group real-valued predictions into n buckets of width $\frac{1}{n}$. Here n is a parameter controlling the coarseness of our calibration guarantee. For any coarseness parameter n and bucket index $i \in [n-1]$, we write $B_n(i) = \left[\frac{i-1}{n}, \frac{i}{n}\right)$ and $B_n(n) = \left[\frac{n-1}{n}, 1\right]$ so that these buckets partition the unit interval. Conversely, given a $\overline{\mu} \in [0,1]$, define $B_n^{-1}(\overline{\mu}) \in [n]$ in the obvious way i.e. $B_n^{-1}(\overline{\mu}) = i$ where i is such that $\overline{\mu} \in B_n(i)$. When clear from the context, we elide the subscript n and write $B(i)$ and $B^{-1}(\overline{\mu})$.

For any $S \subseteq [T]$ and $i \in [n]$, we write

$$S(i) = \{t \in S : \overline{\mu}_t \in B_n(i)\}.$$

In words, $S(i)$ corresponds to the subset of rounds in S where the mean prediction falls in the i^{th} bucket.

(Simple) calibration asks for the sequence of predictions to be α-mean-consistent on all sets $[T](i)$ for $i \in [n]$ — i.e. for the subset of rounds in which the prediction fell into the i^{th} bucket, for all i. Multicalibration asks for the predictions to be calibrated not just on the overall sequence, but also simultaneously on all the subsequences corresponding to each group $G \in \mathcal{G}$. In our notation, it asks for mean consistency on each set $G(i)$, for every group $G \in \mathcal{G}$ and $i \in [n]$.

Definition 2.2 (Mean-Multicalibration)**.** *Given a transcript π_T, we say that the mean predictions $\{\overline{\mu}_t\}_{t=1}^T$ are* (α, n)-mean multicalibrated *with respect to $\mathcal{G}$ if we have that for every $G \in \mathcal{G}$ and $i \in [n]$, the mean-predictions are α-mean consistent on $G_T(i)$:*

$$|\mu(G_T(i)) - \overline{\mu}(G_T(i))| \leq \alpha \frac{T}{|G_T(i)|}.$$

Remark 2.2. *Note that we define mean multicalibration (and our other notions of multivalidity, shortly) to have two parameters: n, which controls the coarseness of the guarantee, and α, which controls the error of the guarantee. These parameters can be set independently — in the sense that we will be able to achieve (α, n) mean multicalibration for any pair (α, n) — but they should be interpreted together. For example, to avoid the trivial solution in which the learner simply selects uniformly at random at each iteration (thereby guaranteeing that $|G_T(i)| \leq T/n$ for all G, i), we should set $\alpha \ll \frac{1}{n}$.*

(Mean, Moment) Predictions In this case, the prediction domain is the product of the unit interval with itself: $\mathcal{P}_{(\text{mean,moment})} = [0,1] \times [0,1]$. In each round t, the learner selects $p_t = (\overline{\mu}_t, \overline{m}_t^k)$ with the goal of matching $\mathbb{E}[y_t|x_t]$ and $\mathbb{E}[(y_t - \mathbb{E}[y_t|x_t])^k|x_t]$ respectively — the conditional label expectation, and its conditional k^{th} central moment. For simplicity, we assume throughout that k is even, so the k^{th} moment has nonnegative range, but there is no obstacle other than notation to handling odd moments as well.

We group continuous predictions $(\overline{\mu}, \overline{m}^k)$ into a finite set of discrete buckets—again, defined with respect to a pair of discretization parameters n and n'. Recall our bucketing notation for mean prediction: for any $i \in [n-1]$, we wrote $B_n(i) = \left[\frac{i-1}{n}, \frac{i}{n}\right)$ and $B_n(n) = \left[\frac{n-1}{n}, 1\right]$. Here we generalize this notation to pairs, and write for any $i \in [n]$ and $j \in [n']$:

$$B_{n,n'}(i,j) = \{(a,b) \in [0,1] \times [0,1] : a \in B_n(i), b \in B_{n'}(j)\}.$$

If $n = n'$, we will write $B_n(i,j)$. Once again, when n and n' are clear from the context, we may elide the subscript (n, n') entirely.

Analogously to our notation for mean prediction, for any $S \subseteq [T]$ we write

$$m^k(S) = \frac{1}{|S|}\sum_{t \in S}(y_t - \mu(S))^k, \quad \overline{m}^k(S) = \frac{1}{|S|}\sum_{t \in S}\overline{m}_t^k$$

for the empirical k^{th} central moment of the label distribution on the subsequence S, and for the average of the moment prediction on S, respectively. Just as with mean consistency, moment consistency asks that these two quantities be approximately equal on a set S.

Definition 2.3 (Moment Consistency)**.** *Given a transcript π_T, we say that moment predictions $\{\overline{m}_t^k\}_{t=1}^T$ are α-moment consistent on set $S \subseteq [T]$ if*

$$|m^k(S) - \overline{m}^k(S)| \leq \alpha \frac{T}{|S|}.$$

It is not sensible to ask for moment consistency on arbitrary sets S, because higher central moments are not linear, and so even true conditional label moments would not satisfy moment consistency conditions on arbitrary sets S. True conditional label moments *do* satisfy moment consistency on sets of points x that share the same label mean, however, and so this is what we will ask of our predictions as well (See Jung et al. [2020] for an extensive discussion of this condition and its applications). To that end, for any $S \subseteq [T]$ and $i \in [n], j \in [n']$, we write

$$S(i,j) = \left\{t \in S : (\overline{\mu}_t, \overline{m}_t^k) \in B_{n,n'}(i,j)\right\}.$$

In words, $S(i,j)$ corresponds to the subset of rounds in S in which our predicted mean and moment fall into the bucket $B_{n,n'}(i,j)$.

Definition 2.4 (Mean-Conditioned Moment Multicalibration)**.** *Given a transcript π_T, we say that the (mean, moment) predictions $\{(\overline{\mu}_t, \overline{m}_t^k)\}_{t=1}^T$ are (α, β, n, n')*-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$*, if for every $i \in [n], j \in [n']$ and $G \in \mathcal{G}$, we have that the mean predictions are α-mean consistent on $G_T(i,j)$ and the moment predictions are β-moment consistent on $G_T(i,j)$:*

$$|\mu(G_T(i,j)) - \overline{\mu}(G_T(i,j))| \leq \alpha \frac{T}{|G_T(i,j)|},$$
$$|m^k(G_T(i,j)) - \overline{m}^k(G_T(i,j))| \leq \beta \frac{T}{|G_T(i,j)|}.$$

Interval Predictions In this case, the prediction domain is the set of ordered pairs of endpoints in the unit interval: $\mathcal{P}_{\text{interval}} = \{(\ell, u) : \ell \leq u,\ u, \ell \in [0,1]\}$. Given a pair $(\ell, u) \in \mathcal{P}_{\text{interval}}$, we say that it *covers* a label $y \in [0,1]$ if y falls between ℓ and u, which we write as $\text{Cover}((\ell, u), y) = 1$. To avoid issues of "double counting", we define coverage in the same manner as we defined our bucketing, using intervals that are closed on the left but open on the right, with the exception of $u = 1$:

$$\text{Cover}((\ell, u), y) = \begin{cases} \mathbb{1}(y \in [\ell, u)) & \text{if } u < 1, \\ \mathbb{1}(y \in [\ell, u]) & \text{if } u = 1. \end{cases}$$

In each round t, we will predict an interval $p_t = (\overline{\ell}_t, \overline{u}_t)$ with the goal of achieving $\mathbb{E}[\text{Cover}((\overline{\ell}_t, \overline{u}_t), y)|x_t] = 1 - \delta$ for some target coverage probability $1 - \delta \in [0,1]$. We again bucket our coverage intervals using a discretization parameter n, using the same notation as for moment predictions.

For any $S \subseteq [T]$ and $i \leq j \in [n]$, we write

$$S(i,j) = \left\{t \in S : (\overline{\ell}_t, \overline{u}_t) \in B_n(i,j)\right\}.$$

In words, $S(i,j)$ corresponds to the subset of rounds in S in which our predicted interval's endpoints are in buckets i and j, respectively. We can now define multivalidity analogously to how we defined multicalibration.

For any $S \subseteq [T]$, we write

$$\overline{H}(S) = \frac{1}{|S|} \sum_{t \in S} \text{Cover}((\overline{\ell}_t, \overline{u}_t), y_t).$$

Definition 2.5. *We say that interval predictions* $\{(\overline{\ell}_t, \overline{u}_t)\}_{t=1}^T$ *are* α-consistent *on set* S *with respect to failure probability* $\delta \in (0,1)$*, if the following holds:*

$$|\overline{H}(S) - (1-\delta)| \leq \alpha \frac{T}{|S|}.$$

Definition 2.6. *Given a transcript* π_T*, we say that the interval predictions are* (α, n)-multivalid with respect to δ and $\mathcal{G}$*, if for every* $i \leq j \in [n]$ *and* $G \in \mathcal{G}$*, we have that the interval predictions are* α*-consistent on* $G_T(i,j)$ *with respect to coverage probability* $1-\delta$*:*

$$|\overline{H}(G_T(i,j)) - (1-\delta)| \leq \alpha \frac{T}{|G_T(i,j)|}.$$

2.3 Zero-sum Games

Our analysis will hinge on properties of zero-sum games, and in particular on the minimax theorem.

Definition 2.7. *A zero-sum game is defined by:*

1. *A* minimization *player with a convex and compact strategy space* $\mathcal{Q}_1 \subseteq \mathbb{R}^{d_1}$ *for some* $d_1 \in (0, \infty)$*.*
2. *A* maximization *player with a convex and compact strategy space* $\mathcal{Q}_2 \subseteq \mathbb{R}^{d_2}$ *for some* $d_2 \in (0, \infty)$*.*
3. *An* objective function $u : \mathcal{Q}_1 \times \mathcal{Q}_2 \to \mathbb{R}$*, concave in its first argument and convex in its second argument.*

Zero-sum games are often defined by endowing each player with a finite set of *pure* strategies X_1, X_2. The convex compact strategy sets $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are then formed by allowing players to randomize over their pure strategies and taking $\mathcal{Q}_1 = \Delta X_1$, $\mathcal{Q}_2 = \Delta X_2$ to be the probability simplices over the pure strategies of each player. An objective function $u : X_1 \times X_2 \to \mathbb{R}$ can be linearly extended to $\mathcal{Q}_1$ and $\mathcal{Q}_2$ in the natural way (i.e. by taking expectations over the randomized strategies of each player) – i.e. for any $Q_1 \in \mathcal{Q}_1$ and $Q_2 \in \mathcal{Q}_2$, we write $u(Q_1, Q_2) = \mathbb{E}_{x_1 \sim Q_1, x_2 \sim Q_2}[u(x_1, x_2)]$.

In a zero-sum game, the minimization player chooses some action $Q_1 \in \mathcal{Q}_1$ and the maximization player chooses some action $Q_2 \in \mathcal{Q}_2$, resulting in objective value $u(Q_1, Q_2)$. The goal of the minimization player is to minimize the objective value, and the goal of the maximization player is to maximize it. The key property of zero-sum games, first proved by von Neumann for the case of games with finite sets of pure strategies and generalized to general zero-sum games of the form considered in Definition 2.7 by Sion, is that the order of play does not affect the objective value that each player can guarantee. This is captured in the minimax theorem, which says that whether the minimization player *first* gets to observe the strategy of the maximization player, and *then* best respond, or whether she must first announce her strategy and allow the maximization player to best respond, she is able to guarantee herself the same value.

Theorem 2.1 (Sion's Minimax Theorem)**.** *For any zero-sum game* $(\mathcal{Q}_1, \mathcal{Q}_2, u)$*:*

$$\min_{Q_1 \in \mathcal{Q}_1} \max_{Q_2 \in \mathcal{Q}_2} u(Q_1, Q_2) = \max_{Q_2 \in \mathcal{Q}_2} \min_{Q_1 \in \mathcal{Q}_1} u(Q_1, Q_2).$$

The minimax theorem justifies the following definitions:

Definition 2.8 (Value, Equilibrium, and Best Response)**.** *The* value *of a zero-sum game* $(\mathcal{Q}_1, \mathcal{Q}_2, u)$ *is the unique* $v \in \mathbb{R}$ *such that*

$$\min_{Q_1 \in \mathcal{Q}_1} \max_{Q_2 \in \mathcal{Q}_2} u(Q_1, Q_2) = \max_{Q_2 \in \mathcal{Q}_2} \min_{Q_1 \in \mathcal{Q}_1} u(Q_1, Q_2) = v.$$

We say that a strategy for the minimization player $Q_1^* \in \mathcal{Q}_1$ *is a (minimax) equilibrium strategy if it guarantees that the objective value is at most the value of the game, for any strategy* $Q_2 \in \mathcal{Q}_2$ *of the maximization player:*

$$\max_{Q_2 \in \mathcal{Q}_2} u(Q_1^*, Q_2) = v.$$

We say that Q_2 *is a best response for the maximization player in response to* Q_1^* *if it realizes the above maximum.*

In our analysis, we will identify the Learner with the minimization player and the Adversary with the maximization player, and so will denote their strategy spaces as $\mathcal{Q}^L$ and $\mathcal{Q}^A$ respectively.

3 Online Mean Multicalibration

In this section, we show how to obtain mean multicalibrated estimators in an online adversarial setting. Our derivation also serves as a warm up example of our general technique, which we also instantiate (in somewhat more involved settings) in Sections 4 and 5 to derive online algorithms for mean-conditioned moment multicalibrated estimators and for multivalid prediction intervals respectively.

3.1 An Outline of Our Approach

At a high level, the derivation of our algorithm and its proof of correctness proceeds as follows:

1. For each group $G \in \mathcal{G}$, $i \in [n]$, and transcript π_s up to period s, we define an empirical quantity $V_s^{G,i}$ (Definition 3.1) which represents the calibration error that our algorithm has incurred with respect to group G over those of the rounds 1 through s when the i^{th} bucket was predicted. These quantities are defined so that if for each G and i, $|V_T^{G,i}|$ is small, then our algorithm is approximately multicalibrated with respect to $\mathcal{G}$ across T rounds.

   The premise of our algorithm will be to greedily make decisions at each round s so as to minimize the *maximum possible increase* of these quantities ($\max_{G,i} |V_{s+1}^{G,i}| - \max_{G,i} |V_s^{G,i}|$), in the worst case over the choices of the adversary. If we could bound this quantity at every round, then by telescoping, we would have a bound on $\max_{G,i} |V_T^{G,i}|$ at the end of the interaction, and therefore a guarantee of mean multicalibration.

2. The increase in the maximum value of $|V_{s+1}^{G,i}|$ is inconvenient to work with, and so we instead define a smooth potential function L_s (Definition 3.2) corresponding to a soft-max function which upper bounds $\max_{G,i} |V_s^{G,i}|$. Our design goal instead becomes to upper bound the increase in our potential function from round to round: $\Delta_{s+1} = L_{s+1} - L_s$. We view this as defining a zero-sum game, in which the learner's goal is to minimize this increase, and the adversary's goal is to maximize it.

3. We show that for each fixed distribution that the adversary could employ at each round $s+1$, there is a prediction the learner could employ (if only she knew the adversary's distribution) that would guarantee that the increase in potential Δ_{s+1} is small. Intuitively, this is because if we knew the true joint distribution over feature label pairs, then we could predict the true conditional expectations, $\overline{\mu}_{s+1} = \mathbb{E}[y_{s+1}|x_{s+1}]$, which would be perfectly calibrated on all groups. Of course, the learner does not have the luxury of knowing the adversary's distribution before choosing her own. But this thought experiment establishes the value of the game, and so we can conclude via the minimax theorem that there must be some fixed *distribution* over prediction rules that the learner can play that will guarantee Δ_{s+1} being small against *all* actions of the adversary.

4. Step 3 suffices to argue for the *existence* of an algorithm obtaining multicalibration guarantees (Algorithm 1). However, to actually derive an implementable algorithm we need to find a way to compute the equilibrium strategy at each round, whose existence was argued in Step 3. A priori, this seems daunting because the learner's strategy space consists of all randomized mappings between $\mathcal{X}$ and $\mathcal{Y}$, and the adversary's strategy space consists of all joint distributions on $\mathcal{X} \times \mathcal{Y}$. However, we derive a simple algorithm in Section 3.3 that implements the optimal equilibrium strategy needed to realize Step 3. Informally, we are able to do so by representing the mapping between $\mathcal{X}$ and $\mathcal{Y}$ only implicitly, and delaying all computation until x_t has been chosen. We then show that the equilibrium strategy for the learner has a simple structure and randomizes over only at most 2 predictions. Our final algorithm (Algorithm 2) simply computes the relevant portion of the equilibrium strategy at each round and then samples from it.

5. To apply the minimax theorem, and to derive a concrete algorithm, we need to restrict our algorithm to making predictions in $[0,1]$ that are discretized at units of $1/rn$ for some $r > 1$. This parameter r appears in our final bounds, but neither the runtime of our algorithm nor our convergence rate has any dependence on r, and so it can be imagined to be arbitrarily small. Taking it to be $r = 1/\sqrt{T}$ causes it to become a low order term in our final bounds.

Finally, in Appendix A, we give a standard online-to-offline conversion to show how to use our Algorithm 2 to solve offline (batch) multicalibration problems. This gives optimal sample complexity bounds for the offline problem, yielding an improvement over those proven in Hébert-Johnson et al. [2018], Jung et al. [2020]. The crux of the improvement is that unlike the algorithms given in Hébert-Johnson et al. [2018], Jung et al. [2020], our algorithm takes only a single pass over the data, and so avoids complications that arise from data re-use. However, unlike previous batch algorithms which make deterministic predictions, the batch algorithm that we obtain through this reduction makes randomized predictions.

3.2 An Existential Derivation of the Algorithm and Multicalibration Bounds

We begin by defining notation $V_s^{G,i}$ for the (unnormalized) *portion* of the mean calibration error corresponding to each group $G \in \mathcal{G}$ and bucket $i \in [n]$:

Definition 3.1. *Given a transcript $\pi_s = ((x_t, \overline{\mu}_t, y_t))_{t=1}^s$, we define the mean calibration error for a group $G \in \mathcal{G}$ and bucket $i \in [n]$ at time s to be:*

$$V_s^{G,i}(\pi_s) = |G_s(i)| \left(\mu\left(G_s(i)\right) - \overline{\mu}\left(G_s(i)\right)\right) = \sum_{t=1}^{s} \mathbb{1}[\overline{\mu}_t \in B(i), x_t \in G]\left(y_t - \overline{\mu}_t\right) \tag{1}$$

When the transcript is clear from context we will sometimes simply write $V_s^{G,i}$.

Observe that our definition of mean multicalibration (Definition 2.2) corresponds to asking that $|V_s^{G,i}|$ be small for all i, G.

Observation 3.1. *Fix a transcript π_T. If for all $G \in \mathcal{G}$, $i \in [n]$, we have that:*

$$\left|V_T^{G,i}\right| \leq \alpha T,$$

then the corresponding sequence of predictions is (α, n)-mean multicalibrated with respect to $\mathcal{G}$.

We next define a surrogate loss function that we can use to bound our calibration error.

Definition 3.2 (Surrogate loss function)**.** *Fixing a transcript $\pi_s \in \Pi^*$ and a parameter $\eta \in [0, \frac{1}{2}]$, define a surrogate calibration loss function at day s as:*

$$L_s(\pi_s) = \sum_{\substack{G \in \mathcal{G}, \\ i \in [n]}} \left(\exp(\eta V_s^{G,i}) + \exp(-\eta V_s^{G,i})\right).$$

When the transcript π_s is clear from context, we will sometimes simply write L_s.

We will leave η unspecified for now, and choose it later to optimize our bounds. Observe that this "soft-max style" function allows us to tightly upper bound our calibration loss:

Observation 3.2. *For any transcript π_T, and any $\eta \in [0, \frac{1}{2}]$, we have that:*

$$\max_{G \in \mathcal{G}, i \in [n]} \left|V_T^{G,i}\right| \leq \frac{1}{\eta} \ln(L_T) \leq \max_{G \in \mathcal{G}, i \in [n]} \left|V_T^{G,i}\right| + \frac{\ln\left(2|\mathcal{G}|n\right)}{\eta}.$$

Part of our analysis will depend on viewing the transcript as a random variable: in this case, in keeping with our convention for random variables, we refer to it as $\tilde{\pi}$. The associated random variables tracking calibration and surrogate loss are denoted $\tilde{V}$ and $\tilde{L}$ respectively.

Our goal is to find a strategy for the learner that guarantees that our surrogate loss L_T remains small. Towards this end, we define $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1})$ to be the expected increase in the surrogate loss function in the event that the adversary plays feature vector x_{s+1} *and* the learner plays prediction $\overline{\mu}_{s+1}$. Here the expectation is over the only remaining source of randomness after the conditioning — the distribution over labels y_{s+1} (which we observe is determined, once we fix π_s and x_{s+1}).

Definition 3.3 (Conditional Change in Surrogate Loss)**.**

$$\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1}) = \mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\left[\tilde{L}_{s+1} - L_s \middle| x_{s+1}, \overline{\mu}_{s+1}, \pi_s\right].$$

We begin with a simple bound on $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1})$:

Lemma 3.1. *For any transcript $\pi_s \in \Pi^*$, any $x_{s+1} \in \mathcal{X}$, and any $\overline{\mu}_{s+1} \in \mathcal{P}_{mean}$ such that $\overline{\mu}_{s+1} \in B(i)$ for some $i \in [n]$:*

$$\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1}) \leq \eta\left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[\tilde{y}_{s+1}] - \overline{\mu}_{s+1}\right) C_s^i(x_{s+1}) + 2\eta^2 L_s,$$

where for each $i \in [n]$:

$$C_s^i(x_{s+1}) \equiv \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i}) - \exp(-\eta V_s^{G,i}). \tag{2}$$

Proof. Fix any transcript $\pi_s \in \Pi^*$ (which defines L_s), feature vector $x_{s+1} \in \mathcal{X}$, and $\overline{\mu}_{s+1}$ such that $\overline{\mu}_{s+1} \in B(i)$ for some $i \in [n]$. By direct calculation, we obtain:

$$\begin{aligned}
&\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1}) \\
=&\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\left[\sum_{G\in\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i})\left(\exp(\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1})) - 1\right) + \exp(-\eta V_s^{G,i})\left(\exp(-\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1})) - 1\right)\right], \\
\leq&\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\left[\sum_{G\in\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i})\left(\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1}) + 2\eta^2\right) + \exp(-\eta V_s^{G,i})\left(-\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1}) + 2\eta^2\right)\right], \\
=&\eta\left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[\tilde{y}_{s+1}] - \overline{\mu}_{s+1}\right) \sum_{G\in\mathcal{G}(x_{s+1})} \left(\exp(\eta V_s^{G,i}) - \exp(-\eta V_s^{G,i})\right) + 2\eta^2 \sum_{G\in\mathcal{G}(x_{s+1})} \left(\exp(\eta V_s^{G,i}) + \exp(-\eta V_s^{G,i})\right), \\
\leq&\eta\left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[\tilde{y}_{s+1}] - \overline{\mu}_{s+1}\right)\left(\sum_{G\in\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i}) - \exp(-\eta V_s^{G,i})\right) + 2\eta^2 L_s, \\
=&\eta\left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[\tilde{y}_{s+1}] - \overline{\mu}_{s+1}\right) C_s^i(x_{s+1}) + 2\eta^2 L_s.
\end{aligned}$$

Here, the first inequality follows from the fact that for $0 < |x| < \frac{1}{2}$, $\exp(x) \leq 1 + x + 2x^2$. ∎

Using this bound, we define a zero-sum game between the learner and the adversary and use the minimax theorem to conclude that the learner always has a strategy that guarantees that the per-round increase in surrogate loss can be bounded. To satisfy the convexity and compactness requirements of the minimax theorem, it will be convenient for us to imagine that the learner's pure strategy space is a finite, discrete subset of $\mathcal{P}_{\text{mean}} = [0, 1]$. To this end, we define the following discretization for any $r \in \mathbb{N}$ (here n is the discretization parameter we use to define the coarseness of our bucketing):

$$\mathcal{P}^{rn} = \left\{0, \frac{1}{rn}, \frac{2}{rn}, \ldots, 1\right\}.$$

We use this discretization also in our algorithm in Section 3.3 — but we remark at the outset that the need to discretize is only for technical reasons, and our algorithm will have no dependence — neither in runtime nor in its convergence rate — on the value of r that we choose, so we can imagine the discretization to be arbitrarily fine.

To simplify notation, for each $\overline{\mu} \in \mathcal{P}^{rn}$, define $C_s^{\overline{\mu}} \equiv C_s^i$ where $i \in [n]$ s.t. $\overline{\mu} \in B_n(i)$.

Lemma 3.2. *For any transcript $\pi_s \in \Pi^*$, any $x_{s+1} \in \mathcal{X}$, and any $r \in \mathbb{N}$ there exists a distribution over predictions for the learner $Q_{s+1}^L \in \Delta\mathcal{P}^{rn}$, such that regardless of the adversary's choice of distribution of y_{s+1} over $\Delta\mathcal{Y}$, we have that:*

$$\mathop{\mathbb{E}}_{\overline{\mu} \sim Q_{s+1}^L} [\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu})] \leq L_s \left(\frac{\eta}{rn} + 2\eta^2 \right).$$

Proof. We define a zero-sum game played between the learner (the minimization player) and the adversary (the maximization player). The learner's pure strategy space is the set of discrete predictions $X_1 = \mathcal{P}^{rn}$. The adversary's pure strategy space is (a priori) the set of all distributions over labels in $[0, 1]$. However, we will observe in a moment that the objective function of our game depends only on the *expected value* of the label, and so without loss of generality, we will be able to take the adversary's full strategy space to be the set of all pure strategies, i.e., $\mathcal{Q}^A = [0, 1]$ (which is closed and convex), because it already spans the set of realizable expectations. As usual, we take the learner's full strategy space to be the set of distributions over pure strategies: $\mathcal{Q}^L = \Delta\mathcal{P}^{rn}$.

Fix the transcript π_s and the feature vector x_{s+1}. We define the objective of this game to be the upper bound we proved on $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu})$ in Lemma 3.1. For each $\overline{\mu} \in \mathcal{P}^{rn}$ and each $y \in [0, 1]$, we let:

$$u(\overline{\mu}, y) = \eta \, (y - \overline{\mu}) \, C_s^{\overline{\mu}}(x_{s+1}) + 2\eta^2 L_s.$$

Note that for any distribution over labels y of the adversary, the expected objective value depends on his strategy only through $\mathbb{E}[\tilde{y}]$ because the above objective function is linear in y: that is, $\mathbb{E}_{\tilde{y}}[u(\overline{\mu}, \tilde{y})] = u(\overline{\mu}, \mathbb{E}[\tilde{y}])$. Thus we are justified in our reduced-form representation of the adversary's full strategy as choosing $\mathbb{E}[\tilde{y}]$ in the interval $[0, 1]$.

We now establish the value of this game. Observe that for any strategy of the adversary (which fixes $\mathbb{E}[\tilde{y}]$), the learner can respond by playing $\overline{\mu}^* = \operatorname{argmin}_{\overline{\mu} \in \mathcal{P}^{rn}} |\mathbb{E}[\tilde{y}] - \overline{\mu}|$, and that because of our discretization, $\min |\mathbb{E}[\tilde{y}] - \overline{\mu}^*| \leq \frac{1}{rn}$. Therefore, the value of the game is at most:

$$\begin{aligned} \max_{y \in [0,1]} \min_{\overline{\mu}^* \in \mathcal{P}^{rn}} u(\overline{\mu}^*, y) &\leq \max_{\overline{\mu} \in \mathcal{P}^{rn}} \frac{\eta}{rn} \left| C_s^{\overline{\mu}}(x_{s+1}) \right| + 2\eta^2 L_s, \\ &\leq L_s \left(\frac{\eta}{rn} + 2\eta^2 \right). \end{aligned}$$

Here the latter inequality follows since $C_s^{\overline{\mu}}(x_{s+1}) \leq L_s$ for all $\overline{\mu} \in \mathcal{P}^{rn}$, by observation. We can now apply the minimax theorem (Theorem 2.1) to conclude that there exists a fixed distribution $Q_{s+1}^L \in \mathcal{Q}^L$ for the learner that guarantees that simultaneously for *every* label $y \in [0, 1]$ that might be chosen by the adversary:

$$\mathop{\mathbb{E}}_{\overline{\mu} \sim Q_{s+1}^L} [u(\overline{\mu}, y)] \leq L_s \left(\frac{\eta}{rn} + 2\eta^2 \right),$$

as desired. ∎

Corollary 3.1. *For every $r \in \mathbb{N}$, $s \in [T]$, $\pi_s \in \Pi^*$, and $x_{s+1} \in \mathcal{X}$ (which fixes L_s and Q_{s+1}^L), and any distribution over $\mathcal{Y}$:*

$$\mathop{\mathbb{E}}_{\overline{\mu}_{s+1} \sim Q_{s+1}^L} [\tilde{L}_{s+1} | \pi_s] = L_s + \mathop{\mathbb{E}}_{\overline{\mu}_{s+1} \sim Q_{s+1}^L} [\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1})] \leq L_s \left(1 + \frac{\eta}{rn} + 2\eta^2 \right).$$

Lemma 3.2 defines (existentially) an algorithm that the learner can use to make predictions—Algorithm 1. We will now show that Algorithm 1 (if we could compute the distributions Q_t^L) results in multicalibrated predictions. In Section 3.3 we show a simple and efficient method for sampling from Q_t^L.

Algorithm 1: A Generic Multicalibrator

for $t = 1, \ldots, T$ **do**
Observe x_t. Given π_{t-1} and x_t, let $Q_t^L \in \mathcal{Q}_t^L$ be the distribution over predictions whose existence is established in Lemma 3.2.
Sample $\overline{\mu} \sim Q_t^L$ and predict $\overline{\mu}_t = \overline{\mu}$

We now prove two convergence bounds for Algorithm 1. The first will bound its multicalibration error *in expectation*, and the other will provide a high probability bound. To show these bounds, we first state a helper theorem that will be useful not just in this section, but also in deriving the final convergence bounds for the algorithms presented in Sections 4 and 5. The proof is in Appendix D.

Theorem 3.1. *Consider a nonnegative random process $\tilde{X}_t$ adapted to the filtration $\mathcal{F}_t = \sigma(\pi_t)$, where $\tilde{X}_0$ is constant a.s. Suppose we have that for any period t, and any π_{t-1}, $\mathbb{E}[\tilde{X}_t|\pi_{t-1}] \leq X_{t-1}(1 + \eta c + 2\eta^2)$ for some $\eta \in [0, \frac{1}{2}], c \in [0, 1]$. Then we have that:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{X}_T] \leq X_0 \exp\left(T\eta c + 2T\eta^2\right). \tag{3}$$

Further, define a process $\tilde{Z}_t$ adapted to the same filtration by $\tilde{Z}_t = Z_{t-1} + \ln \tilde{X}_t - \mathbb{E}[\ln(\tilde{X}_t)|\pi_{t-1}]$. Suppose that $|Z_t - Z_{t-1}| \leq 2\eta$, where $Z_0 = 0$ a.s. Then, with probability $1 - \lambda$,

$$\ln(X_T(\pi_T)) \leq \ln(X_0) + T\left(\eta c + 2\eta^2\right) + \eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}. \tag{4}$$

We are now ready to bound our multicalibration error. As a straightforward consequence of Corollary 3.1 and the first part of Theorem 3.1, we have the following Corollary.

Corollary 3.2. *Against any adversary, Algorithm 1 instantiated with discretization parameter r results in surrogate loss satisfying:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T] \leq 2|\mathcal{G}|n \exp\left(\frac{T\eta}{rn} + 2T\eta^2\right).$$

Proof. Note that the first part of Theorem 3.1 applies to the process L with $L_0 = 2|\mathcal{G}|n$ and $c = \frac{1}{rn}$. The bound follows by plugging these values into (3). □

Next, we can convert this into a bound on Algorithm 1's expected calibration error:

Theorem 3.2. *When Algorithm 1 is run using n buckets for calibration, discretization $r \in \mathbb{N}$, and $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n)}{2T}} \in (0, 1/2)$, then against any adversary, its sequence of mean predictions is (α, n)-multicalibrated with respect to $\mathcal{G}$, where:*

$$\mathbb{E}[\alpha] \leq \frac{1}{rn} + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n)}{T}}.$$

For $r = \frac{\sqrt{T}}{\epsilon n\sqrt{2\ln(2|\mathcal{G}|n)}}$ this gives:

$$\mathbb{E}[\alpha] \leq (2 + \epsilon)\sqrt{\frac{2}{T}\ln(2|\mathcal{G}|n)}.$$

Here the expectation is taken over the randomness of the transcript π_T.

Proof. From Observation 3.1, it suffices to show that

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G\in\mathcal{G},i\in[n]}|\tilde{V}_T^{G,i}|\right] \le \frac{1}{rn} + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n)}{T}}.$$

We begin by computing a bound on the (exponential of) the expectation of this quantity:

$$\begin{aligned}
\exp\left(\eta \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}|\tilde{V}_T^{G,i}|\right]\right) &\le \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\exp\left(\eta\max_{G,i}|\tilde{V}_T^{G,i}|\right)\right],\\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}\exp\left(\eta|\tilde{V}_T^{G,i}|\right)\right],\\
&\le \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}\left(\exp\left(\eta\tilde{V}_T^{G,i}\right)+\exp\left(-\eta\tilde{V}_T^{G,i}\right)\right)\right],\\
&\le \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\sum_{G,i}\left(\exp\left(\eta\tilde{V}_T^{G,i}\right)+\exp\left(-\eta\tilde{V}_T^{G,i}\right)\right)\right],\\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T],\\
&\le 2|\mathcal{G}|n\exp\left(\frac{T\eta}{rn}+2T\eta^2\right).
\end{aligned}$$

Here the first step is by Jensen's inequality and the last one follows from Corollary 3.2. Taking the logarithm of both sides and dividing by ηT, we have

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}|\tilde{V}_T^{G,i}|\right] \le \frac{\ln(2|\mathcal{G}|n)}{\eta T} + \frac{1}{rn} + 2\eta.$$

Choosing $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n)}{2T}}$, we thus obtain the desired inequality

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}|\tilde{V}_T^{G,i}|\right] \le \frac{1}{rn} + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n)}{T}}. \quad \square$$

Now, given $\tilde{L}$, let us define its associated martingale process $\tilde{Z}$ as in the second part of Theorem 3.1. The next lemma shows that the increments of $\tilde{Z}$ are uniformly bounded over all rounds t. The proof is in Appendix D.

Lemma 3.3. *At any round $t \in [T]$ and for any realized transcript π_t, $|Z_t - Z_{t-1}| \le 2\eta$.*

We can now use the second part of Theorem 3.1 to prove a high probability bound on the multicalibration error of Algorithm 1.

Theorem 3.3. *When Algorithm 1 is run using n calibration buckets, discretization $r \in \mathbb{N}$ and $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n)}{2T}} \in (0, 1/2)$, then against any adversary, its sequence of mean predictions is α-multicalibrated, with respect to $\mathcal{G}$ with probability $1-\lambda$ over the randomness of the transcript π_T, for*

$$\alpha \le \frac{1}{rn} + 4\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n}{\lambda}\right)}.$$

Choosing $r = \frac{\sqrt{T}}{\epsilon n\sqrt{2\ln(2|\mathcal{G}|n/\lambda)}}$, this gives:

$$\alpha \le (4+\epsilon)\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n}{\lambda}\right)}.$$

Proof. By Lemma 3.3, the second part of Theorem 3.1 applies; plugging in $L_0 = 2|\mathcal{G}|n$ and $c = \frac{1}{rn}$, we have:

$$\ln(L_T(\pi_T)) \leq \ln(2|\mathcal{G}|n) + T\left(\frac{\eta}{rn} + 2\eta^2\right) + \eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}.$$

Now, note that

$$\begin{aligned}
\exp\left(\eta \max_{G,i} |V_T^{G,i}|\right) &= \max_{G,i} \exp\left(\eta |V_T^{G,i}|\right), \\
&\leq \max_{G,i} \left(\exp\left(\eta V_T^{G,i}\right) + \exp\left(-\eta V_T^{G,i}\right)\right), \\
&\leq \sum_{G,i} \left(\exp\left(\eta V_T^{G,i}\right) + \exp\left(-\eta V_T^{G,i}\right)\right), \\
&= L_T(\pi_T).
\end{aligned}$$

Taking log on both sides and dividing both sides by ηT, we get

$$\frac{1}{T}\max_{G,i} |V_T^{G,i}| \leq \frac{1}{\eta T}\ln(L_T(\pi_T)) \leq \frac{\ln(2|\mathcal{G}|n)}{\eta T} + \frac{1}{rn} + 2\eta + \sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}}.$$

Choosing $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n)}{2T}}$, we thus obtain the desired inequality

$$\frac{1}{T}\max_{G,i} |V_T^{G,i}| \leq \frac{1}{rn} + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n)}{T}} + \sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}} \leq \frac{1}{rn} + 4\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n}{\lambda}\right)}.$$ □

Remark 3.1. *In both Theorems 3.2 and 3.3, the dependence on* $\log(|\mathcal{G}|)$ *can be replaced with a dependence on* $\log(d)$ *under the assumption that* $|\mathcal{G}(x_t)| \leq d$ *for all* t *— i.e. that each observed data point is contained in only boundedly many groups. This gives us non-trivial guarantees even when* $\mathcal{G}$ *is infinitely large. See Appendix B for details.*

3.3 Deriving an Efficient Algorithm via Equilibrium Computation

Algorithm 2: Von Neumann's Mean Multicalibrator(η, n, r)

for $t = 1, \ldots, T$ **do**
  Observe x_t and compute for each $i \in [n]$ $C_{t-1}^i(x_t)$ as defined in (2).
  if $C_{t-1}^i(x_t) > 0$ for all $i \in [n]$ **then**
    Predict $\overline{\mu}_t = 1$.
  else if $C_{t-1}^i(x_t) < 0$ for all $i \in [n]$ **then**
    Predict $\overline{\mu}_t = 0$.
  else
    Find $i^* \in [n-1]$ such that $C_{t-1}^{i^*}(x_t) \cdot C_{t-1}^{i^*+1}(x_t) \leq 0$
    Define $0 \leq q_t \leq 1$ such that $q_t C_{t-1}^{i^*}(x_t) + (1-q_t)C_{t-1}^{i^*+1}(x_t) = 0$. In other words, define it as follows (using the convention that $0/0 = 1$):

$$q_t = \frac{|C_{t-1}^{i^*+1}(x_t)|}{|C_{t-1}^{i^*+1}(x_t)| + |C_{t-1}^{i^*}(x_t)|}.$$

    Predict $\overline{\mu}_t = \frac{i^*}{n} - \frac{1}{rn}$ with probability q_t and $\overline{\mu}_t = \frac{i^*}{n}$ with probability $1 - q_t$.

In Section 3.2, we derived Algorithm 1 and proved that it results in mean multicalibrated predictions. However, Algorithm 1 was not defined explicitly: it relies on the distributions Q_t^L, whose existence we showed in Lemma 3.2 but which we did not explicitly construct. In this section, we derive a scheme for sampling from these distributions Q_t^L, which leads to Algorithm 2 — an explicit, efficient implementation of Algorithm 1.

Theorem 3.4. *Algorithm 2 implements Algorithm 1. In particular it obtains the multicalibration guarantees proven in Theorems 3.2 and 3.3.*

Proof. Recall that Algorithm 1 samples at every round $s+1$ from a distribution Q_{s+1}^L that is a minimax equilibrium strategy of a game between the learner and the adversary, with objective function

$$u(\overline{\mu}, y) = \eta\,(y - \overline{\mu})\,C_s^{\overline{\mu}}(x_{s+1}) + 2\eta^2 L_s.$$

The equilibrium structure of the game is preserved under positive affine transformations, so instead we consider

$$u(\overline{\mu}, y) = (y - \overline{\mu})\,C_s^{\overline{\mu}}(x_{s+1}).$$

We wish to find a distribution $Q_{s+1}^L \in \mathcal{Q}^L$ that guarantees — against any strategy of the adversary — an objective value that is at most the bound on the value of the game we proved in Lemma 3.2. For the transformed game, this bound is:

$$\max_{y\in[0,1]} \mathop{\mathbb{E}}_{\overline{\mu}\sim Q_{s+1}} [u(\overline{\mu}, y)] \le \frac{1}{rn} L_s.$$

We can start by characterizing the best response of the adversary.

Observation 3.3. *For any $Q^L \in \mathcal{Q}^L$:*

$$\max_{y\in[0,1]} \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} [u(\overline{\mu}, y)] = \left(\mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} [C_s^{\overline{\mu}}(x_{s+1})] \right)^+ - \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} \left[\overline{\mu} C_s^{\overline{\mu}}(x_{s+1}) \right],$$

where $(x)^+ = \max(x, 0)$.

Proof. Note that:

$$\begin{aligned} u(\mu, y) &= (y - \overline{\mu})\,C_s^{\overline{\mu}}(x_{s+1}) \\ &= y C_s^{\overline{\mu}}(x_{s+1}) - \overline{\mu} C_s^{\overline{\mu}}(x_{s+1}). \end{aligned}$$

Observe that only the first term depends on y. Therefore, if the learner plays according to Q^L, then the adversary will choose y so as to maximize the linear expression $y\,\mathbb{E}_{\overline{\mu}\sim Q^L}[C_s^{\overline{\mu}}(x_{s+1})]$. This is always maximized either at $y = 0$ or $y = 1$. It is maximized at $y = 1$ when $\mathbb{E}_{\overline{\mu}\sim Q^L}[C_s^{\overline{\mu}}(x_{s+1})] > 0$, and at $y = 0$ otherwise. ☐

Finally, we can reduce the analysis to three disjoint cases:

1. $C_s^i(x_{s+1}) > 0$ for all $i \in [n]$: Then for any distribution Q^L, by Observation 3.3 we have:

$$\max_{y\in[0,1]} \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} [u(\overline{\mu}, y)] = \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} [C_s^{\overline{\mu}}(x_{s+1})] - \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} \left[\overline{\mu} C_s^{\overline{\mu}}(x_{s+1}) \right].$$

   In this case, letting Q^L be a point mass on $\overline{\mu} = 1$ achieves a value of $0 < \frac{1}{rn} L_s$.

2. $C_s^i(x_{s+1}) < 0$ for all $i \in [n]$: Then for any distribution Q^L, by Observation 3.3 we have:

$$\max_{y\in[0,1]} \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} [u(\overline{\mu}, y)] = - \mathop{\mathbb{E}}_{\overline{\mu}\sim Q^L} \left[\overline{\mu} C_s^{\overline{\mu}}(x_{s+1}) \right]$$

   In this case, letting Q^L be a point mass on $\overline{\mu} = 0$ achieves a value of $0 < \frac{1}{rn} L_s$.

3. In the remaining case, there must exist some index $i^* \in [n-1]$ such that either $C_s^{i^*}(x_{s+1})$ and $C_s^{i^*+1}(x_{s+1})$ have opposite signs, or such that at least one of them takes value exactly zero. Randomizing as in the algorithm results in:

$$\begin{aligned}
&\max_{y\in[0,1]} \mathop{\mathbb{E}}_{\overline{\mu}\sim Q_{s+1}^L}[u(\overline{\mu}, y)] \\
=&\left(\mathop{\mathbb{E}}_{\overline{\mu}\sim Q_{s+1}^L}\left[C_s^{\overline{\mu}}(x_{s+1})\right]\right)^+ - E_{\overline{\mu}\sim Q_{s+1}^L}\left[\overline{\mu} C_s^{\overline{\mu}}(x_{s+1})\right] \\
=&\left(q_{s+1}C_s^{i^*}(x_{s+1}) + (1-q_{s+1})C_s^{i^*+1}(x_{s+1})\right)^+ - \left(q_{s+1}\left(\tfrac{i^*}{n} - \tfrac{1}{rn}\right)C_s^{i^*}(x_{s+1}) + (1-q_{s+1})\tfrac{i^*}{n}C_s^{i^*+1}(x_{s+1})\right) \\
=&\frac{1}{rn}C_s^{i^*}(x_{s+1}) \\
\leq&\frac{1}{rn}L_s.
\end{aligned}$$

Algorithm 2 plays according to this distribution Q_{s+1}^L at every round, which completes the proof. □

Running Time Our algorithm is elementary, and given values for $C_{t-1}^i(x_t)$, it runs in time per iteration which is linear in the number of buckets n. For large collections of groups $\mathcal{G}$, the bulk of the computational cost is due to the first step of Algorithm 2, in which we compute the quantities $C_{t-1}^i(x_t)$ as in Equation 2:

$$C_{t-1}^i(x_t) \equiv \sum_{\mathcal{G}(x_t)} \exp(\eta V_{t-1}^{G,i}) - \exp(-\eta V_{t-1}^{G,i})$$

These quantities are a sum over every group $G \in \mathcal{G}$ such that $x_t \in G$. In the worst case, we can compute this by enumerating over all such groups, and we obtain runtime that is linear in $|\mathcal{G}|$. However, for any class $\mathcal{G}$ such that we can efficiently enumerate the set of groups containing x_t (i.e. $\mathcal{G}(x_t)$), our per-round runtime is only linear in $|\mathcal{G}(x_t)|$, which may be substantially smaller than $|\mathcal{G}|$. For example, this property holds for collections $\mathcal{G}$ of groups induced by conjunctions or disjunctions of binary features. Finally, we observe that our runtime is entirely independent of the choice of the discretization parameter r.

4 Online Moment Multicalibration

4.1 An Outline of Our Approach

In this section, we derive an online algorithm for supplying mean and k^{th}-moment predictions that are mean-conditioned moment multicalibrated with respect to some collection of groups $\mathcal{G}$, as defined in Definition 2.4. We follow the same basic strategy that we developed in Section 3 for making multicalibrated mean predictions. In particular, the first few steps of our approach exactly mirror the approach in Section 3: Analogously to Steps 1 and 2 of Section 3.1 we define calibration losses and a convenient soft-max style surrogate loss function and bound the increase to that surrogate loss function at each round. However, we make a couple of important deviations.

1. The first complication that arises is that moment consistency is not a linearly separable constraint across rounds (because moments are nonlinear). However, we are able to define linearly separable "pseudo-moment" consistency losses M and prove in Lemma 4.1 that if both our pseudo-moment consistency losses M and our mean consistency losses V are small then our predictions are mean-conditioned moment multicalibrated.

2. The next complication arises when we attempt to define a zero-sum game using our bound on the per-round increase of the surrogate loss. The bound on the loss that we obtain for mean-conditioned

moment multicalibration is nonlinear in both the learner's (mean) prediction and the adversary's choice of label y. We cannot directly apply a minimax theorem because the necessary concavity and convexity conditions are not satisfied. Our argument instead requires a change of variables: we show that in the game we define, the adversary's payoff, fixing the strategy of the learner, is linear in the first k (uncentered) moments of the distribution over the labels chosen by the adversary. We also expand the strategy space of the adversary to allow him to pick k arbitrary real numbers, representing the first k centered moments of his label distribution, unencumbered by the requirement that these chosen values actually correspond to the moments of any real label distribution. Enlarging the adversary's strategy space in this way can only *increase* the value of the game, and so the upper bounds we prove on the value of this simplified game continue to hold for the original game. Moreover, a minimax theorem applies to this transformed game, and therefore guarantees the *existence* of a prediction strategy for the learner that is approximately mean-conditioned moment multicalibrated.

3. In order to implement this strategy with an explicit efficient prediction algorithm, we need to solve a game in which the learner has r^2nn' pure strategies. Doing this naively would inherit a running time dependence on r, a discretization parameter that we want to take to be very small. However, we prove a "structure theorem" about the enlarged game described above: that without loss of generality, the learner need only randomize over a support of at most $4nn'$ pure strategies. With this structure theorem in hand, we show that the equilibrium computation problem can be cast as a linear program with $4nn'$ variables and 2^k+1 constraints. If k is a small constant (e.g. $k=2$ for variance multicalibration), then this linear program can be explicitly described and solved. But even when k is too large to enumerate all 2^k constraints, we show that there is a separation oracle that runs in time $O(k)$, allowing us to efficiently solve this linear program using the Ellipsoid algorithm. In Appendix C, we show that there exist solutions to the learner's problem that have small support—in which the learner mixes over at most $k+1$ strategies.

4.2 An Existential Derivation of the Algorithm and Moment Multicalibration Bounds

We will calibrate our mean predictions $\{\overline{\mu}_t\}_{t=1}^T$ over n buckets, and k^{th} moment predictions $\{\overline{m}^k\}_{t=1}^T$ over $n' < n$ buckets. As before, we introduce notation to denote the *portion* of the mean calibration error corresponding to each pair of buckets (i,j) and group G, and consider a similar quantity that serves as a proxy for the portion of the moment calibration error corresponding to each group $G \in \mathcal{G}$ and buckets $i \in [n]$, $j \in [n']$. We will need an extra piece of notation: for any $i \in [n]$, define $\hat{\mu}_i \equiv \frac{2i-1}{2n}$. For any $i \in [n]$ and $\overline{\mu} \in B_n(i)$, we abuse notation and write $\hat{\mu}_{\overline{\mu}} = \hat{\mu}_i$.

Definition 4.1. *Given a transcript $\pi_s = ((x_t, (\overline{\mu}_t, \overline{m}_t^k), y_t))_{t=1}^s$, for each group $G \in \mathcal{G}$ and buckets $i \in [n], j \in [n']$ at time s, we write*

$$V_s^{G,i,j}(\pi_s) = \sum_{t=1}^{s} \mathbb{1}[\overline{\mu}_t \in B_n(i), \overline{m}_t^k \in B_n(j), x_t \in G]\,(y_t - \overline{\mu}_t)\,,$$

$$M_s^{G,i,j}(\pi_s) = \sum_{t=1}^{s} \mathbb{1}[\overline{\mu}_t \in B_n(i), \overline{m}_t^k \in B_n(j), x_t \in G]\left((y_t - \hat{\mu}_i)^k - \overline{m}_t^k\right).$$

When the transcript π_s is clear from context we will simply write $V_s^{G,i,j}, M_s^{G,i,j}$.

In words, $V_s^{G,i,j}$ calculates the difference between the true mean and the mean of our predictions over the subset of periods up to s in which the realized feature vector was in group G and the learner predicted a mean $\overline{\mu} \in B_n(i)$ and a moment $\overline{m}^k \in B_{n'}(j)$. $M_s^{G,i,j}$ defines a similar quantity for moments — but not exactly. Instead of calculating the empirical moment around the empirical mean (i.e. $(y_t - \mu(G_s(i,j)))^k$), we center around $\hat{\mu}_i$, i.e. the middle of the bucket $B_n(i)$. We do this to make $M_s^{G,i,j}$ linearly separable across rounds.

We show, using an argument similar[5] to Jung et al. [2020], that if our mean predictions are sufficiently calibrated — which ensures $\hat{\mu}_i \approx \mu(G_T(i,j))$ — then we can still bound the mean-conditioned moment multicalibration error through our proxy quantity $M_s^{G,i,j}$.

Lemma 4.1. *For a given $i \in [n], j \in [n']$ and $G \in \mathcal{G}$, if $\frac{1}{T}|V_T^{G,i,j}| \le \alpha$, $\frac{1}{T}|M_T^{G,i,j}| \le \beta$, then we have*

$$|\mu(G_T(i,j)) - \overline{\mu}(G_T(i,j))| \le \frac{\alpha T}{|G_T(i,j)|}, \tag{Mean Consistency}$$

$$\left|m^k(G_T(i,j)) - \overline{m}^k(G_T(i,j))\right| \le \frac{(\beta + k\alpha + \frac{k}{2n})T}{|G_T(i,j)|}. \tag{Moment Consistency}$$

Proof. It is easy to see mean-consistency:

$$\frac{|G_T(i,j)|}{T}\,|\overline{\mu}(G_T(i,j)) - \mu(G_T(i,j))| = \frac{1}{T}\left|\sum_{t \in G_T(i,j)} (\overline{\mu}_t - y_t)\right| = \frac{1}{T}|V_T^{G,i,j}| \le \alpha.$$

Now, we show that we achieve mean-conditioned moment consistency. First note that

$$\frac{1}{T}|M_T^{G,i,j}| = \frac{1}{T}\left|\sum_{t \in G_T(i,j)} \overline{m}_t^k - (\hat{\mu}_i - y_t)^k\right| \le \beta.$$

Now,

$$\begin{aligned}
&\left|m^k(G_T(i,j)) - \frac{1}{|G_T(i,j)|}\sum_{t \in G_T(i,j)} (y_t - \hat{\mu}_i)^k\right| \\
=&\left|\frac{1}{|G_T(i,j)|}\sum_{t \in G_T(i,j)} \left((y_t - \hat{\mu}_i) + (\hat{\mu}_i - \mu(G_T(i,j)))\right)^k - (y_t - \hat{\mu}_i)^k\right|, \\
\le&\frac{k}{|G_T(i,j)|}\sum_{t \in G_T(i,j)} |\hat{\mu}_i - \mu(G_T(i,j))|, \\
=&\frac{k}{|G_T(i,j)|}\sum_{t \in G_T(i,j)} |\hat{\mu}_i - \overline{\mu}(G_T(i,j)) + \overline{\mu}(G_T(i,j)) - \mu(G_T(i,j))|, \\
\le&\frac{k}{|G_T(i,j)|}\sum_{t \in G_T(i,j)} |\hat{\mu}_i - \overline{\mu}(G_T(i,j))| + |\overline{\mu}(G_T(i,j)) - \mu(G_T(i,j))|, \\
\le&\frac{Tk(\alpha + \frac{1}{2n})}{|G_T(i,j)|},
\end{aligned}$$

where the first inequality follows from the fact that $|a^k - b^k| \le k|a-b|$ for any $a, b \in [0,1]$ with $a = (y_t - \hat{\mu}_i) + (\hat{\mu}_i - \mu(G_T(i,j)))$ and $b = y_t - \hat{\mu}_i$. The last inequality follows from the guarantee of mean consistency as shown above in the proof and the fact that $\overline{\mu}(G_T(i,j)) \in B_n(i)$ and $|\hat{\mu}_i - x| \le \frac{1}{2n}$ for any $x \in B_n(i)$.

[5] $(y_t - \hat{\mu}_i)^k$ roughly corresponds to what is referred to as a pseudo-moment in Jung et al. [2020].

Therefore, we can invoke the triangle inequality to conclude

$$\begin{aligned}
&\left|m^k(G_T(i,j)) - \overline{m}^k(G_T(i,j))\right| \\
\leq &\left|m^k(G_T(i,j)) - \frac{1}{|G_T(i,j)|}\sum_{t\in G_T(i,j)}(y_t-\hat{\mu}_i)^k\right| + \left|\frac{1}{|G_T(i,j)|}\sum_{t\in G_T(i,j)}(y_t-\hat{\mu}_i)^k - \overline{m}^k(G_T(i,j))\right| \\
\leq &\frac{(\beta + k\alpha + \frac{k}{2n})T}{|G_T(i,j)|}.
\end{aligned}$$

□

This lemma implies that if we can force each term $V_s^{G,i,j}, M_s^{G,i,j}$ to be small, then we will have achieved our desired goal of mean-conditioned moment multicalibration (Definition 2.4).

Observation 4.1. *Suppose a transcript π_T is such that for all $i \in [n], j \in [n']$ and $G \in \mathcal{G}$, we have that $|V_T^{G,i,j}|, |M_T^{G,i,j}| \leq \alpha T$. Then the predictions are (α, β, n, n')-mean-conditioned moment multicalibrated in the sense of Definition 2.4 for $\beta = (k+1)\alpha + \frac{k}{2n}$.*

Remark 4.1. *Note that with this parametrization, we can take α as small as we like relative to n, and by choosing an appropriately large value of n, we can take $\beta = (k+1)\alpha + \frac{k}{2n}$ as small as we like relative to n'.*

As before, we define a surrogate loss function at each round s.

Definition 4.2 (Surrogate Loss)**.** *Fixing a transcript $\pi_s \in \Pi^*$ and a parameter $\eta \in [0, \frac{1}{2}]$, define:*

$$L_s(\pi_s) = \sum_{\substack{G\in\mathcal{G},\\ i\in[n], j\in[n']}} \left(\exp(\eta V_s^{G,i,j}) + \exp(-\eta V_s^{G,i,j}) + \exp(\eta M_s^{G,i,j}) + \exp(-\eta M_s^{G,i,j})\right),$$

where V and M are functions of π_s as defined in Definition 4.1. When the transcript π_s is clear from context we will sometimes simply write L_s.

As before, our goal is to find a strategy for the learner that guarantees that our surrogate loss L_T remains small. Towards this end, we define $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k)$ to be the expected increase in the surrogate loss function in the event that the adversary plays feature vector x_{s+1} *and* the learner predicts $(\overline{\mu}, \overline{m}^k)$. Here the expectation is over the only remaining source of randomness after the conditioning — the distribution over labels y_{s+1}, which for any adversary is defined once we fix π_s and x_{s+1}.

Definition 4.3 (Conditional Change in Surrogate Loss)**.**

$$\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k) = \mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\left[\tilde{L}_{s+1} - L_s \middle| \pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k\right].$$

We again show a simple bound on $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k)$:

Lemma 4.2. *For any transcript $\pi_s \in \Pi^*$, any $x_{s+1} \in \mathcal{X}$, and any predictions $\overline{\mu}, \overline{m}^k \in [0,1]$ such that $\overline{\mu} \in B_n(i)$ and $\overline{m}^k \in B_{n'}(j)$ for some $i \in [n]$ and $j \in [n']$:*

$$\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k) \leq \eta\left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[\tilde{y}_{s+1}] - \overline{\mu}\right) C_s^{\overline{\mu},\overline{m}^k}(x_{s+1}) + \eta\left(\mathop{\mathbb{E}}_{\tilde{y}}(\tilde{y}_{s+1} - \hat{\mu}_{\overline{\mu}})^k - \overline{m}^k\right) D_s^{\overline{\mu},\overline{m}^k}(x_{s+1}) + 2\eta^2 L_s,$$

where

$$C_s^{\overline{\mu},\overline{m}^k}(x_{s+1}) = C_s^{i,j}(x_{s+1}) = \sum_{G\in\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i,j}) - \exp(-\eta V_s^{G,i,j}), \tag{5}$$

$$D_s^{\overline{\mu},\overline{m}^k}(x_{s+1}) = D_s^{i,j}(x_{s+1}) = \sum_{G\in\mathcal{G}(x_{s+1})} \exp(\eta M_s^{G,i,j}) - \exp(-\eta M_s^{G,i,j}). \tag{6}$$

For economy of notation, we will generally elide the dependence on x_{s+1} for the C and D quantities and simply write $C_s^{i,j}, D_s^{i,j}$ when the feature vector is clear from context.

Proof. To see this, observe that by definition:

$$\begin{aligned}
&\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k) \\
&= \mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\Bigg[\sum_{\mathcal{G}(x_{s+1})} \underbrace{\exp(\eta V_s^{G,i,j})\left(\exp\left(\eta\left(\tilde{y}_{s+1}-\overline{\mu}\right)\right)-1\right)+\exp(-\eta V_s^{G,i,j})\left(\exp\left(-\eta\left(\tilde{y}_{s+1}-\overline{\mu}\right)\right)-1\right)}_{*} \\
&\qquad + \underbrace{\exp(\eta M_s^{G,i,j})\exp\left(\eta\left(\left(\tilde{y}_{s+1}-\hat{\mu}_{\overline{\mu}}\right)^k-\overline{m}^k\right)-1\right)+\exp(-\eta M_s^{G,i,j})\exp\left(-\eta\left(\left(\tilde{y}_{s+1}-\hat{\mu}_{\overline{\mu}}\right)^k-\overline{m}^k\right)-1\right)}_{**}\Bigg].
\end{aligned}$$

Using the fact that for $0 < |x| < \frac{1}{2}$, $\exp(x) \le 1 + x + 2x^2$, we have that

$$\begin{aligned}
* &\le \exp(\eta V_s^{G,i,j})\left(\eta\left(y_{s+1}-\overline{\mu}\right)+2\eta^2\right)+\exp(-\eta V_s^{G,i,j})\left(-\eta\left(y_{s+1}-\overline{\mu}\right)+2\eta^2\right), \\
** &\le \exp(\eta M_s^{G,i,j})\left(\eta\left(\left(y_{s+1}-\hat{\mu}_{\overline{\mu}}\right)^k-\overline{m}^k\right)+2\eta^2\right)+\exp(-\eta M_s^{G,i,j})\left(-\eta\left(\left(\tilde{y}_{s+1}-\hat{\mu}_{\overline{\mu}}\right)^k-\overline{m}^k\right)+2\eta^2\right).
\end{aligned}$$

Now, using the linearity of expectation and distributing the outer expectation to each relevant term where $\tilde{y}_{s+1}$ appears, we get

$$\begin{aligned}
&\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k) \\
&\le \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i,j})\left(\eta\left(\mathbb{E}[\tilde{y}_{s+1}]-\overline{\mu}\right)+2\eta^2\right)+\exp(-\eta V_s^{G,i,j})\left(-\eta\left(\mathbb{E}[\tilde{y}_{s+1}]-\overline{\mu}\right)+2\eta^2\right) \\
&\qquad + \exp(\eta M_s^{G,i,j})\left(\eta\left(\mathbb{E}\left[\left(\tilde{y}_{s+1}-\hat{\mu}_{\overline{\mu}}\right)^k\right]-\overline{m}^k\right)+2\eta^2\right)+\exp(-\eta M_s^{G,i,j})\left(-\eta\left(\mathbb{E}\left[\left(\tilde{y}_{s+1}-\hat{\mu}_{\overline{\mu}}\right)^k\right]-\overline{m}^k\right)+2\eta^2\right).
\end{aligned}$$

Collecting terms appropriately and observing that

$$\sum_{\mathcal{G}(x_{s+1})}\left(\exp(\eta V_s^{G,i,j})+\exp(-\eta V_s^{G,i,j})+\exp(\eta M_s^{G,i,j})+\exp(-\eta M_s^{G,i,j})\right) \le L_s,$$

we have the desired bound. □

As before, we proceed by defining a zero-sum game between the learner and the adversary and using the minimax theorem to conclude that the learner always has a strategy that guarantees a bounded per-round increase in surrogate loss. To satisfy the convexity and compactness requirements of the minimax theorem, we will again consider a game where the learner's pure strategy space is a finite subset of $\mathcal{P}_{(\text{mean,moment})}$. To this end, we define the following grids for any $r \in \mathbb{N}$ (n and n' are the coarseness parameters of our bucketings from above):

$$\begin{aligned}
\mathcal{P}^{rn} &= \left\{0, \frac{1}{rn}, \frac{2}{rn}, \ldots, 1\right\}, \\
\mathcal{P}^{rn'} &= \left\{0, \frac{1}{rn'}, \frac{2}{rn'}, \ldots, 1\right\}.
\end{aligned}$$

As in the previous section, the need to discretize is only for technical reasons, and our algorithm has no dependence — neither in runtime nor in its convergence rate — on the value of r that we choose, so we can imagine the discretization to be arbitrarily fine.

Lemma 4.3. *For any transcript $\pi_s \in \Pi^*$ and any $x_{s+1} \in \mathcal{X}$, there exists a distribution over predictions for the learner $Q^L_{s+1} \in \Delta(\mathcal{P}^{rn} \times \mathcal{P}^{rn'})$, such that regardless of the adversary's choice of distribution of y_{s+1} over $\Delta\mathcal{Y}$, we have that:*

$$\mathop{\mathbb{E}}_{(\overline{\mu},\overline{m}^k)\sim Q^L_{s+1}}\left[\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k)\right] \le L_s\left(\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right).$$

Proof. Fix the transcript π_s and the feature vector x_{s+1}. As before, we define a zero-sum game played between the learner (the minimization player) and the adversary (the maximization player), where the objective function of the game equals the upper bound on $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k)$ from Lemma 4.2. Then, we again show that for every strategy of the adversary (i.e. distribution over y), there exists a best response for the learner that guarantees the objective function of the game is small. Finally, we appeal to the minimax theorem to conclude that there always exists a strategy for the learner that guarantees small objective value against any strategy of the adversary.

More precisely, consider the following objective function for the game:

$$\begin{aligned} u((\overline{\mu}, \overline{m}^k), y) &= \eta\,(y - \overline{\mu})\, C_s^{\overline{\mu}, \overline{m}^k} + \eta \left((y - \hat{\mu}_{\overline{\mu}})^k - \overline{m}^k \right) D_s^{\overline{\mu}, \overline{m}^k} + 2\eta^2 L_s \\ &= \eta\,(y - \overline{\mu})\, C_s^{\overline{\mu}, \overline{m}^k} + \eta \left(\left(\sum_{\ell=0}^{k} \binom{k}{\ell} (-\hat{\mu}_{\overline{\mu}})^{k-\ell} y^\ell \right) - \overline{m}^k \right) D_s^{\overline{\mu}, \overline{m}^k} + 2\eta^2 L_s \end{aligned}$$

where the pure strategy space for the learner is $X_1 = \mathcal{P}^{rn} \times \mathcal{P}^{rn'}$ and that of the adversary is (a priori) the set of all distributions over $[0, 1]$. However, we observe that the expected value of the objective for any label distribution over $[0, 1]$ is linear in $\mathbb{E}[y], \ldots, \mathbb{E}[y^k]$. So the payoff for any mixed strategy of the adversary is determined only by the associated k terms: $\mathbb{E}[y], \ldots, \mathbb{E}[y^k]$.

With this observation in mind, we perform a change of variables and define a new game with an enlarged strategy space for the adversary. In the new game, the strategy space for the learner remains $\mathcal{Q}^L = \Delta(\mathcal{P}^{rn} \times \mathcal{P}^{rn'})$. The strategy space for the adversary becomes $\mathcal{Q}^A = [0, 1]^k$, representing a choice for each of the values $\mathbb{E}[y], \ldots \mathbb{E}[y^k]$. Note that this strategy space for the adversary is unencumbered by the requirement that these chosen values actually correspond to any feasible label distribution over $[0, 1]$. The objective function of the game is obtained by replacing each term $\mathbb{E}[y^\ell]$ from our previous objective function with ψ_ℓ:

$$u((\overline{\mu}, \overline{m}^k), \psi) = \quad \eta\,(\psi_1 - \overline{\mu})\, C_s^{\overline{\mu}, \overline{m}^k} + \eta \left(\left(\hat{\mu}_{\overline{\mu}}^k + \sum_{\ell=1}^{k} \binom{k}{\ell} (-\hat{\mu}_{\overline{\mu}})^{k-\ell} \psi_\ell \right) - \overline{m}^k \right) D_s^{\overline{\mu}, \overline{m}^k} + 2\eta^2 L_s.$$

As we have noted, in the original game, the set of achievable moments $\mathbb{E}[y], \ldots, \mathbb{E}[y^k]$ is a strict subset of $[0, 1]^k$. However, enlarging the strategy space of the maximization player can only increase the (max min) value of the game, so the upper bound we are about to prove on the game value against this more powerful adversary also applies to the adversary who is implicitly choosing moments $\mathbb{E}[y], \ldots, \mathbb{E}[y^k]$ via some distribution over $[0, 1]$.

Note that u thus defined is linear in both players' strategies, and the strategy spaces for both players $\mathcal{Q}^L$ and $\mathcal{Q}^A$ are compact and convex. Hence, Sion's minimax theorem (Theorem 2.1) applies to this game. We now establish (a bound on) the value of this game. Observe that for any strategy of the adversary, the learner can pick $\overline{\mu} \in \mathcal{P}^{rn}$ as close as possible to ψ_1, and then pick $\overline{m}^k \in \mathcal{P}^{rn'}$ as close as possible to $\hat{\mu}_{\overline{\mu}}^k + \sum_{\ell=1}^{k} \binom{k}{\ell} (-\hat{\mu}_{\overline{\mu}})^{k-\ell} \psi_\ell$. Therefore, since $C_s^{\overline{\mu}, \overline{m}^k}, D_s^{\overline{\mu}, \overline{m}^k} \leq L_s$ by definition, we have that:

$$\forall \psi \in [0, 1]^k, \exists (\overline{\mu}, \overline{m}^k) \in (\mathcal{P}^{rn} \times \mathcal{P}^{rn'}) \text{ s.t. } u((\overline{\mu}, \overline{m}^k), \psi) \leq L_s \left(\frac{\eta}{rn} + \frac{\eta}{rn'} + 2\eta^2 \right).$$

We can now apply the minimax theorem (Theorem 2.1) to conclude that there exists a fixed distribution $Q_{s+1}^L \in \mathcal{Q}^L$ for the learner that guarantees objective value that is at most the above bound for every choice of the adversary, i.e.

$$\exists Q_{s+1}^L \in \mathcal{Q}^L \text{ s.t. } \forall \psi \in [0, 1]^k : \; u(Q_{s+1}^L, \psi) \leq L_s \left(\frac{\eta}{rn} + \frac{\eta}{rn'} + 2\eta^2 \right),$$

as desired. ∎

Corollary 4.1. *For every $s \in [T]$, $\pi_s \in \Pi^*$, $x_{s+1} \in \mathcal{X}$ (which fixes L_s and Q_{s+1}^L), and every adversary (which fixes a distribution over $\mathcal{Y}$):*

$$\mathop{\mathbb{E}}_{Q_{s+1}^L} [\tilde{L}_{s+1} | \pi_s] = L_s + \mathop{\mathbb{E}}_{Q_{s+1}^L} [\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}, \overline{m}^k) | \pi_s] \leq L_s \left(1 + \frac{\eta}{rn} + \frac{\eta}{rn'} + 2\eta^2 \right).$$

Lemma 4.3 defines (existentially) an algorithm that the learner can use to make predictions—Algorithm 3. We will now show that Algorithm 3 (if we could compute the distributions Q_t^L) results in mean-conditioned moment multicalibrated predictions. In Section 4.3 we show how to compute Q_t^L.

Algorithm 3: A Generic Mean Moment Multicalibrator

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

Observe x_t. Given π_{t-1} and x_t, let $Q_t^L \in \Delta(\mathcal{P}^{rn} \times \mathcal{P}^{rn'})$ be the distribution over predictions whose existence is established in Lemma 4.3.

Sample $\overline{\mu}, \overline{m}^k \sim Q_t^L$ and predict $(\overline{\mu}_t, \overline{m}_t^k) = (\overline{\mu}, \overline{m}^k)$.

We are now ready to bound our multicalibration error. The results that follow mirror the structure of Section 3.2: essentially, we apply Theorem 3.1 to the surrogate loss function of this section. As a straightforward consequence of Corollary 4.1 and the first part of Theorem 3.1, we have the following result.

Corollary 4.2. *Against any adversary, Algorithm 3 instantiated with discretization parameter r results in surrogate loss satisfying:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T] \leq 4|\mathcal{G}|n \cdot n' \cdot \exp\left(\frac{T\eta}{rn} + \frac{T\eta}{rn'} + 2T\eta^2\right).$$

Proof. Note that the first part of Theorem 3.1 applies in this case to the process L, with $L_0 = 4|G|n \cdot n'$ and $c = \frac{1}{rn} + \frac{1}{rn'}$. The bound follows by plugging these values into (3). □

Next, we can convert this into a bound on Algorithm 1's expected calibration error, using Theorem 3.1. The proof mirrors the argument in Section 3 and can be found in the Appendix.

Theorem 4.1. *When Algorithm 3 is run using bucketing coarseness parameters n and n', discretization parameter $r \in \mathbb{N}$, and $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n \cdot n')}{2T}} \in (0, 1/2)$, then against any adversary, its sequence of mean-moment predictions is (α, β, n, n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$, where $\beta = (k+1)\alpha + \frac{k}{2n}$ and:*

$$\mathbb{E}[\alpha] \leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n \cdot n')}{T}}.$$

For $r = \frac{\sqrt{T}(n+n')}{\varepsilon n \cdot n' \cdot \sqrt{2\ln(4|\mathcal{G}|n \cdot n')}}$, this gives:

$$\mathbb{E}[\alpha] \leq (2+\varepsilon)\sqrt{\frac{2}{T}\ln(4|\mathcal{G}|n \cdot n')}.$$

Here the expectation is taken over the randomness of the transcript π_T.

We can similarly use the second part of Theorem 3.1 to prove a high probability bound on the multicalibration error of Algorithm 3. The proof is in the Appendix.

Theorem 4.2. *When Algorithm 3 is run using bucketing coarseness parameters n and n', discretization $r \in \mathbb{N}$ and $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n \cdot n')}{2T}} \in (0, 1/2)$, then against any adversary, with probability $1-\lambda$ over the randomness of the transcript, its sequence of predictions is (α, β, n, n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$ for $\beta = (k+1)\alpha + \frac{k}{2n}$ and:*

$$\alpha \leq \frac{1}{rn} + \frac{1}{rn'} + 4\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n \cdot n'}{\lambda}\right)}.$$

For $r = \frac{\sqrt{T}(n+n')}{\epsilon n \cdot n' \sqrt{2\ln(4|\mathcal{G}|n \cdot n'/\lambda)}}$, this gives:

$$\alpha \leq (4+\epsilon)\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n \cdot n'}{\lambda}\right)}.$$

4.3 Deriving an Efficient Algorithm via Equilibrium Computation

Previously, we derived Algorithm 3 and proved that it results in mean-conditioned moment multicalibrated predictions. But Algorithm 3 is not explicitly defined, as it relies on the distributions Q_t^L whose existence we showed in Lemma 4.3 but which we did not explicitly construct. In this section, we show how to efficiently solve for this distribution Q_t^L using a linear program with $4n \cdot n'$ variables and $2^k + 1$ constraints. If k is a small constant (e.g. $k = 2$ for variance multicalibration), then this linear program can be explicitly described and solved. But even when k is too large to enumerate all 2^k constraints, we show that there is a separation oracle that runs in time $O(k)$, allowing us to efficiently solve this linear program (i.e. in time polynomial in $n, n', T, |\mathcal{G}|$, and k) using the Ellipsoid algorithm.

Recall that in our simplified game, the learner has pure strategies $(\overline{\mu}, \overline{m}^k) \in \mathcal{P}^{rn} \times \mathcal{P}^{rn'}$, and the adversary has strategy space $\mathcal{Q}^A = [0,1]^k$. Since the objective function is linear in the adversary's action ψ, we can view this as the set of mixed strategies over the 2^k pure strategies $\psi \in \{0,1\}^k$. We recall the objective function:

$$u((\overline{\mu}, \overline{m}^k), \psi) = \quad \eta\,(\psi_1 - \overline{\mu})\, C_s^{\overline{\mu},\overline{m}^k} + \eta \left(\left(\hat{\mu}_{\overline{\mu}}^k + \sum_{\ell=1}^{k} \binom{k}{\ell} (-\hat{\mu}_{\overline{\mu}})^{k-\ell} \psi_\ell \right) - \overline{m}^k \right) D_s^{\overline{\mu},\overline{m}^k} + 2\eta^2 L_s.$$

Since the equilibrium structure stays the same under positive affine transformations of the objective function, for the purposes of computing equilibria, we may redefine the objective function to be:

$$u((\overline{\mu}, \overline{m}^k), \psi) = \quad (\psi_1 - \overline{\mu})\, C_s^{\overline{\mu},\overline{m}^k} + \left(\left(\hat{\mu}_{\overline{\mu}}^k + \sum_{\ell=1}^{k} \binom{k}{\ell} (-\hat{\mu}_{\overline{\mu}})^{k-\ell} \psi_\ell \right) - \overline{m}^k \right) D_s^{\overline{\mu},\overline{m}^k}. \tag{7}$$

The specific values of $C_s^{\overline{\mu},\overline{m}^k}$, $\hat{\mu}_{\overline{\mu}}$ and $D_s^{\overline{\mu},\overline{m}^k}$ do not matter for the analysis that follows—but what is relevant is that by definition, they are constant for any two $(\overline{\mu}, \overline{m}^k)$ and $(\overline{\mu}', \overline{m}^{k\prime})$ both in the same bucket — in other words, if $\exists i \in [n], j \in [n']$ such that $(\overline{\mu}, \overline{m}^k), (\overline{\mu}', \overline{m}^{k\prime}) \in B_{n,n'}(i,j)$. We wish to find a minimax strategy for the learner in this game, i.e. to find a solution to

$$\operatorname*{argmin}_{Q^L \in \mathcal{Q}^L} \max_{Q^A \in \mathcal{Q}^A} u(Q^L, Q^A).$$

A priori, the learner has $r^2 n' n$ pure strategies (i.e. $|\mathcal{P}^{rn} \times \mathcal{P}^{rn'}| = r^2 n' n$), and a minimax strategy could potentially be supported over all of them (causing our algorithm to have running time depending on r). However, we prove that we can without loss of generality reduce the size of the learner's pure strategy space to $4n'n$ (Lemma 4.4), which will eliminate any running time dependence on r and allow us to choose as fine a discretization as we like. We also show in Appendix C that the learner always has a minimax strategy that randomizes over a support of at most $k+1$ actions. Thus, as with mean multicalibration, we need only make limited use of randomness (at least for k small).

We first reduce the space of "relevant" pure strategies for the learner — intuitively, points that are at—or just barely below—the boundary of a bucket:

$$\hat{\mathcal{P}}^{r,n} = \bigcup_{i \in [n-1]} \left\{ \frac{i-1}{n}, \frac{i}{n} - \frac{1}{rn} \right\} \bigcup \left\{ \frac{n-1}{n}, 1 \right\} \subset \mathcal{P}^{rn},$$

$$\hat{\mathcal{P}}^{r,n'} = \bigcup_{i \in [n'-1]} \left\{ \frac{i-1}{n'}, \frac{i}{n'} - \frac{1}{rn'} \right\} \bigcup \left\{ \frac{n'-1}{n'}, 1 \right\} \subset \mathcal{P}^{rn'}.$$

Given these sets, define $\hat{\mathcal{Q}}_{r,n,n'}^L \equiv \Delta\left(\hat{\mathcal{P}}^{r,n} \times \hat{\mathcal{P}}^{r,n'}\right) \subset \mathcal{Q}^L$.

Lemma 4.4. *In the game with objective function u as defined in (7), the value of the game is unaffected if the learner is restricted to mixed strategies in $\hat{\mathcal{Q}}_{r,n,n'}^L$, a set of distributions which in particular have support over at most $4nn'$ actions. In other words:*

$$\min_{Q^L \in \mathcal{Q}^L} \max_{Q^A \in \mathcal{Q}^A} u(Q^L, Q^A) = \min_{\hat{Q}^L \in \hat{\mathcal{Q}}_{r,n,n'}^L} \max_{Q^A \in \mathcal{Q}^A} u(\hat{Q}^L, Q^A).$$

Proof. Fix any strategy $Q^L \in \mathcal{Q}^L$. Since $\hat{\mathcal{Q}}^L_{r,n,n'} \subseteq \mathcal{Q}^L$, it is sufficient to show that there exists a strategy $\hat{Q}^L \in \hat{\mathcal{Q}}^L_{r,n,n'}$ such that:

$$\max_{Q^A \in \mathcal{Q}^A} u(Q^L, Q^A) \geq \max_{Q^A \in \mathcal{Q}^A} u(\hat{Q}^L, Q^A).$$

To see this, first observe that we can regroup terms in the objective function (7) and write it as:

$$u((\overline{\mu}, \overline{m}^k), \psi) = -\overline{\mu} C_s^{\overline{\mu},\overline{m}^k} + \hat{\mu}_{\overline{\mu}}^k D_s^{\overline{\mu},\overline{m}^k} - \overline{m}^k D_s^{\overline{\mu},\overline{m}^k} + \sum_{\ell=1}^{k} \psi_\ell F_\ell^{\overline{\mu},\overline{m}^k} \tag{8}$$

$$\text{where } F_1^{\overline{\mu},\overline{m}^k} = C_s^{\overline{\mu},\overline{m}^k} - k\hat{\mu}_{\overline{\mu}}^{k-1} C_s^{\overline{\mu},\overline{m}^k}, \tag{9}$$

$$\forall \ell > 1, \ell \in [n]: \; F_\ell^{\overline{\mu},\overline{m}^k} = \binom{k}{\ell} (-\hat{\mu}_{\overline{\mu}})^{k-\ell} D_s^{\overline{\mu},\overline{m}^k}. \tag{10}$$

Further, by definition for any $\overline{\mu}, \overline{\mu}' \in B_n(i)$ for some $i \in [n]$ and $\overline{m}^k, \overline{m}^{k\prime} \in B_{n'}(j)$, we have, for $X = C, D$,

$$X_s^{\overline{\mu},\overline{m}^k} = X_s^{\overline{\mu}',\overline{m}^{k\prime}} = X_s^{i,j},$$
$$\hat{\mu}_{\overline{\mu}} = \hat{\mu}_{\overline{\mu}'},$$

and therefore this equality holds for $X = F$ as well. Against a given strategy Q^L for the learner, the adversary' payoff from pure strategy ψ is:

$$u(Q^L, \psi) = \sum_{(\overline{\mu},\overline{m}^k)} Q^L(\overline{\mu}, \overline{m}^k) \left(-\overline{\mu} C_s^{\overline{\mu},\overline{m}^k} + \hat{\mu}_{\overline{\mu}}^k D_s^{\overline{\mu},\overline{m}^k} - \overline{m}^k D_s^{\overline{\mu},\overline{m}^k} + \sum_{\ell=1}^{k} \psi_\ell F_\ell^{\overline{\mu},\overline{m}^k} \right),$$

which, given the previous fact about F, can be rewritten as

$$u(Q^L, \psi) = \underbrace{\sum_{(\overline{\mu},\overline{m}^k)} Q^L(\overline{\mu}, \overline{m}^k) \left(-\overline{\mu} C_s^{\overline{\mu},\overline{m}^k} + \hat{\mu}_{\overline{\mu}}^k D_s^{\overline{\mu},\overline{m}^k} - \overline{m}^k D_s^{\overline{\mu},\overline{m}^k} \right)}_{(*)}$$
$$+ \underbrace{\sum_{\ell=1}^{k} \psi_\ell \sum_{\substack{i \in [n], \\ j \in [n']}} F_\ell^{i,j} \left(\sum_{(\overline{\mu},\overline{m}^k) \in B(i,j)} Q^L(\overline{\mu}, \overline{m}^k) \right)}_{(**)}.$$

Observe that term $(*)$ is independent of ψ. Therefore, fixing a Q^L, it is equivalent for the adversary to maximize $(**)$. By observation, for any mixed strategy of the learner Q^L, the adversary's incentives are only affected through the induced distribution over buckets.

So, given Q^L, the best response of the adversary is preserved for any other strategy $\hat{Q}^L$ that maintains the same mass on each bucket, i.e. for all $i \in [n]$ and $j \in [n']$, $\sum_{(\overline{\mu},\overline{m}^k) \in B(i,j)} \left(Q^L(\overline{\mu}, \overline{m}^k) - \hat{Q}^L(\overline{\mu}, \overline{m}^k) \right) = 0$. Consider the learner's problem of minimizing the objective value among strategies of this form, i.e. preserving the mass on each bucket. This reduces to solving, for each $i \in [n], j \in [n']$, the optimization problem

$$\min_{\hat{Q}^L \geq 0} \sum_{(\overline{\mu},\overline{m}^k) \in B(i,j)} \hat{Q}^L(\overline{\mu}, \overline{m}^k) \left(-\overline{\mu} C_s^{i,j} + \hat{\mu}_i^k D_s^{i,j} - \overline{m}^k D_s^{i,j} \right)$$
$$\text{s.t.} \sum_{(\overline{\mu},\overline{m}^k) \in B(i,j)} \left(Q^L(\overline{\mu}, \overline{m}^k) - \hat{Q}^L(\overline{\mu}, \overline{m}^k) \right) = 0.$$

Within a bucket, the coefficients $\left(-\overline{\mu}C_s^{i,j} + \hat{\mu}_i^k D_s^{i,j} - \overline{m}^k D_s^{i,j}\right)$ are linear in $\overline{\mu}, \overline{m}^k$ and therefore there must exist a solution that puts all mass $\sum_{(\overline{\mu},\overline{m}^k)\in B(i,j)} Q^L(\overline{\mu},\overline{m}^k)$ on an extreme point of the bucket. For example, if $i \in [n-1]$, $j \in [n'-1]$; all mass can be placed without loss of generality on one of the four points in $\left\{\frac{i-1}{n}, \frac{i}{n} - \frac{1}{rn}\right\} \times \left\{\frac{j-1}{n'}, \frac{j}{n} - \frac{1}{rn'}\right\}$. If $i = n$, the corresponding set is $\{\frac{n-1}{n}, 1\}$, and if $j = n'$, the corresponding set is $\{\frac{n'-1}{n'}, 1\}$. Moving all the mass in each bucket to the optimal corner point, we have that for any strategy Q^L of the learner, there exists $\hat{Q}^L \in \hat{\mathcal{Q}}^L_{r,n,n'}$ such that $\max_{Q^A\in\mathcal{Q}^A} u(Q^L, Q^A) \geq \max_{Q^A\in\mathcal{Q}^A} u(\hat{Q}^L, Q^A)$, as desired. This concludes the proof. □

The result is that to compute the equilibrium strategy for the learner, it suffices to solve:

$$\operatorname*{argmin}_{Q^L\in\hat{\mathcal{Q}}^L_{r,n,n'}} \max_{\psi\in\{0,1\}^k} u(Q^L, \psi).$$

We can directly express this as a linear program with $4nn'$ variables and $2^k + 1$ constraints — see Linear Program 1.

$$\begin{aligned}
&\min_{Q^L\in\hat{\mathcal{Q}}^L_{r,n,n'}} \gamma \text{ s.t.} \\
\forall \psi \in \{0,1\}^k :& u(Q^L, \psi) \leq \gamma, \\
&\sum\nolimits_{(\overline{\mu},\overline{m}^k)\in\hat{\mathcal{P}}^{r,n}\times\hat{\mathcal{P}}^{r,n'}} Q^L((\overline{\mu},\overline{m}^k)) = 1, \\
\forall\, (\overline{\mu},\overline{m}^k) \in \hat{\mathcal{P}}^{r,n}\times\hat{\mathcal{P}}^{r,n'} :& Q^L((\overline{\mu},\overline{m}^k)) \geq 0.
\end{aligned}$$

Figure 1: A Linear Program for Computing a Minimax Equilibrium Strategy for the Learner at Round t.

This is a linear program in $4nn' + 1$ variables, with $2^k + 1$ constraints. If k is a constant, this is a polynomially sized linear program that can be solved explicitly. If k is superconstant, we will see that we can still solve the linear program with the Ellipsoid algorithm, because we can efficiently find violated constraints.

Algorithm 4: Von Neumann's Mean Moment Multicalibrator

INPUT: $\epsilon > 0$.

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

Observe x_t and compute $C^{\overline{\mu},\overline{m}^k}_{t-1}(x_t), D^{\overline{\mu},\overline{m}^k}_{t-1}(x_t), (F^{\overline{\mu},\overline{m}^k}_{\ell,t-1}(x_t))_{\ell=1}^n$ for each $(\overline{\mu},\overline{m}^k) \in \hat{\mathcal{P}}^{r,n}\times\hat{\mathcal{P}}^{r,n'}$ as in Equations (5, 6, 9, 10).

Find an ϵ-approximate solution to the linear program from Figure 1, to obtain solution $Q^L_t \in \hat{\mathcal{Q}}^L_{r,n,n'}$.

Predict $(\overline{\mu}_t, \overline{m}^k_t) = (\overline{\mu},\overline{m}^k)$ with probability $Q^L_t((\overline{\mu},\overline{m}^k))$.

We thus obtain the following theorem:

Theorem 4.3. *Algorithm 4 implements Algorithm 3. In particular, it obtains multivalidity guarantees arbitrarily close to those of Theorems 4.1 and 4.2. Namely, for any desired $\epsilon > 0$, we have the following.*

Choosing $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n\cdot n'+\epsilon)}{2T}} \in (0, 1/2)$, against any adversary, over the randomness of the transcript, the sequence of mean-moment predictions produced by Algorithm 4 is (α, β, n, n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$ where $\beta = (k+1)\alpha + \frac{k}{2n}$ and:

$$\mathbb{E}[\alpha] \leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n\cdot n' + \epsilon)}{T}}.$$

For $r = \frac{\sqrt{T}(n+n')}{\epsilon' n\cdot n'\cdot\sqrt{2\ln(4|\mathcal{G}|n\cdot n'+\epsilon)}}$, this gives:

$$\mathbb{E}[\alpha] \leq (2+\epsilon')\sqrt{\frac{2}{T}\ln(4|\mathcal{G}|n\cdot n' + \epsilon)}.$$

Moreover, choosing $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n\cdot n')+\epsilon T}{2T}} \in (0, 1/2)$*, with probability* $1-\lambda$ *over the randomness of the transcript* π_T *we have*

$$\alpha \leq \frac{1}{rn} + \frac{1}{rn'} + 4\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right) + 2\epsilon}.$$

For $r = \frac{(n+n')}{\epsilon' n\cdot n'\sqrt{\frac{2}{T}\ln(4|\mathcal{G}|n\cdot n'/\lambda)+2\epsilon}}$*, this gives:*

$$\alpha \leq (4+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right) + 2\epsilon}.$$

The runtime of Algorithm 6 scales as $O(|\mathcal{G}|)$ *with the total number of groups* $|\mathcal{G}|$*, and is polynomial in* n, n', T, k*, and* $\log(\frac{1}{\epsilon})$ *(and is independent of* r*).*

Remark 4.2. *As before, if* $|\mathcal{G}(x_t)|$ *is efficiently enumerable, then the running time dependence on* $|\mathcal{G}|$ *can be replaced with a dependence on* $|\mathcal{G}(x_t)|$*.*

Proof. First consider the running time of the algorithm. The quantities $C_{t-1}^{\overline{\mu},\overline{m}^k}(x_t), D_{t-1}^{\overline{\mu},\overline{m}^k}(x_t), F_{\ell,t-1}^{\overline{\mu},\overline{m}^k}(x_t)$ are simple sums, which can be computed in time linear in $|\mathcal{G}|$ (or $|\mathcal{G}(x_t)|$ if it is efficiently enumerable) and T. The linear program has $4nn'+1$ variables, and 2^k+1 constraints. If k is a constant, this is polynomially sized. Now consider the case in which k is large. In this case we will solve the linear program by applying the Ellipsoid algorithm to its "rational" modification (see below). The runtime of this approach is polynomial under several well-known conditions, which are given in the following theorem:

Theorem 4.4 (Schrijver [1986], Corollary 14.1a)**.** *For an optimization program of a linear objective with rational coefficients over a rational polyhedron* P *in* $\mathbb{R}^q$ *for which we are given a separation oracle, the Ellipsoid algorithm solves it exactly in time polynomial in the following parameters: the number of variables* q*, the largest bit complexity* ϕ *of any linear inequality defining* P*, the bit complexity* c *of the objective function, and the runtime of a separation oracle.*

Linear Program 1 has finitely many constraints so its feasible region is a polyhedron. However, exponential terms in the coefficients of the constraints associated with the adversarial best-responses (which are due to our definition of the soft-max surrogate loss) prevent it from being *rational*. To fix this, we only keep $O(\log\frac{1}{\epsilon})$ bits of precision after the integer part of every coefficient of LP 1, resulting in a new LP whose coefficients are all rational and within $\pm\frac{\epsilon}{2}$ from their original values in LP 1. The new LP indeed has a rational polyhedron as its feasible region. We now pause to see that solving the rational LP achieves value within ϵ of the desired optimum of LP 1. This is shown more generally in the following technical lemma, which we will reuse in Section 5.3; its proof is deferred to the Appendix.

Lemma 4.5. *Consider a linear program of the following form, with variables* $x \in \mathbb{R}^m$*,* $\gamma \in \mathbb{R}$ *for some* m*:*

$$\textit{Minimize } \gamma, \quad \textit{subject to:} \quad Ax \leq \gamma \mathbf{1}^m,\ x\cdot \mathbf{1}^m = 1,\ x \geq 0.$$

Here, $\mathbf{1}^m \in \mathbb{R}^m$ *is the all-ones vector, and* $A = (a_{ji})$ *is a finite matrix with real entries.*

Take any $\epsilon > 0$*. Modify the above linear program by replacing matrix* A *with matrix* $\tilde{A} = (\tilde{a}_{ji})$*, where each* $\tilde{a}_{ji}$ *is a rational number within* $\pm\frac{\epsilon}{2}$ *from* a_{ji}*, obtained by truncating* a_{ji} *to* $O(\log\frac{1}{\epsilon})$ *bits of precision. Then, any optimal solution* $(x^{*,r}, \gamma^{*,r})$ *of the resulting rational linear program is an* ϵ*-approximately optimal feasible solution of the original linear program.*

Linear Program 1 is of the type given in Lemma 4.5, so we have that solving the rational LP gives the desired ϵ-approximation to the optimum of Linear Program 1. Now we verify that all linear constraints

of the rational version of LP 1 have polynomial bit complexity. Recall that the left side of any constraint bounding the objective function can be written as:

$$u(Q^L,\psi)=\underbrace{\sum_{(\overline{\mu},\overline{m}^k)} Q^L(\overline{\mu},\overline{m}^k)\left(-\overline{\mu}C_{t-1}^{\overline{\mu},\overline{m}^k}+\hat{\mu}_{\overline{\mu}}^k D_{t-1}^{\overline{\mu},\overline{m}^k}-\overline{m}^k D_{t-1}^{\overline{\mu},\overline{m}^k}\right)}_{(*)}+\underbrace{\sum_{\ell=1}^{k}\psi_\ell\sum_{\substack{i\in[n],\\ j\in[n']}} F_\ell^{i,j}\left(\sum_{(\overline{\mu},\overline{m}^k)\in B(i,j)} Q^L(\overline{\mu},\overline{m}^k)\right)}_{(**)}.$$

There are $4nn'+1$ variables. We can bound the coefficient in which any $Q^L(\overline{\mu},\overline{m}^k)$ appears in (*) by:

$$\max_{\overline{\mu},\overline{m}^k}\sum_G \exp(\eta V_{t-1}^{G,i,j})-\exp(-\eta V_{t-1}^{G,i,j})+2\left(\exp(\eta M_{t-1}^{G,i,j})-\exp(-\eta M_{t-1}^{G,i,j})\right)\le|\mathcal{G}|(6\exp(\eta 2T))\le 6|\mathcal{G}|\exp(2T).$$

The coefficient of any variable $Q^L(\overline{\mu},\overline{m}^k)$ in (**) is at most:

$$\sum_{\ell=1}^{k}\psi_\ell\sum_{\substack{i\in[n],\\ j\in[n']}} F_\ell^{i,j}\le k\cdot(nn')\cdot\max_{i,j}\left\{2^k\left(\sum_G 2\exp(\eta M_T^{G,i,j})\right)\right\}\le 2^{k+1}k|\mathcal{G}|nn'\cdot\exp(2T).$$

Recalling that we are also keeping $O(\log\frac{1}{\epsilon})$ bits of precision for each coefficient, it follows that the maximum bit complexity of any constraint is bounded by

$$O\left(2\cdot 4nn'\cdot\left(\log\left(2^{k+1}k|\mathcal{G}|nn'\cdot\exp(2T)\right)+\log\frac{1}{\epsilon}\right)\right)=\text{poly}\left(n,n',|\mathcal{G}|,T,k,\log\frac{1}{\epsilon}\right).$$

Of course, the objective value, which is simply γ, also has polynomial bit complexity.

Next, we describe an efficient separation oracle for the LP. Consider a candidate solution (Q^L,γ). The constraint requiring that Q^L be a probability distribution can be checked explicitly. Thus, it remains to either find a violated constraint corresponding to some pure strategy $\psi\in\{0,1\}^k$ of the adversary, or to assert that none exists. But this reduces to the problem of finding the most violated such constraint, which corresponds to the adversary's pure best response problem. Note that only the (**) term of the objective function (see the formula above) depends on the adversary's action. Thus, the best response problem of the adversary corresponds to finding

$$\psi^*=\arg\max_{\psi\in\{0,1\}^k}\sum_{\ell=1}^{k}\psi_\ell\sum_{i\in[n],j\in[n']} F_\ell^{i,j}\sum_{(\overline{\mu},\overline{m}^k)\in B(i,j)} Q^L(\overline{\mu},\overline{m}^k).$$

The best response for the adversary given a fixed distribution Q^L can be computed by setting each coordinate $\ell\in[k]$ independently to be either 0 or 1: namely, $\psi_\ell=1$ if $\sum_{\substack{i\in[n],\\ j\in[n']}} F_\ell^{i,j}\left(\sum_{(\overline{\mu},\overline{m}^k)\in B(i,j)} Q^L(\overline{\mu},\overline{m}^k)\right)\ge 0$ and $\psi_\ell=0$ otherwise. This takes $O(k)$ iterations, at each of which the expression whose sign determines ψ_ℓ is computed in polynomial time. Once the adversary's best response has been computed, the oracle simply outputs the corresponding constraint if it is violated, and otherwise it asserts that the proposed solutions is feasible. Thus, we have a polynomial-time separation oracle for Linear Program 1.

This completes the proof that Linear Program 1 can be solved, at each round, to precision $\epsilon>0$ in time polynomial in $n,n',\log|\mathcal{G}|,T,k,\log\frac{1}{\epsilon}$. The runtime of Algorithm 4 is therefore also $\text{poly}(n,n',|\mathcal{G}|,T,k,\log\frac{1}{\epsilon})$, where the dependence on $|\mathcal{G}|$ is $O(|\mathcal{G}|)$ — since at the beginning of each round t, we precompute the coefficients of the linear program in time linear in $|\mathcal{G}|$, and the Ellipsoid runs in time polynomial in $\log|\mathcal{G}|$.

Finally, we need to demonstrate that the claimed multivalidity guarantees (which are a function of the chosen $\epsilon>0$) indeed hold. If we were exactly solving the linear program, this would be immediate from Lemma 4.4 and the fact that Linear Program 1 is directly solving for:

$$\operatorname*{argmin}_{Q^L\in\hat{\mathcal{Q}}^L_{r,n,n'}}\max_{\psi\in\{0,1\}^k} u(Q^L,\psi).$$

We only need to verify that our approximate guarantees follow from approximately solving the linear program.

Lemma 4.6. *Algorithm 4 achieves the multivalidity guarantees specified in Theorem 4.3.*

The proof of this lemma involves repeating several calculations from Section 4.2 with an ϵ error term, and so is deferred to the Appendix. □

5 Online Multivalid Marginal Coverage

5.1 An Outline of Our Approach

In this section, we derive an online algorithm for supplying prediction intervals with a coverage target $1-\delta$ that are multivalid with respect to some collection of groups $\mathcal{G}$. When $\mathcal{G} = \{\mathcal{X}\}$, this corresponds to giving simple marginal prediction intervals — a similar problem as solved by conformal prediction[6], but without requiring distributional assumptions. For richer classes $\mathcal{G}$, we obtain correspondingly stronger guarantees. We follow the same basic strategy that we developed in Section 3 for making multicalibrated mean predictions, with a couple of important deviations.

1. First, we observe that even in the distributional setting, it is *not* always possible to provide prediction intervals that have coverage probability exactly $1-\delta$. Consider, for example, the case in which the label distribution is a point mass. Then, any prediction interval will have coverage probability either 0 or 1 — in both cases, bounded away from the target $1-\delta$. More generally, if we are giving prediction intervals with endpoints in some discrete set $\{0, 1/rn, \ldots, 1\}$, in order for there to exist prediction intervals with approximately the desired coverage probability in the distributional setting, the distribution must not be overly concentrated on any sub-interval of width $1/rn$. We define a sufficient smoothness condition (Definition 5.2) for appropriately tight prediction intervals to be guaranteed to exist in the distributional setting — a condition that becomes increasingly mild as we take our discretization parameter r to be larger. We then derive — existentially, using the minimax theorem — the existence of an online algorithm that gives prediction intervals that are multivalid at the desired coverage probability when played against an adversary who is constrained at every round to play smooth label distributions. We observe (Remark 5.2) that our smoothness condition is very mild, in the sense that we can *enforce it ourselves* by adding noise $U[-\epsilon, \epsilon]$ to the adversary's labels, rather than making assumptions about the adversary. When we do this, the intervals we obtain continue to have valid coverage if we widen both endpoints by ϵ.

2. To instantiate our algorithm, we again need to compute equilibrium strategies for an appropriately defined game for our learner to sample from. Unlike in the cases of mean and moment multicalibration, however, the equilibrium strategies in this case do not appear to have any nice structure. We can still derive an efficient algorithm, however, by solving a linear program at each round to compute an equilibrium of the corresponding game. Because we assume that our adversary plays label distributions that are appropriately smooth, the adversary has exponentially many pure strategies in this game, and so we cannot efficiently enumerate all of the constraints in our equilibrium computation program. Instead, we show that a simple greedy algorithm is able to implement a separation oracle, which allows us to solve the linear program efficiently using the Ellipsoid algorithm.

5.2 An Existential Derivation of the Algorithm and Multicoverage Bounds

Our goal in this section is to derive an algorithm which at each round, makes predictions $(\overline{\ell}_t, \overline{u}_t) \in \mathcal{P}_{\text{interval}}$ that are multivalid with respect to some target coverage probability $1-\delta$.

[6] In fact, even with $\mathcal{G} = \{\mathcal{X}\}$ the guarantees are stronger than the marginal guarantees promised by conformal prediction techniques, because they remain valid even conditioning on the prediction. This is important and rules out trivial solutions, like predicting the full interval with probability $1-\delta$ and an empty interval with probability δ.

Towards this end, we define the coverage error of a group G and interval (ℓ, u):

Definition 5.1. *Given a transcript $\pi_s = (x_t, (\overline{\ell}_t, \overline{u}_t), y_t)_{t=1}^s$, we define the coverage error for a group $G \in \mathcal{G}$ and bucket $(i,j) \in [n] \times [n]$ at time s to be:*

$$V_s^{G,(i,j)} = \sum_{t=1}^{s} \mathbb{1}[x_t \in G, (\overline{\ell}_t, \overline{u}_t) \in B_n(i,j)] \cdot v_\delta((\overline{\ell}_t, \overline{u}_t), y_t),$$

where $v_\delta((\ell, u), y) = \text{Cover}((\ell, u), y) - (1 - \delta)$.

Just as before, our coverage error serves as a bound on our multicoverage error.

Observation 5.1. *Fix a transcript π_T. If for all $G \in \mathcal{G}$, and buckets $(i,j) \in [n] \times [n]$, we have that:*

$$\left| V_T^{G,(i,j)} \right| \leq \alpha T$$

then the corresponding sequence of prediction intervals are (α, n)-multivalid with respect to $\mathcal{G}$.

We now pause to observe that even in the easier distributional setting where data are drawn from a fixed distribution: $(x, y) \sim \mathcal{D}$ — there may not be any interval $(\ell, u) \in \mathcal{P}_{\text{interval}}$ that satisfies the desired target coverage value, i.e. that guarantees that $|\mathbb{E}_{(x,y)\sim\mathcal{D}}[v_\delta((\ell, u), y]|$ is small. Consider for example a label distribution that places all its mass on a single value $y = i \in [0, 1]$. Then any interval (ℓ, u) covers the label with probability 1 or probability 0, which for $\delta \notin \{0, 1\}$ is bounded away from our target coverage probability. 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, we define a class of smooth distributions for which achieving (approximately) the target coverage is always possible for some interval (ℓ, u) defined over an appropriately finely discretized range:

$$\mathcal{P}^{rn}_{\text{interval}} = \{(i,j) \in \mathcal{P}_{\text{interval}} : i, j \in \mathcal{P}^{rn}\},$$

where as before, $\mathcal{P}^{rn}$ is the uniform grid on $[0, 1]$, $\{0, \frac{1}{rn}, \ldots, 1\}$. We show that we can similarly achieve (approximately) our target coverage goals in the online adversarial setting when the adversary is constrained to playing smooth distributions.

Definition 5.2. *A label distribution $Q \in \Delta\mathcal{Y}$ is* (ρ, rn)-smooth *if for any $0 \leq a \leq b \leq 1$ such that $|a-b| \leq \frac{1}{rn}$,*

$$\Pr_{y\sim Q}[y \in [a, b]] \leq \rho.$$

We say that a joint distribution $\mathcal{D} \in \Delta(\mathcal{X} \times \mathcal{Y})$ is (ρ, rn)-smooth *if for every $x \in \mathcal{X}$, the marginal label distribution conditional on x, $\mathcal{D}|_x$, is (ρ, rn)-smooth.*

Observation 5.2. *For any $\delta \in [0, 1]$ and any fixed (ρ, rn)-smooth label distribution Q, there always exists some interval $(\overline{\ell}, \overline{u}) \in \mathcal{P}^{rn}_{interval}$ such that $|\Pr_{y\sim Q}[\text{Cover}((\ell, u), y)] - (1 - \delta)| \leq \rho$.*

Remark 5.1. *The assumption of (ρ, rn)-smoothness becomes more mild for any ρ as $r \to \infty$. Just as for mean and moment multicalibration, in which our error bounds inevitably depend on the level of discretization r that we choose, here our error bounds will depend on the smoothness level ρ of the adversary's distributions at the discretization level r that we choose. Finally, observe that smoothness is an extremely mild condition in that we can enforce it ourselves if we so choose, rather than assuming that the adversary is constrained. We elaborate on this in Remark 5.2.*

Definition 5.3. *We write $\mathcal{Q}_{\rho,rn}$ for the set of all (ρ, rn) smooth distributions over $[0, 1]$. We write $\hat{\mathcal{Q}}_{\rho,rn}$ for the set of all (ρ, rn)-smooth distributions whose support belongs to the grid $\mathcal{P}^{rn} = \{0, \frac{1}{rn}, \ldots, 1\}$:*

$$\hat{\mathcal{Q}}_{\rho,rn} \equiv \Delta\mathcal{P}^{rn} \cap \mathcal{Q}_{\rho,rn}.$$

We will show (in Lemma 5.3) that when the learner is restricted to selecting intervals from $\mathcal{P}^{rn}_{\text{interval}}$, without loss of generality, rather than considering adversaries that play arbitrary distributions over $\mathcal{Q}_{\rho,rn}$, it suffices to consider adversaries that play discrete distributions from $\hat{\mathcal{Q}}_{\rho,rn}$, which will be more convenient for us.

To bound the maximum absolute value of our coverage errors across all groups and interval predictions, we again introduce the same style of surrogate loss function:

Definition 5.4 (Surrogate loss)**.** *Fixing a transcript $\pi_s \in \Pi^*$ and a parameter $\eta \in (0, 1/2)$, define a surrogate coverage loss function at day s as:*

$$L_s(\pi_s) = \sum_{\substack{G \in \mathcal{G}, \\ (i,j) \in [n] \times [n]}} \left(\exp(\eta V_s^{G,(i,j)}) + \exp(-\eta V_s^{G,(i,j)}) \right),$$

where $V_s^{G,(i,j)}$ are implicitly functions of π_s. When the transcript is clear from context we will sometimes simply write L_s.

Once again, $0 < \eta < \frac{1}{2}$ is a parameter that we will set later.

As before, we proceed by bounding the conditional change in the surrogate loss function:

Definition 5.5 (Conditional Change in Surrogate Loss)**.** *Fixing $\pi_s \in \Pi^*$, $x_{s+1} \in \mathcal{X}$ and an interval $(\ell, u) \in \mathcal{P}^{rn}_{interval}$, define the conditional change in surrogate loss to be:*

$$\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{t+1}, \overline{u}_{t+1})) = \mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[\tilde{L}_{s+1} - L_s | x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1}), \pi_s].$$

Lemma 5.1. *For every transcript $\pi_s \in \Pi^*$, every $x_{s+1} \in \mathcal{X}$, and every $(\overline{\ell}_{s+1}, \overline{u}_{s+1}) \in B_n(i,j)$ we have that:*

$$\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1})) \leq \left(\eta(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1})]) \right) C_s^{i,j}(x_{s+1}) + 2\eta^2 L_s,$$

where for each $i \leq j \in [n]$, we have defined

$$C_s^{i,j}(x_{s+1}) \equiv \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)}) - \exp(-\eta V_s^{G,(i,j)}).$$

When x_{s+1} is clear from context, for notational economy, we will elide it and simply write $C_s^{i,j}$.

As in Section 4, we defer proofs that mirror previous arguments to the Appendix.

Next, we abuse notation and write $V_s^{G,(\ell,u)}$ to denote $V_s^{G,(i,j)}$ for $i, j \in [n] \times [n]$ such that $(\ell, u) \in B_n(i,j)$. Given $(\ell, u) \in \mathcal{P}_{\text{interval}}$ such that $(\ell, u) \in B_n(i,j)$, we let $C_s^{\ell,u} \equiv C_s^{i,j}$, with the latter defined in the statement of Lemma 5.1. That is, fixing π_s and x_{s+1}, for any $(\ell, u) \in \mathcal{P}_{\text{interval}}$ such that $(\ell, u) \in B_n(i,j)$,

$$C_s^{\ell,u}(x_{s+1}) \equiv C_s^{i,j}(x_{s+1}) = \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)}) - \exp(-\eta V_s^{G,(i,j)}), \tag{11}$$

where in turn the V's are as defined in Definition 5.1.

Lemma 5.2 (Value of the Game)**.** *For any $x_{s+1} \in \mathcal{X}$, any adversary restricted to playing (ρ, rn)-smooth distributions, and any transcript $\pi_s \in \Pi^*$, there exists a distribution over predictions for the learner $Q^L_{s+1} \in \Delta\mathcal{P}^{rn}_{interval}$ which guarantees that:*

$$\mathop{\mathbb{E}}_{(\overline{\ell},\overline{u}) \sim Q^L_{s+1}} \left[\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1})) \right] \leq L_s \left(\eta\rho + 2\eta^2 \right).$$

Proof. We again proceed by defining a zero-sum game with objective function equal to the upper bound on $\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1}))$ that we proved in Lemma 5.1:

$$u((\ell, u), y) = \eta \cdot v_\delta((\ell, u), y) \cdot C_s^{\ell,u} + 2\eta^2 L_s.$$

Here, the strategy space for the learner (the minimization player) is the set of all distributions over $\mathcal{P}^{rn}_{\text{interval}}$: $\mathcal{Q}^L = \Delta\mathcal{P}^{rn}_{\text{interval}}$. A priori, the strategy space for the adversary is $\mathcal{Q}_{\rho,rn}$ the set of all (ρ, rn)-smooth distributions, but we show that it suffices to take $\mathcal{Q}^A = \hat{\mathcal{Q}}_{\rho,rn}$, the set of all discrete (ρ, rn)-smooth distributions (i.e. restricting the adversary in this way does not change the value of the game).

Lemma 5.3. *For any strategy $Q^L \in \Delta\mathcal{P}^{rn}_{interval}$ for the learner, the adversary has a best response amongst the set of all (ρ, rn)-smooth distributions with support only over the discretization $\{0, 1/rn, \ldots, 1\}$. In other words, for any $Q^L \in \Delta\mathcal{P}^{rn}_{interval}$, there exists a $\hat{Q}^A \in \hat{\mathcal{Q}}_{\rho,rn}$ such that:*

$$\hat{Q}^A \in \operatorname*{argmax}_{Q^A \in \mathcal{Q}_{\rho,rn}} \mathop{\mathbb{E}}_{\substack{(\ell,u)\sim Q^L, \\ y \sim Q^A}} [u((\ell, u), y)].$$

Proof. Fix any $Q^{A'} \in \operatorname{argmax}_{Q^A \in \mathcal{Q}_{\rho,rn}} \mathbb{E}_{(\ell,u)\sim Q^L, y\sim Q^A}[u((\ell, u), y)]$ — i.e. an arbitrary (ρ, rn)-smooth best response for the maximization player. We will construct a discrete (ρ, rn)-smooth $\hat{Q}^A \in \hat{\mathcal{Q}}_{\rho,rn}$ that obtains the same objective value, as follows. For each $\frac{i}{rn} \in \{0, 1/rn, \ldots, 1\}$, let:

$$\Pr_{y\sim Q^A}\left[y = \tfrac{i}{rn}\right] = \Pr_{y\sim Q^{A'}}\left[y \in \left[\frac{i}{rn}, \frac{i+1}{rn}\right)\right].$$

Observe first by construction that Q^A is a discrete probability distribution (because $Q^{A'}$ is a probability distribution over $[0, 1]$, and the set of intervals $[\frac{i}{rn}, \frac{i+1}{rn})$ partition the unit interval), and that Q^A is (ρ, rn)-smooth because $Q^{A'}$ is (ρ, rn)-smooth — we have $\Pr_{y\sim Q^A}[y = \frac{i}{rn}] \le \rho$ for all i. Finally observe that (by definition) for any $(\ell, u) \in \mathcal{P}^{rn}_{\text{interval}}$, $\ell, u \in \{0, 1/rn, \ldots, 1\}$.

Therefore, we have that for any $(\ell, u) \in \mathcal{P}^{rn}_{\text{interval}}$, any $i \in \{0, 1, \ldots, n\}$, and any $y, y' \in \left[\frac{i}{rn}, \frac{i+1}{rn}\right)$, $u((\ell, u), y) = u((\ell, u), y')$. To see this, note that $y \ge \ell$ if and only if $y' \ge \ell$, and $y < u$ if and only if $y' < u$. Since $v_\delta((\ell, u), y)$ is a function only of the indicators of the event that $\ell \le y < u$, this proves the claim. □

Recall (from Observation 5.2) that for any (ρ, rn)-smooth label distribution Q^A, there exists an interval $(\ell, u) \in \mathcal{P}^{rn}_{\text{interval}}$ such that $|\Pr_{y\sim Q^A}[y \in [\ell, u)] - (1 - \delta)| \le \rho$, meaning there exists $(\overline{\ell}, \overline{u})$ such that $\mathbb{E}_{\tilde{y}_{s+1}}[v_\delta((\overline{\ell}, \overline{u}), \tilde{y}_{s+1})] \le \rho$. We can thus bound the value of the game we have defined as follows:

$$\begin{aligned}
\max_{Q^A \in \hat{\mathcal{Q}}_{\rho,rn}} \min_{(\ell,u)\in\mathcal{P}^{rn}_{\text{interval}}} \mathop{\mathbb{E}}_{y\sim Q^A}[u(\ell, u), y] &\le \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(\ell,u)})\,(\eta\rho) + \exp(-\eta V_s^{G,(\ell,u)})\,(\eta\rho) + 2\eta^2 L_s, \\
&\le L_s(\eta\rho + 2\eta^2).
\end{aligned}$$

It is easy to verify that $\Delta\mathcal{P}^{rn}_{\text{interval}}$ and $\hat{\mathcal{Q}}_{\rho,rn}$ are both compact sets (closed and bounded in a finite dimensional Euclidean space) and convex. The lemma then follows by applying the minimax theorem (Theorem 2.1). □

Corollary 5.1. *For every $s \in [T]$, $\pi_s \in \Pi^*$, and $x_{s+1} \in \mathcal{X}$ (which fixes L_s and Q^L_{s+1}), and any distribution over $\mathcal{Y}$:*

$$\mathop{\mathbb{E}}_{(\ell,u)\sim Q^L_{s+1}}[\tilde{L}_{s+1}|\pi_s] \le L_s + \mathop{\mathbb{E}}_{(\overline{\ell},\overline{u})\sim Q^L_{s+1}}\left[\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1}))\right] < L_s\left(1 + \eta\rho + 2\eta^2\right).$$

As with mean multicalibration, Lemma 5.2 defines (existentially) an algorithm that the learner can use to make predictions — Algorithm 5. We will now show that Algorithm 5 (if we could compute the distributions Q_t^L) results in multivalid prediction intervals.

Algorithm 5: A Generic Multivalid Predictor

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

Observe x_t. Given π_{t-1} and x_t, let $Q_t^L \in \Delta\mathcal{P}^{rn}_{\text{interval}}$ be the distribution over prediction intervals whose existence is established in Lemma 5.2.

Sample $(\overline{\ell}, \overline{u}) \sim Q_t^L$ and predict $(\overline{\ell}_t, \overline{u}_t) = (\overline{\ell}, \overline{u})$

Lemma 5.4. *Against any adversary who is constrained to playing (ρ, rn)-smooth distributions, Algorithm 5 results in surrogate loss satisfying:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T] \leq 2|\mathcal{G}|n^2 \exp\left(T\eta\rho + 2T\eta^2\right).$$

Proof. Using Corollary 5.1, the first part of Theorem 3.1 applies in this case to the process L with $L_0 = 2|G|n^2$ and $c = \rho$. The bound follows by plugging these values into (3). □

Finally, we can calculate a bound on our expected multivalidity error. The proof (which mirrors similar claims in previous sections) is in the Appendix.

Theorem 5.1. *When Algorithm 5 is run using n buckets, discretization parameter r and $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)}{2T}} \in (0, 1/2)$, then against any adversary constrained to playing (ρ, rn)-smooth distributions, its sequence of interval predictions is α-multivalid with respect to $\mathcal{G}$ in expectation over the randomness of the transcript π_T, where:*

$$\mathbb{E}[\alpha] \leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2)}{T}}.$$

We can also use the second part of Theorem 3.1 to prove a high probability bound on the multicalibration error of Algorithm 5. The proof is in the Appendix.

Theorem 5.2. *When Algorithm 5 is run using n buckets, discretization parameter r and $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)}{2T}} \in (0, 1/2)$, then against any adversary who is constrained to playing (ρ, rn)-smooth distributions, its sequence of interval predictions is α-multivalid with respect to $\mathcal{G}$ with probability $1 - \lambda$ over the randomness of the transcript π_T:*

$$\alpha \leq \rho + 4\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n^2}{\lambda}\right)}.$$

Remark 5.2. *The hypothesis of our theorems has an assumption: that the adversary is restricted to playing (ρ, rn)-smooth distributions. This may be reasonable if we are not in a truly adversarial setting, and are simply concerned with unknown distribution shift. But what if we are truly in an adversarial environment? It turns out that in order to have a useful algorithm, we need not make* any *assumptions on the adversary at all. Observe that if we randomly perturb observed labels with uniform noise: $\hat{y}_t = y_t + U(-\epsilon, \epsilon)$, then the distribution on our perturbed points will be $\left(\frac{1}{2rn\epsilon}, rn\right)$-smooth by construction. Now recall that r is a parameter that we can select. By taking $r = \frac{1}{2\rho n\epsilon}$, we obtain that the distribution on the perturbed points is (ρ, rn)-smooth, for a value of ρ that we can take as small as we like. Taking $\rho = 1/\sqrt{T}$ ($r = \frac{\sqrt{T}}{2n\epsilon}$) makes the contribution of ρ to the multivalidity error a low order term. If we feed these perturbed labels to our algorithm, we will obtain prediction intervals that are multivalid for the perturbed labels. But observe that if we simply widen each of our prediction intervals by ϵ at each end, so that we predict the interval $[\overline{\ell}_t - \epsilon, \overline{u}_t + \epsilon)$, then our intervals continue to have coverage probability at least $1 - \delta$ for the original, unperturbed labels. We can similarly take ϵ as small as we like. Our algorithm in Section 5.3 will have running time depending polynomially on r, so with this construction obtains a polynomial dependence on $1/\epsilon$.*

5.3 Deriving an Efficient Algorithm via Equilibrium Computation

In this section, we show how to implement Algorithm 5 to efficiently sample from the distributions Q_t^L whose existence we established in Lemma 5.2. We do this by efficiently computing an equilibrium strategy Q_t^L using the Ellipsoid algorithm by solving the linear program in Figure 2. This linear program has $(rn)^2+1$ variables and (a priori) an infinite number of constraints. However, as we will show:

1. The number of constraints can in fact be taken to be finite (albeit exponentially large), and
2. We have an efficient separation oracle to identify violated constraints.

Together, this allows us to apply the Ellipsoid algorithm.

$$\begin{aligned}
&\min_{Q^L \in \mathcal{P}^{rn}_{\text{interval}}} \quad \gamma \text{ s.t.} \\
\forall Q^A \in \hat{\mathcal{Q}}_{\rho,rn} : &\sum\nolimits_{y \in \mathcal{P}^{rn}} Q^A(y) \left(\sum\nolimits_{(\ell,u) \in \mathcal{P}^{rn}_{\text{interval}}} Q^L((\ell,u)) \left(v_\delta((l,u),y) C^{\ell,u}_{t-1}(x_t) \right) \right) \leq \gamma, \\
&\sum\nolimits_{(\ell,u) \in \mathcal{P}^{rn}_{\text{interval}}} Q^L((\ell,u)) = 1, \\
\forall\, (\ell,u) \in \mathcal{P}^{rn}_{\text{interval}} : &\, Q^L((\ell,u)) \geq 0.
\end{aligned}$$

Figure 2: A Linear Program for Computing a Minimax Equilibrium Strategy for the Learner at Round t.

Algorithm 6: Von Neumann's Multivalid Predictor

INPUT: $\epsilon > 0$.
for $t = 1, \ldots, T$ **do**
  Observe x_t and compute $C^{\ell,u}_{t-1}(x_t)$ for each $(\ell,u) \in \mathcal{P}^{rn}_{\text{interval}}$ as in (11).
  Solve the Linear Program from Figure 2 using the Ellipsoid algorithm, with Algorithm 7 as a separation oracle, to obtain an ϵ-approximate solution $Q_t^L \in \Delta\mathcal{P}^{rn}_{\text{interval}}$.
  Predict $(\overline{\ell}_t, \overline{u}_t) = (\ell, u)$ with probability $Q_t^L((\ell,u))$.

Theorem 5.3. *Algorithm 6 implements Algorithm 5. In particular, it obtains multivalidity guarantees arbitrarily close to those of Theorems 5.1 and 5.2. Namely, for any desired $\epsilon > 0$, we have the following.*

Choosing $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2+\epsilon)}{2T}} \in (0, 1/2)$, we have against any adversary constrained to playing (ρ, rn)-smooth distributions that the sequence of prediction intervals produced by Algorithm 6 is α-multivalid with respect to $\mathcal{G}$ in expectation over the randomness of the transcript π_T, where:

$$\mathbb{E}[\alpha] \leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2+\epsilon)}{T}}.$$

Moreover, choosing $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)+\epsilon T}{2T}} \in (0, 1/2)$, we have, with probability $1-\lambda$ over the randomness of the transcript π_T,

$$\alpha \leq \rho + 4\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n^2}{\lambda}\right) + 2\epsilon}.$$

The runtime of Algorithm 6 is linear in $|\mathcal{G}|$, and polynomial in r, n, T, and $\log(\frac{1}{\epsilon})$.

Remark 5.3. *As with all of our other algorithms, the dependence on $|\mathcal{G}|$ can be replaced at each round with a possibly substantially smaller dependence on the number of groups which contain x_t, $|\mathcal{G}(x_t)|$, whenever this set is efficiently enumerable.*

Proof. Recall that at each round t we need to find an equilibrium strategy for the learner in the zero-sum game defined by the objective function:

$$\begin{aligned} u((\ell,u),y) &= \eta v_\delta((\ell,u),y)C_{t-1}^{\ell,u} + 2\eta^2 L_{t-1} \\ &= \eta\left(\text{Cover}((\ell,u),y) - (1-\delta)\right)C_{t-1}^{\ell,u} + 2\eta^2 L_{t-1}. \end{aligned}$$

In this game, the strategy space for the learner is the set of all distributions over discrete intervals: $\mathcal{Q}^L = \Delta\mathcal{P}^{rn}_{\text{interval}}$, and (by Lemma 5.3), the action space for the adversary can be taken to be the set of all discrete smooth distributions: $\mathcal{Q}^A = \hat{\mathcal{Q}}_{\rho,rn}$.

The equilibrium structure of a game is invariant to adding and multiplying the objective function by a constant. Hence we can proceed to solve the game with the objective function:

$$u((\ell,u),y) = \left(\text{Cover}((\ell,u),y) - (1-\delta)\right)C_{t-1}^{\ell,u}.$$

To compute an equilibrium of the game, we need to solve for a distribution Q^L satisfying:

$$Q^L \in \operatorname*{argmin}_{Q^L \in \Delta\mathcal{P}^{rn}_{\text{interval}}} \max_{Q^A \in \hat{\mathcal{Q}}_{\rho,rn}} \mathop{\mathbb{E}}_{\substack{y\sim Q^A,\\(\ell,u)\sim Q^L}} [u(\ell,u),y)].$$

We can write this as a linear program, over the $O((rn)^2)$ variables $Q^L((\ell,u))$: see Figure 2. A priori, this linear program has infinitely many constraints.[7] Nevertheless, we show that we can efficiently implement a *separation oracle*, which given a candidate solution (Q^L,γ), can find a violated constraint whenever one exists. This is sufficient to efficiently find, using the Ellipsoid algorithm, a feasible solution of the linear program achieving value within any desired $\epsilon > 0$ of the optimum.

Algorithm 7: A Separation Oracle for Linear Program 2

INPUT: A proposed solution Q^L, γ for Linear Program 2
OUTPUT: A violated constraint of Linear Program 2 if one exists, or a certification of feasibility.
for $i = 0, 1 \ldots, rn$ **do**
Compute

$$W_i \equiv \sum\nolimits_{(\ell,u)\in\mathcal{P}^{rn}_{\text{interval}}:\text{Cover}((\ell,u),\frac{i}{rn})=1} Q^L((\ell,u))C_{t-1}^{\ell,u}$$

Let $\sigma : \{0,\ldots,rn\} \to \{0,\ldots,rn\}$ be a permutation such that:

$$W_{\sigma(0)} \geq W_{\sigma(1)} \geq \ldots \geq W_{\sigma(rn)}.$$

for $i = 0, 1 \ldots, rn$ **do**
Set $Q^A(\sigma(i)) = \min(\rho, 1 - \sum_{j=0}^{i-1} Q^A(\sigma(j))$
if $\sum_{y\in\mathcal{P}^{rn}} Q^A(y)\left(\sum_{(\ell,u)\in\mathcal{P}^{rn}_{\text{interval}}} Q^L((\ell,u))\left(v_\delta((l,u),y)C_{t-1}^{\ell,u}\right)\right) > \gamma$, or Q^L not a prob. dist. **then**
return the violated constraint.
return FEASIBLE

We will identify the output of Algorithm 5 with the distribution Q^A associated with the constraint it outputs. Observe that if there is a violation (i.e. the proposed solution Q^L, γ is infeasible), and there are ties, i.e. indices i and j such that $W_i = W_j$, then there are multiple candidate Q^A's that could be the output of Algorithm 7. To that end, note that a solution Q^A can be output by Algorithm 7 if and only if it is *greed-induced*:

Definition 5.6. *Let W_i be defined as in Algorithm 7 for $i \in \{0,\ldots,rn\}$. We say that a distribution $Q^L \in \hat{\mathcal{Q}}_{\rho,rn}$ is* greed-induced *if for every pair of indices i and j such that $W_i > W_j$:*

$$Q^A(j) > 0 \implies Q^A(i) = \rho.$$

[7] Although in fact, in the proof of Lemma 5.5, we will show that without loss of generality we can equivalently impose only finitely (but exponentially) many constraints.

Lemma 5.5. *Algorithm 7 is a separation oracle for the Linear Program in Figure 2. It runs in time* $O((rn)^3)$.

Proof. Recall that a separation oracle is given a candidate distribution $Q^L \in \Delta\mathcal{P}^{rn}_{\text{interval}}$ and a value $\gamma \in \mathbb{R}$, and must determine if there is any $Q^A \in \hat{\mathcal{Q}}_{\rho,rn}$ such that:

$$\sum\nolimits_{y\in\mathcal{P}^{rn}} Q^A(y)\left(\sum\nolimits_{(\ell,u)\in\mathcal{P}^{rn}_{\text{interval}}} Q^L((\ell,u))\left(v_\delta((l,u),y)C^{\ell,u}_{t-1}\right)\right) > \gamma.$$

Suppose the learner is playing a distribution $Q^L \in \Delta\mathcal{P}^{rn}_{\text{interval}}$ over intervals. The adversary will seek to maximize the objective function over the set of (ρ, rn)-smooth distributions $Q^A \in \hat{\mathcal{Q}}_{\rho,rn}$. Recall that $v_\delta((\ell,u),y) = \text{Cov}((\ell,u),y) - (1-\delta)$. Therefore, fixing a distribution Q^L for the learner, there are terms in the objective function that are independent of the adversary's actions (roughly, those corresponding to the $(1-\delta)$ term), and hence irrelevant to the inner maximization problem (i.e the adversary's best response). We define the following quantity $\tilde{u}$ which eliminates these y-independent terms:

$$\begin{aligned}\tilde{u}(Q^L, Q^A) &= \sum_{i\in\{0,\ldots,rn\}} Q^A\left(\frac{i}{rn}\right)\sum\nolimits_{(\ell,u)\in\mathcal{P}^{rn}_{\text{interval}}:\text{Cover}((\ell,u),\frac{i}{rn})=1} Q^L((\ell,u))C^{\ell,u}_{t-1},\\ &= \sum_{i\in\{0,\ldots,rn\}} Q^A\left(\frac{i}{rn}\right)W_i.\end{aligned}$$

Observe that for any $Q^L \in \Delta\mathcal{P}_{\text{interval}}$:

$$\operatorname*{argmax}_{Q^A\in\hat{\mathcal{Q}}_{\rho,rn}}\left(\mathop{\mathbb{E}}_{\substack{\tilde{y}\sim Q^A,\\(\tilde{\ell},\tilde{u})\sim Q^L}}[u((\tilde{\ell},\tilde{u}),\tilde{y})]\right) = \operatorname*{argmax}_{Q^A\in\hat{\mathcal{Q}}_{\rho,rn}}\tilde{u}(Q^L,Q^A).$$

Hence, to derive a separation oracle, it suffices to find an algorithm which maximizes $\tilde{u}$ given a fixed distribution over intervals Q^L for the learner. This is how we proceed.

Observe that by the argument above, the adversary's problem is equivalent to solving:

$$\begin{aligned}\max_{Q^A}\sum\nolimits_{i\in\{0,\ldots,rn\}} Q^A\left(\frac{i}{rn}\right)W_i,\\ \sum\nolimits_{i\in\{0,\ldots,rn\}} Q^A\left(\frac{i}{rn}\right) = 1,\\ \forall\, i\in\{0,\ldots,rn\} : Q^A\left(\frac{i}{rn}\right)\le\rho,\\ \forall\, i\in\{0,\ldots,rn\} : Q^A\left(\frac{i}{rn}\right)\ge 0.\end{aligned}$$

By observation, this is a fractional knapsack problem—the value of each item $i \in \{0,\ldots,rn\}$ is W_i, the quantity of each item i is ρ, and the total capacity is 1. Therefore the optimal solution is greed-induced.

To bound the runtime of Algorithm 7, first observe that checking that Q^L is a probability distribution takes time $O((rn)^2\log rn)$. Now, we focus on the remaining constraints. Since the quantities $C^{\ell,u}_{t-1}$ are precomputed at the beginning of round t, the separation oracle computes W_i for each $i \in \{0,\ldots,rn\}$ in time $O((rn)^2)$, and hence we can compute all W_i's in time $O((rn))^3$. All that remains is to sort the indices W_i which takes time $O(rn\ln rn)$, which is a low order term. Altogether, this results in a runtime of $O((rn)^3)$ for Algorithm 7. ☐

Now, we verify that Algorithm 6 runs efficiently — to do so, we need to show that the Ellipsoid algorithm can efficiently (approximately) solve Linear Program 2.

Lemma 5.6. *Each run of the Ellipsoid algorithm within Algorithm 6 solves the LP to a desired accuracy $\epsilon > 0$ in runtime $\text{poly}(rn, \log|\mathcal{G}|, T, \log\frac{1}{\epsilon})$. Consequently, Algorithm 6 runs in time $\text{poly}(rn, |\mathcal{G}|, T, \log\frac{1}{\epsilon})$, where the dependence on $|\mathcal{G}|$ is $O(|\mathcal{G}|)$.*

Proof. To ensure the Ellipsoid has polynomial runtime, we need to satisfy the conditions of Theorem 4.4.

We first check that the feasible set of Linear Program 2 is a polyhedron, i.e. that it has *finitely many faces*. By Lemma 5.5 above, the adversary always has a *greed-induced* best-response Q^A constructed by Algorithm 7. Every distribution Q^A output by Algorithm 7 corresponds to selecting $\lfloor\frac{1}{\rho}\rfloor$ "full" buckets that will have probability ρ each and one bucket for the remaining probability mass, so there are at most $rn \cdot \binom{rn}{\lfloor\frac{1}{\rho}\rfloor} = O(rn \cdot 2^{rn})$ such distributions. The feasible set of Linear Program 2 is thus equivalently given by the corresponding finitely many $(O(rn \cdot 2^{rn}))$ constraints.

Thus, the feasible region of LP 2 is indeed a polyhedron; however, exponential terms in the coefficients of the constraints associated with the adversarial best-responses (which are due to our definition of the soft-max surrogate loss) prevent it from being *rational*. To fix this, we only keep $O(\log\frac{1}{\epsilon})$ bits of precision after the integer part of every coefficient of the original LP, resulting in a new LP whose coefficients are all rational and within $\pm\frac{\epsilon}{2}$ from their original values in LP 2. The new LP indeed has a rational polyhedron as its feasible region.

We now observe that Linear Program 2 has the form given in Lemma 4.5. This implies that by solving the just described rational LP corresponding to LP 2 *exactly*, we will obtain the desired ϵ-*approximate* solution to Linear Program 2. With this in mind, it remains to bound the bit complexity of the rational LP.

Consider any constraint of the rational LP. The coefficient of each variable $Q^L((\ell, u))$ has absolute value at most:

$$\begin{aligned}
\max_{(\ell,u)\in\mathcal{P}_{\text{interval}}} \sum_{G\in\mathcal{G}} \exp(\eta V_{t-1}^{G,(\ell,u)}) - \exp(-\eta V_{t-1}^{G,(\ell,u)}) &\leq |\mathcal{G}| 2\exp\left(\eta \max_{G\in\mathcal{G},(\ell,u)\in\mathcal{P}_{\text{interval}}} \left|V_{t-1}^{G,(\ell,u)}\right|\right) \\
&\leq 2|\mathcal{G}|\exp(\eta T) \\
&\leq 2|\mathcal{G}|\exp(T).
\end{aligned}$$

Thus, every constraint in the rational LP has bit complexity at most:

$$O\left((rn)^2 \cdot \left(\log|\mathcal{G}| + T + \log\frac{1}{\epsilon}\right)\right),$$

where the $\log\frac{1}{\epsilon}$ term reflects the chosen precision. This is polynomial in $r, n, T, \log|\mathcal{G}|$, and $\log\frac{1}{\epsilon}$. Also, the objective function, which is simply γ, takes $O((rn)^2)$ bits to write down.

We may now apply Theorem 4.4 with the parameters $q = O((rn)^2)$, $\phi = O\left((rn)^2(\log|\mathcal{G}| + T + \log\frac{1}{\epsilon})\right)$, $c = O((rn)^2)$. The runtime of the separation oracle (which, we note, applies to the rational LP just as it did for the original LP) is $O((rn)^3)$ by Lemma 5.5. Hence, the Ellipsoid algorithm will solve Linear Program 2 with accuracy ϵ in time $\text{poly}(rn, \log|\mathcal{G}|, T, \log\frac{1}{\epsilon})$.

Hence, Algorithm 6 has time complexity $\text{poly}(rn, |\mathcal{G}|, T, \log\frac{1}{\epsilon})$ — where the dependence on $|\mathcal{G}|$ is linear, because we precompute the $C_{t-1}^{\ell,u}$'s once at the beginning of each round t, taking time linear in $|\mathcal{G}|$, and the runtime of the Ellipsoid algorithm is polylogarithmic in $|\mathcal{G}|$. (We remark once more that the dependence on $|\mathcal{G}|$ can be reduced to a dependence on $|\mathcal{G}(x_t)|$ if $\mathcal{G}(x_t)$ is efficiently enumerable, and that this might be much smaller.) □

Finally, we need to demonstrate that the claimed multivalidity guarantees (which are a function of the chosen $\epsilon > 0$) indeed hold.

Lemma 5.7. *Algorithm 6 achieves the multivalidity guarantees stated in Theorem 5.3.*

The proof of this lemma involves repeating several calculations from Section 5.2 with an ϵ error term, and so is deferred to the Appendix. □

6 Augmenting an Existing Learning Algorithm

For simplicity of exposition, throughout this paper, we have described our algorithms as predicting properties of the arriving labels y_t directly. But often that is not what we want: instead, we have some procedure $f_t : \mathcal{X} \to \mathcal{Y}$ making point predictions — that is, mapping features to labels — and we are interested in properties of the *residuals* $f_t(x_t) - y_t$. For example, f_t may be some complicated (but powerful) learning procedure — for example, maybe at every round, we train a neural network on the data we have observed so far to predict the labels of new observations. It may be that the labels y have high variance, but that the residuals $y_t - f_t(x_t)$ are tightly concentrated around zero (because f_t is highly accurate). To quantify the uncertainty of our predictions, we want to provide prediction intervals related to our predictions $f_t(x_t)$ — that is, to compute prediction intervals for the residuals. We may similarly be interested in the variance of the residuals, etc.

We can easily use the algorithms we have developed in this paper for this. We have no understanding of f_t or the distribution on predictions $f_t(x_t)$ it induces (say, because f_t varies substantially from round to round because of retraining) — but because our algorithms handle adversarially chosen sequences of examples, they apply equally well when we feed them the residuals rather than the original labels. We have derived our algorithms under the scaling that $y_t \in [0, 1]$, and the residuals $y_t - f_t(x_t)$ may lie in $[-1, 1]$, so to apply the same bounds we have derived, we need to compute *centered residuals* $y'_t = \frac{1}{2} + \frac{1}{2}(y_t - f_t(x_t))$. (This simply corresponds to a rescaling and a shift so that the residuals again lie in $[0, 1]$. Thus, the following algorithm is able to provide prediction intervals around the predictions of an arbitrary sequence of predictors f_t (and similar constructions work for predicting means and variances of the residuals):

Algorithm 8: Endowing Arbitrary Point Predictors with Prediction Intervals

Instantiate $\mathcal{A}$, a copy of Algorithm 6.
for $t = 1, \ldots, T$ **do**
  Observe x_t, and compute a point prediction $f_t(x_t)$ (for an arbitrary procedure f_t).
  Feed x_t to $\mathcal{A}$ and receive a prediction interval $(\overline{\ell}_t, \overline{u}_t)$.
  Output point prediction $f_t(x_t)$ and prediction interval $(f_t(x_t) + 2\overline{\ell}_t - 1, f_t(x_t) + 2\overline{u}_t - 1)$.
  Observe y_t and feed the centered residual $y'_t = \frac{1}{2} + \frac{1}{2}(y_t - f_t(x_t))$ to $\mathcal{A}$

We observe that $y_t \in [f_t(x_t) + 2\overline{\ell}_t - 1, f_t(x_t) + 2\overline{u}_t - 1)$ if and only if $y'_t \in [\overline{\ell}_t, \overline{u}_t)$ by construction, and so the prediction intervals produced by Algorithm 8 inherit the (α, n)-multivalidity guarantees of Algorithm 6 (Theorems 5.1 and 5.2): that with probability $1 - \lambda$:

$$\alpha \leq \rho + 4\sqrt{\frac{2}{T} \ln\left(\frac{2|\mathcal{G}|n^2}{\lambda}\right)}.$$

(the bound on expected multivalidity error holds as well). Here ρ is a smoothness parameter that depends on both the discretization r we choose for our algorithm and the distribution over residuals at each round. Note that as discussed in Remark 5.2, with an appropriate selection of r, for any $\epsilon > 0$, we can make ρ as small as we like by perturbing the centered residuals y'_t with uniform noise $U(-\epsilon, \epsilon)$, at the cost of needing to widen our prediction intervals by ϵ on each end, i.e. predicting at each round:

$$(f_t(x_t) + 2\overline{\ell}_t - 1 - \epsilon, f_t(x_t) + 2\overline{u}_t - 1 + \epsilon).$$

The computational cost of this is polynomial in $1/\epsilon$ and $1/\rho$, and the gain that we get by applying these perturbations is that we need assume nothing at all about either the adversarial sequence of examples, or about the properties of our predictors f_t.

Acknowledgements

We thank Aaditya Ramdas for helpful discussions about conformal prediction, as well as pointers to the literature. We thank Sergiu Hart, Dean Foster, Drew Fudenberg, and Rakesh Vohra for helpful discussions

about calibration, as well as pointers to the literature. We also thank Ashish Rastogi for discussions about uncertainty estimation in practice. Gupta, Jung, Noarov, and Roth are supported in part by NSF grants CCF-1763307 and CCF-1934876, and a grant from the Simons Foundation. Pai is supported in part by NSF grant CCF-1763349.

References

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.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

Alexandra Chouldechova and Aaron Roth. A snapshot of the frontiers of fairness in machine learning. *Communications of the ACM*, 63(5):82–89, 2020.

A Philip Dawid. The well-calibrated bayesian. *Journal of the American Statistical Association*, 77(379): 605–610, 1982.

Devdatt P Dubhashi and Alessandro Panconesi. *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press, 2009.

Cynthia Dwork, Michael P Kim, Omer Reingold, Guy N Rothblum, and Gal Yona. Learning from outcomes: Evidence-based rankings. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 106–125. IEEE, 2019.

Cynthia Dwork, Michael P Kim, Omer Reingold, Guy N Rothblum, and Gal Yona. Outcome indistinguishability. *arXiv preprint arXiv:2011.13426*, 2020.

Dean P Foster. A proof of calibration via blackwell's approachability theorem. *Games and Economic Behavior*, 29(1-2):73–78, 1999.

Dean P Foster and Sergiu Hart. Smooth calibration, leaky forecasts, finite recall, and nash dynamics. *Games and Economic Behavior*, 109:271–293, 2018.

Dean P Foster and Sergiu Hart. Forecast-hedging and calibration. 2019.

Dean P Foster and Sham M Kakade. Calibration via regression. In *2006 IEEE Information Theory Workshop-ITW'06 Punta del Este*, pages 82–86. IEEE, 2006.

Dean P Foster and Rakesh V Vohra. Asymptotic calibration. *Biometrika*, 85(2):379–390, 1998.

Dean P Foster, Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Complexity-based approach to calibration with checking rules. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 293–314, 2011.

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.

Drew Fudenberg and David K Levine. An easier way to calibrate. *Games and economic behavior*, 29(1-2): 131–137, 1999a.

Drew Fudenberg and David K Levine. Conditional universal consistency. *Games and Economic Behavior*, 29(1-2):104–130, 1999b.

Chirag Gupta, Aleksandr Podkopaev, and Aaditya Ramdas. Distribution-free binary classification: prediction sets, confidence intervals and calibration. *Advances in Neural Information Processing Systems*, 33, 2020.

Sergiu Hart. Calibrated forecasts: The minimax proof. 2020. URL http://www.ma.huji.ac.il/~hart/papers/calib-minmax.pdf.

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.

Christopher Jung, Changhwa Lee, Mallesh M Pai, Aaron Roth, and Rakesh Vohra. Moment multicalibration for uncertainty estimation. *arXiv preprint arXiv:2008.08037*, 2020.

Sham M Kakade and Dean P Foster. Deterministic calibration and nash equilibrium. In *International Conference on Computational Learning Theory*, pages 33–48. Springer, 2004.

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.

Michael P Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 247–254, 2019.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

Ehud Lehrer. Any inspection is manipulable. *Econometrica*, 69(5):1333–1347, 2001.

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.

David Oakes. Self-calibrating priors do not exist. *Journal of the American Statistical Association*, 80(390): 339–339, 1985.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. *Advances in Neural Information Processing Systems*, 30:5680–5689, 2017.

Mingda Qiao and Gregory Valiant. Stronger calibration lower bounds via sidestepping. *arXiv preprint arXiv:2012.03454*, 2020.

Alvaro Sandroni, Rann Smorodinsky, and Rakesh V Vohra. Calibration with many checking rules. *Mathematics of operations Research*, 28(1):141–153, 2003.

Alexander Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Inc., USA, 1986. ISBN 0471908541.

Eliran Shabat, Lee Cohen, and Yishay Mansour. Sample complexity of uniform convergence for multicalibration. *arXiv preprint arXiv:2005.01757*, 2020.

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.

Rakesh V Vohra. *Advanced mathematical economics*. Routledge, 2004.

A Batch Prediction

A.1 Preliminaries

In the batch setting, there is an (unknown) probability distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. Let $\mathcal{D}_{\mathcal{X}}$ refer to the induced marginal distribution on $\mathcal{X}$ and let $\mathcal{D}_{\mathcal{Y}}$ refer to the induced marginal distribution on $\mathcal{Y}$. In the batch setting, rather than talking about a sequence of predictions, we need to refer to calibration properties of a single predictor with respect to the data distribution. We here modify the definition of consistency and calibration accordingly — but we will show how to convert calibration guarantees from the online setting to calibration guarantees in the offline setting.

Given n independent draws from $\mathcal{D}$, denoted by $D = \{(x_t, y_t)\}_{t=t}^{T}$ the corresponding dataset. Given some x, our goal is to predict various properties of $\mathcal{D}|x$.

Mean Predictions For mean prediction, we use a (possibly randomized) predictor $\overline{\mu} : \mathcal{X} \to [0, 1]$ that tries to predict the conditional mean $\mathbb{E}[y|x]$. Given a set $S \subseteq \mathcal{X}$, we write

$$\mu(S) = \mathop{\mathbb{E}}_{\overline{\mu}}[\mathop{\mathbb{E}}_{\mathcal{D}}[y|x \in S]], \quad \overline{\mu}(S) = \mathop{\mathbb{E}}_{\overline{\mu}}[\mathop{\mathbb{E}}_{\mathcal{D}}[\overline{\mu}(x)]|x \in S]]$$

for the conditional mean of labels on the distribution conditional on $x \in S$ and our conditional mean prediction. For calibration guarantees, we will be concerned with sets that depend on realizations of the randomized predictor $\overline{\mu}$, so it is important that in the above expressions, S appears inside the expectation over $\overline{\mu}$. Otherwise, we essentially use the same notation as in the online setting except instead of averaging over the empirical distribution, we average over the true distribution.

As in the online setting, we "bucket" our real valued predictions into n buckets of width $\frac{1}{n}$, which serves as a measure of granularity of our calibration guarantee. Given a set $S \subseteq \mathcal{X}$ and mean predictor $\overline{\mu}$, we write

$$S(\overline{\mu}, i) \equiv \{x \in S : \overline{\mu}(x) \in B_n(i)\}$$

to be the set of points in S whose mean predictions fall into the i^{th} bucket. When $\overline{\mu}$ is a randomized predictor, we think of $S(\overline{\mu}, i)$ as a random set where the randomness is over the random bits of $\overline{\mu}$.

Definition A.1 (Mean Consistency)**.** *Call a mean predictor $\overline{\mu}$ α-mean consistent on a set S over distribution $\mathcal{D}$ if*

$$|\mu(S) - \overline{\mu}(S)| \leq \frac{\alpha}{\Pr_{\overline{\mu}, \mathcal{D}_{\mathcal{X}}}[x \in S]}.$$

We note that we include the randomness of $\overline{\mu}$ when writing the measure of the set S because we will be interested in random sets S defined as a function of randomized predictors $\overline{\mu}$.

We are now ready to define calibration, which asks for mean consistency on particular sets defined by the mean predictor itself:

Definition A.2 (Mean-Multicalibration)**.** *Fix a set $S \subseteq \mathcal{X}$ and a true distribution $\mathcal{D}$. A mean predictor $\overline{\mu}$ is (α, n)-mean calibrated on a set S over distribution $\mathcal{D}$ if it is α-mean consistent on every set $S(\overline{\mu}, i)$ over $\mathcal{D}$, i.e. if for each $i \in [n]$:*

$$|\mu(S(\overline{\mu}, i)) - \overline{\mu}(S(\overline{\mu}, i))| \leq \frac{\alpha}{\Pr_{\overline{\mu}, \mathcal{D}_{\mathcal{X}}}[x \in S(\overline{\mu}, i)]}.$$

We say that $\overline{\mu}$ is α-mean multicalibrated with respect to (a collection of sets) $\mathcal{G}$ over $\mathcal{D}$ if it is α-mean calibrated on every $G \in \mathcal{G}$ over $\mathcal{D}$.

(Mean, Moment) Prediction In this case, we use a (randomized) predictor $\overline{\mu} : \mathcal{X} \to [0,1]$ that tries to predict the conditional label mean $\mathbb{E}[y|x]$ and a (randomized) predictor $\overline{m}^k : \mathcal{X} \to [0,1]$ that tries to predict the conditional k^{th} central moment of the label distribution $m^k(x) = \mathbb{E}[(y - \mathbb{E}[y|x])^k|x]$. We again assume that k is even so that the range of the k^{th} moment remains non-negative, but there is no obstacle other than notation to handling odd moments as well. Although for notational convenience we write $\overline{m}^k$ and $\overline{\mu}$ as separate functions, they may use correlated randomness.

Analogously to our notation for mean prediction, we write for any $S \subseteq \mathcal{X}$,

$$m^k(S) = \mathop{\mathbb{E}}_{\overline{\mu},\overline{m}^k}[\mathop{\mathbb{E}}_{\mathcal{D}}[(y - \mu(S))^k | x \in S]] \quad \overline{m}^k(S) = \mathop{\mathbb{E}}_{\overline{\mu},\overline{m}^k}[\mathop{\mathbb{E}}_{\mathcal{D}}[\overline{m}^k(x) | x \in S]].$$

to denote the empirical k^{th} central moment of the label distribution on the subsequence S and for the average of the moment prediction on S, respectively.

Definition A.3 (Moment Consistency)**.** *We say that $(\overline{\mu}, \overline{m}^k)$ is α-moment consistent on set $S \subseteq \mathcal{X}$ if*

$$|m^k(S) - \overline{m}^k(S)| \leq \frac{\alpha}{\Pr_{\overline{\mu},\overline{m}^k,\mathcal{D}_{\mathcal{X}}}[x \in S]}.$$

Once again we include the randomness of $\overline{\mu}, \overline{m}^k$ because we will be concerned with sets that are defined in terms of $\overline{\mu}$ and $\overline{m}^k$.

For any $S \subseteq \mathcal{X}$ and $i \in [n], j \in [n']$, we write

$$S(\overline{\mu}, i, \overline{m}^k, j) = \left\{x \in S : \overline{\mu}(x) \in B_n(i), \overline{m}^k(x) \in B_{n'}(j)\right\}.$$

In words, $S(\overline{\mu}, i, \overline{m}^k, j)$ corresponds to the subset of points in S in which our predicted mean falls in $B_n(i)$ and $B_{n'}(j)$.

Definition A.4 (Mean-Conditioned Moment Multicalibration)**.** *We say that $(\overline{\mu}, \overline{m}^k)$ is (α, β, n, n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$, if for every $i \in [n], j \in [n']$, and $G \in \mathcal{G}$, we have that $\overline{\mu}$ is α-mean consistent on $G(\overline{\mu}, i, \overline{m}^k, j)$ and $\overline{m}^k$ is β-moment consistent on $G(\overline{\mu}, i, \overline{m}^k, j)$:*

$$|\mu(G(\overline{\mu}, i, \overline{m}^k, j)) - \overline{\mu}(G(\overline{\mu}, i, \overline{m}^k, j))| \leq \frac{\alpha}{\Pr_{\overline{\mu},\overline{m}^k,\mathcal{D}_{\mathcal{X}}}[x \in G(\overline{\mu}, i, \overline{m}^k, j)]},$$

$$|m^k(G(\overline{\mu}, i, \overline{m}^k, j)) - \overline{m}^k(G(\overline{\mu}, i, \overline{m}^k, j))| \leq \frac{\beta}{\Pr_{\overline{\mu},\overline{m}^k,\mathcal{D}_{\mathcal{X}}}[x \in G(\overline{\mu}, i, \overline{m}^k, j)]}.$$

For convenience, we sometimes combine the mean and moment predictor into a single predictor $h : \mathcal{X} \to [0,1] \times [0,1]$ and write $h^{\overline{\mu}}(x) = h(x)[0]$ to refer to its mean prediction and $h^{\overline{m}^k}(x) = h(x)[1]$ to refer to its moment prediction. Also, we write $h(x) \in B_{n,n'}(i,j)$ if $h^{\overline{\mu}}(x) \in B_n(i)$ and $h^{\overline{m}^k}(x) \in B_{n'}(j)$. If n and n' are clear from the context, we just write $h(x) \in B(i,j)$.

Interval Prediction In this case, we want to come up with randomized predictors $\overline{\ell} : \mathcal{X} \to [0,1]$ and $\overline{u} : \mathcal{X} \to [0,1]$ such that the probability that y falls between $\overline{\ell}(x)$ and $\overline{u}(x)$ is approximately $1 - \delta$ for some specified failure probability δ. Although for notational convenience we write $\overline{\ell}$ and $\overline{u}$ as separate functions, they may use correlated randomness. Using the notation given in Section 2, we wish to devise $\overline{\ell}, \overline{u}$ such that $\mathbb{E}[\text{Cover}((\overline{\ell}(x), \overline{u}(x)), y)|x] \approx 1 - \delta$.

For any $S \subseteq \mathcal{X}$, we write

$$\overline{H}_{\overline{\ell},\overline{u}}(S) = \mathop{\mathbb{E}}_{\overline{\ell},\overline{u}}[\mathop{\mathbb{E}}_{\mathcal{D}}[\text{Cover}((\overline{\ell}(x), \overline{u}(x)), x)|x \in S]].$$

We again bucket our coverage intervals using a discretization parameter n, using the same notation as we used for moment predictions. For any $S \subseteq \mathcal{X}$ and $i \leq j \in [n]$, we write

$$S(\overline{\ell}, i, \overline{u}, j) = \left\{x \in S : \overline{\ell}(x) \in B_n(i), \overline{u}(x) \in B_n(j)\right\}.$$

For simplicity, we combine $\overline{\ell}$ and $\overline{u}$ into a single predictor $h : \mathcal{X} \to [0,1] \times [0,1]$ and write $h^{\overline{\ell}}(x) = h(x)[0]$ and $h^{\overline{u}}(x) = h(x)[1]$. We say $h(x) \in B_n(i,j)$ if $h^{\overline{\ell}}(x) \in B_n(i)$ and $h^{\overline{u}}(x) \in B_n(j)$. Also, when n is clear from the context, we just write $B(i,j)$.

We can now define multivalidity in a way analogous to how we have defined multicalibration.

Definition A.5. *We say that interval predictor $(\overline{\ell}, \overline{u})$ is α-consistent on set S with respect to the failure probability $\delta \in (0,1)$, if we have the following*

$$|\overline{H}_{\overline{\ell},\overline{u}}(S) - (1-\delta)| \leq \frac{\alpha}{\Pr_{\overline{\ell},\overline{u},\mathcal{D}}[x \in S]}.$$

Definition A.6. *The interval predictors $(\overline{\ell}, \overline{u})$ are (α, n)-multivalid with respect to δ and $\mathcal{G}$ over $\mathcal{D}$, if for every $i \leq j \in [n]$ and $G \in \mathcal{G}$, we have that the interval predictions are α-consistent on $G(\overline{\ell}, i, \overline{u}, j)$ with respect to coverage probability $1-\delta$:*

$$|\overline{H}_{\overline{\ell},\overline{u}}(G(\overline{\ell}, i, \overline{u}, j)) - (1-\delta)| \leq \frac{\alpha}{\Pr_{\overline{\ell},\overline{u},\mathcal{D}}[G(\overline{\ell}, i, \overline{u}, j)]}.$$

A.2 Online to Batch Conversion

In this section, we show how to use our online algorithms to solve the corresponding batch multicalibration problems. In doing so we obtain improved sample complexity bounds for mean and mean-conditioned moment multicalibration for the batch problem, compared to prior work Hébert-Johnson et al. [2018], Jung et al. [2020]. However, in contrast to prior work which in the batch case solves for deterministic predictors, we obtain a randomized predictor via our online-to-offline reduction.

Previously, for any sequence of feature and label pairs $\{(x_t, y_t)\}_{t=1}^T$, we have shown how to construct a sequence of randomized predictors $\{h_t\}_{t=1}^T$ such that the sequence of predictions made from the predictors $\{p_t = h_t(x_t)\}_{t=1}^T$ is multivalid. We viewed the functions $h_t(x)$ only implicitly before, but we consider them explicitly here: for mean multicalibration, $h_t(x)$ is simply the distribution on label predictions $\overline{\mu}$ that would be made by Algorithm 2 at round t, given as input $x_t = x$ after a history defined by the sequence of examples $\{(x_s, y_s)\}_{s=1}^{t-1}$.

In this section, we show that if we have a sample $D = \{(x_t, y_t)\}_{t=1}^T$ that is drawn independently from $\mathcal{D}$, we can feed each element in this sample D one-by-one to our online learning algorithm so as to obtain a sequence of predictors $\{h_t\}_{t=1}^T$. From this, we construct a single (randomized) predictor h that is multivalid over the distribution $\mathcal{D}$. h will simply be the uniform mixture over the set of predictors $\{h_t\}_{t=1}^T$.

A.2.1 Mean prediction

Algorithm 9: Von Neumann's Batch Mean Multicalibrator

INPUT: Training dataset $D = \{(x_t, y_t)\}_{t=1}^{T}$
Training: Run Algorithm 2 on the sequence of examples D to generate a transcript π_T.
Denote by $h_t(x)$ the (randomized) mapping from $\mathcal{X}$ to $[0,1]$ that Algorithm 2 induces as a function of transcript π_{t-1} (the prefix of π_T of length $t-1$).
Prediction: On input x, sample $h^{\text{mean}}(x)$ by selecting $t \sim [T]$ uniformly at random, and then sampling from $h_t(x)$.

More explicitly, select $t \sim [T]$ uniformly at random and:
Compute for each $i \in [n]$ $C_{t-1}^{i}(x)$ as defined in (2) conditioning on π_{t-1}.
if $C_{t-1}^{i}(x) > 0$ for all $i \in [n]$ **then**
  Predict $h^{\text{mean}}(x) = 1$.
else if $C_{t-1}^{i}(x) < 0$ for all $i \in [n]$ **then**
  Predict $h^{\text{mean}}(x) = 0$.
else
  Find $i^* \in [n-1]$ such that $C_{t-1}^{i^*}(,x) \cdot C_{t-1}^{i^*+1}, x) \leq 0$
  Define $0 \leq q_t \leq 1$: (using the convention that $0/0 = 1$)

$$q_t = \frac{|C_{t-1}^{i^*+1}(x)|}{|C_{t-1}^{i^*+1}(x)| + |C_{t-1}^{i^*}(x)|}$$

  Predict $h^{\text{mean}}(x) = \frac{i^*}{n} - \frac{1}{rn}$ with probability q_t and $h^{\text{mean}}(x) = \frac{i^*}{n}$ with probability $1 - q_t$.

Theorem A.1. *Let $D = \{(x_t, y_t)\}_{t=1}^{T}$ be a dataset drawn i.i.d. from $\mathcal{D}$, and suppose T is large enough such that η specified in Theorem 3.3 falls in $(0, 1/2)$. Let $\epsilon, \lambda > 0$. For an appropriately small choice of the discretization parameter r, with probability $1-\lambda$, Algorithm 9 produces a predictor h^{mean} that is (α, n)-mean multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$ where*

$$\alpha = (6+\epsilon)\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n}{\lambda}\right)}$$

Proof. In order to show that h^{mean} is (α, n)-mean multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$, it is sufficient to show for all $G \in \mathcal{G}$ and $i \in [n]$

$$\left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}, h^{\text{mean}}} [\mathbb{1}[h^{\text{mean}}(x) \in B(i), G(x) = 1] \cdot (y - h^{\text{mean}}(x))] \right| \leq \alpha.$$

We can calculate:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}, h^{\text{mean}}} [\mathbb{1}[h^{\text{mean}}(x) \in B(i), G(x) = 1] \cdot (y - h^{\text{mean}}(x))] \\
&= \sum_{(x,y)} \sum_{t=1}^{T} \mathcal{D}[(x,y)] \cdot \Pr[h^{\text{mean}} = h_t] \cdot \Pr[h_t(x) \in B(i)] \cdot \mathbb{1}[G(x) = 1] \cdot (y - h_t(x)) \\
&= \frac{1}{T} \sum_{(x,y)} \sum_{t=1}^{T} \mathcal{D}[(x,y)] \cdot \Pr[h_t(x) \in B(i)] \cdot \mathbb{1}[G(x) = 1] \cdot (y - h_t(x)) \\
&= \frac{1}{T} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}, h_t} [\mathbb{1}[h_t(x) \in B(i), G(x) = 1] \cdot (y - h_t(x))] \qquad (12)
\end{aligned}$$

Therefore, our goal is to upper bound the absolute value of (12). We will show that if $D = \{(x_t, y_t)\}_{t=1}^T$ is sampled i.i.d. from $\mathcal{D}$, the empirical calibration error on the transcript π_T generated during training serves as a good estimate for (12). And because we know from Theorem 3.3 that for every sequence of examples, Algorithm 2 produces predictions that will be empirically calibrated with high probability, our bound will follow.

In particular, we know from Theorem 3.3 that (for an appropriate choice of r) with probability $1 - \lambda/2$ over the randomness of π_T produced in training that for all $i \in [n], G \in \mathcal{G}$:

$$\left| \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\left[\overline{\mu}_t \in B(i), G(x_t) = 1\right] \cdot (y_t - \overline{\mu}_t) \right| \leq (2+\epsilon) \sqrt{\frac{2}{T} \ln\left(\frac{4|\mathcal{G}|n}{\lambda}\right)}.$$

Now, fixing $G \in \mathcal{G}$ and $i \in [n]$, we use the following martingale argument to show that (12) is close to the empirical calibration error with respect to G and i with high probability. Consider the following martingale sequence adapted to the filtration $\mathcal{F}_s = \sigma(\{(x_t, y_t), \overline{\mu}_t\}_{t=1}^s)$:

$$\tilde{Z}_s = Z_{s-1} + \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_s}\left[\mathbb{1}\left[h_s(x) \in B(i), G(x) = 1\right] \cdot (y - h_s(x)) \,|\pi_{s-1}\right] - \mathbb{1}\left[\overline{\mu}_s \in B(i), G(x_s) = 1\right] \cdot (y_s - \overline{\mu}_s).$$

It's easy to see that the above sequence is a martingale: because

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_s}\left[\mathbb{1}\left[h_s(x) \in B(i), G(x) = 1\right] \cdot (y - h_s(x)) \,|\pi_{s-1}\right] \\
&= \mathop{\mathbb{E}}_{(x_s,y_s)\sim\mathcal{D},\overline{\mu}_s}\left[\mathbb{1}\left[\overline{\mu}_s \in B(i), G(x_s) = 1\right] \cdot (y_s - \overline{\mu}_s) \,|\pi_{s-1}\right],
\end{aligned}$$

and so we have $\mathbb{E}[\tilde{Z}_s] = Z_{s-1}$.

Therefore, because $|Z_s - Z_{s-1}| \leq 2$, we can apply Azuma's inequality (Lemma D.2) to get that with probability $1 - \lambda/2$ over the randomness of π_T and D,

$$\left| \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}\left[h_t(x) \in B(i), G(x) = 1\right] \cdot (y - h_t(x))\right] - \sum_{t=1}^{T} \mathbb{1}\left[\overline{\mu}_t \in B(i), G(x_t) = 1\right] \cdot (y_t - \overline{\mu}_t) \right| \leq 2\sqrt{2T \ln\left(\frac{4}{\lambda}\right)}.$$

Therefore, Union bounding the above Azuma's inequality over all $i \in [n]$ and $G \in \mathcal{G}$ gives us the result: we have with probability $1 - \lambda$ over the randomness of $\mathcal{D}$ and π_T,

$$\begin{aligned}
&\frac{1}{T}\left| \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}\left[h_t(x) \in B(i), G(x) = 1\right] \cdot (y - h_t(x))\right] \right|, \\
\leq &\frac{1}{T}\left| \sum_{t=1}^{T} \mathbb{1}\left[\overline{\mu}_t \in B(i), G(x_t) = 1\right] \cdot (y_t - \overline{\mu}_t) \right| + 2\sqrt{\frac{2\ln\left(\frac{4|\mathcal{G}|n}{\lambda}\right)}{T}}, \\
\leq &(6+\epsilon)\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n}{\lambda}\right)}
\end{aligned}$$

for every $i \in [n]$ and $G \in \mathcal{G}$. □

A.2.2 (Mean, Moment) Prediction

We can use the same argument to show that we can feed $D = \{(x_t, y_t)\}_{t=1}^T$ drawn i.i.d. from $\mathcal{D}$ into our Algorithm 4 to obtain a randomized predictor $h^{\text{mean, moment}}$ that is (α, β, n, n')-mean-conditioned-moment

multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$.

Algorithm 10: Von Neumann's Batch Mean Moment Multicalibrator

INPUT: Training dataset $D = \{(x_t, y_t)\}_{t=1}^{T}$
Training: Run Algorithm 4 on the sequence of examples D to generate a transcript π_T.
Denote by $h_t(x)$ the (randomized) mapping from $\mathcal{X}$ to $[0,1] \times [0,1]$ that Algorithm 4 induces as a function of transcript π_{t-1} (the prefix of π_T of length $t-1$).
Prediction: On input x, sample $h^{\text{mean, moment}}(x)$ by selecting $t \sim [T]$ uniformly at random, and then sampling from $h_t(x)$.

More explicitly, select $t \sim [T]$ uniformly at random and:
Compute $C_{t-1}^{\overline{\mu},\overline{m}^k}(x), D_{t-1}^{\overline{\mu},\overline{m}^k}(x), F_{\ell,t-1}^{\overline{\mu},\overline{m}^k}(x)$ for each $(\overline{\mu},\overline{m}^k) \in \hat{\mathcal{P}}^{r,n} \times \hat{\mathcal{P}}^{r,n'}$ as in (5, 6, 9, 10) conditioning on π_{t-1}.
Find an ϵ-approximate solution to the linear program from Figure 1, to obtain solution $Q_t^L \in \hat{\mathcal{Q}}_{r,n,n'}^L$.
Predict $h^{\text{mean, moment}}(x) = (\overline{\mu},\overline{m}^k)$ with probability $Q_t^L((\overline{\mu},\overline{m}^k))$.

Theorem A.2. *Assume $T > 2\ln(\frac{8|\mathcal{G}|n\cdot n'}{\delta})$ and T is sufficiently large such that η used in Theorem 4.3 is in $(0,1/2)$. Let $D = \{(x_t,y_t)\}_{t=1}^{T}$ be a dataset drawn i.i.d. from $\mathcal{D}$. Let $\epsilon,\delta > 0$. For an appropriately small choice of the discretization parameter r, with probability $1-2\lambda$, Algorithm 10 produces a predictor $h^{\text{mean, moment}}$ that is (α,β,n,n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$ where*

$$\alpha = (6+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{8|\mathcal{G}|n\cdot n'}{\lambda}\right)+2\epsilon}$$

$$\beta = (k+3)\left((5+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{8|\mathcal{G}|n\cdot n'}{\lambda}\right)+2\epsilon}\right)+\frac{k}{2n}$$

Proof. Note that in order to show that $h^{\text{mean, moment}}$ is (α,β,n,n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$, it's sufficient to prove the following for every $i \in [n]$, $j \in [n']$, and $G \in \mathcal{G}$:

1. Mean Consistency

$$\begin{aligned}
&\left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h^{\text{mean}}}\left[\mathbb{1}[h^{\text{mean, moment}}(x)\in B(i,j),G(x)=1]\cdot\left(y-h^{\text{mean, moment}}(x)[0]\right)\right]\right| \\
&=\left|\sum_{(x,y)}\sum_{t=1}^{T}\mathcal{D}[(x,y)]\cdot\Pr[h^{\text{mean, moment}}=h_t]\cdot\Pr_{h_t}[h_t(x)\in B(i,j)]\cdot\mathbb{1}[G(x)=1]\cdot\left(y-h_t^{\overline{\mu}}(x)\right)\right| \\
&=\frac{1}{T}\left|\sum_{(x,y)}\sum_{t=1}^{T}\mathcal{D}[(x,y)]\cdot\Pr_{h_t}[h_t(x)\in B(i,j)]\cdot\mathbb{1}[G(x)=1]\cdot\left(y-h_t^{\overline{\mu}}(x)\right)\right| \\
&=\frac{1}{T}\left|\sum_{t=1}^{T}\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}[h_t(x)\in B(i,j),G(x)=1]\cdot\left(y-h_t^{\overline{\mu}}(x)\right)\right]\right| \\
&\le\alpha
\end{aligned}$$

2. Moment Consistency

$$\begin{aligned}
&\left|\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h^{\text{mean}}}\left[\mathbb{1}[h^{\text{mean, moment}}(x)\in B(i,j),G(x)=1]\cdot\left((y-A^G_{i,j})^k-h^{\text{mean, moment}}(x)[1]\right)\right]\right| \\
&=\left|\sum_{(x,y)}\sum_{t=1}^{T}\mathcal{D}[(x,y)]\cdot\Pr[h^{\text{mean, moment}}=h_t]\cdot\Pr_{h_t}[h_t(x)\in B(i,j)]\cdot\mathbb{1}[G(x)=1]\cdot\left((y-A^G_{i,j})^k-h_t^{\overline{m}^k}(x)\right)\right| \\
&=\frac{1}{T}\left|\sum_{(x,y)}\sum_{t=1}^{T}\mathcal{D}[(x,y)]\cdot\Pr_{h_t}[h_t(x)\in B(i,j)]\cdot\mathbb{1}[G(x)=1]\cdot\left((y-A^G_{i,j})^k-h_t^{\overline{m}^k}(x)\right)\right| \\
&=\frac{1}{T}\left|\sum_{t=1}^{T}\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}[h_t(x)\in B(i,j),G(x)=1]\cdot\left((y-A^G_{i,j})^k-h_t^{\overline{m}^k}(x)\right)\right]\right| \\
&\le\beta,
\end{aligned}$$

where $A^G_{i,j}$ is the true conditional mean for $G(\overline{\mu},i,\overline{m}^k,j)$:

$$\begin{aligned}
A^G_{i,j}&=\mathop{\mathbb{E}}_{(x,y),h^{\text{mean, moment}}}\left[\mathbb{1}\left[h^{\text{mean, moment}}(x)\in B(i,j),G(x)=1\right]\cdot y\right] \\
&=\frac{1}{T}\sum_{t=1}^{T}\mathop{\mathbb{E}}_{(x,y),h_t}\left[\mathbb{1}\left[h_t(x)\in B(i,j),G(x)=1\right]\cdot y\right]
\end{aligned}$$

As for mean consistency, the same approach works as in the proof of Theorem A.1.

Lemma A.1. *With probability $1-\lambda$ over the randomness of π_T, $\{(\overline{\mu}_t,\overline{m}^k_t)\}$, Algorithm 10 produces $\{h_t\}_{t=1}^T$ such that for every $i\in[n]$, $j\in[n']$, and $G\in\mathcal{G}$*

$$\frac{1}{T}\left|\sum_{t=1}^{T}\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}\left[h_t(x)\in B(i,j),G(x)=1\right]\cdot\left(y-h_t^{\overline{\mu}}(x)\right)\right]\right|\le(4+\epsilon)\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right)}$$

Proof. Fix $i\in[n],j\in[n']$ and $G\in\mathcal{G}$, and consider the following martingale sequence adapted to the filtration $\mathcal{F}_s=\sigma(\{(x_t,y_t),h_t\}_{t=1}^s)$:

$$\tilde{Z}_s=Z_{s-1}+\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}\left[h_s(x)\in B(i,j),G(x)=1\right]\cdot\left(y-h_s^{\overline{\mu}}(x)\right)\right]-\mathbb{1}\left[\overline{\mu}_s\in B(i,j),G(x_s)=1\right]\cdot(y_s-\overline{\mu}_s).$$

Applying Azuma's inequality (Lemma D.2) gives us that with probability $1-\lambda/2$ over the randomness of drawing D from $\mathcal{D}$ and π_T,

$$\begin{aligned}
&\frac{1}{T}\left|\sum_{t=1}^{T}\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[\mathbb{1}\left[h_t(x)\in B(i,j),G(x)=1\right]\cdot\left(y-h_t^{\overline{\mu}}(x)\right)\right]\right| \\
&\le\frac{1}{T}\left|\sum_{t=1}^{T}\mathbb{1}\left[(\overline{\mu}_t,\overline{m}^k_t)\in B(i,j),G(x_t)=1\right]\cdot(y_t-\overline{\mu}_t)\right|+\sqrt{\frac{8\ln\left(\frac{4}{\lambda}\right)}{T}}
\end{aligned}$$

Now, applying Theorem 4.3 with failure probability $\frac{\lambda}{2}$ and union bounding the above azuma's inequality

over every $i \in [n]$, $j \in [n']$ and $G \in \mathcal{G}$ gives us the result: we have that with probability $1-\lambda$ over π_T and $\mathcal{D}$,

$$\begin{aligned}
&\frac{1}{T}\left|\sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}\left[h_t(x) \in B(i,j), G(x)=1\right]\cdot\left(y - h_t^{\overline{\mu}}(x)\right)\right]\right| \\
&\leq (4+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{8|\mathcal{G}|n\cdot n'}{\lambda}\right)+2\epsilon} + \sqrt{\frac{8\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right)}{T}} \\
&\leq (6+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{8|\mathcal{G}|n\cdot n'}{\lambda}\right)+2\epsilon}
\end{aligned}$$

for every $i \in [n]$, $j \in [n']$ and $G \in \mathcal{G}$. □

As for the moment consistency, due to higher moments' non-linearity, we need an additional application of Azuma's inequality to show that the empirical conditional mean and the true conditional mean, denoted as A above, must be similar. This is to handle the fact that the empirical moment is centered around the empirical mean but the true moment is centered around the true mean.

For convenience, we write

$$A'^{G}_{i,j} = \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\left[(\overline{\mu}_t, \overline{m}_t^k) \in B(i,j), G(x_t)=1\right]\cdot y_t$$

to denote the empirical conditional mean.

Lemma A.2. *Fix $i \in [n]$, $j \in [n']$, and $G \in \mathcal{G}$. With probability $1-\lambda$ over the randomness of drawing D from $\mathcal{D}$ and π_T, we have*

$$|A^G_{i,j} - A'^{G}_{i,j}| \leq \sqrt{\frac{2\ln\left(\frac{2}{\lambda}\right)}{T}}$$

Proof. Consider the following martingale sequence once again adapted to the filtration $\mathcal{F}_s = \sigma(\{(x_t,y_t)\}_{t=1}^s)$:

$$\tilde{Z}_s = Z_{s-1} + \mathbb{1}[(\overline{\mu}_s, \overline{m}_s^k) \in B(i,j), G(x_s)=1]\cdot y_s - \mathop{\mathbb{E}}_{(x,y),h_s}[\mathbb{1}[h_s(x)\in B(i,j), G(x)=1]\cdot y].$$

Applying Azuma's inequality (Lemma D.2) to the above martingale gives us the result. □

Finally, we show that the true and empirical conditional moments when centered around $A^G_{i,j}$ must be close through Azuma's inequality.

Lemma A.3. *Fix $i \in [n]$, $j \in [n']$, and $G \in \mathcal{G}$. With probability $1-\lambda$ over the randomness of drawing D from $\mathcal{D}$ and π_T, we have*

$$\begin{aligned}
&\left|\frac{1}{T}\sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}[h_t(x)\in B(i,j), G(x)=1]\cdot\left((y-A^G_{i,j})^k - h_t^{\overline{m}^k}(x)\right)\right]\right. \\
&\left. - \frac{1}{T}\sum_{t=1}^{T}\mathbb{1}[(\overline{\mu}_t,\overline{m}_t^k)\in B(i,j), G(x_t)=1]\cdot\left((y_t-A^G_{i,j})^k - \overline{m}_t^k\right)\right| \\
&\leq \sqrt{\frac{8\ln\left(\frac{2}{\lambda}\right)}{T}}
\end{aligned}$$

Proof. Consider the following martingale sequence adapted to the filtration $\mathcal{F}_s = \sigma(\{(x_t, y_t), h_t\}_{t=1}^s)$:

$$\begin{aligned}\tilde{Z}_s &= Z_{s-1} \\ &+ \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_s}\left[\mathbb{1}\left[h_s(x) \in B(i,j), G(x) = 1\right] \cdot \left((y-A)^k - h_s^{\overline{m}^k}(x)\right)\right] \\ &- \mathbb{1}\left[(\overline{\mu}_s, \overline{m}_s^k) \in B(i,j), G(x_s) = 1\right] \cdot \left((y_s - A)^k - \overline{m}_s^k\right).\end{aligned}$$

Applying Azuma's to the above martingale gives us the result. ☐

Note that because $\{\overline{\mu}_t, \overline{m}_t^k\}_{t=1}^T$ is (α, β, n, n')-mean-conditioned-moment mutlticalibrated with respect to $\mathcal{G}$, we have

$$\left|\frac{1}{T}\sum_{t=1}^T \mathbb{1}[(\overline{\mu}_t, \overline{m}_t^k) \in B(i,j), G(x_t) = 1] \cdot \left((y_t - {A'}_{i,j}^G)^k - \overline{m}_t^k\right)\right| \leq \beta.$$

Therefore, by union bounding over every $i \in [n]$, $j \in [n']$ and $G \in [\mathcal{G}]$, we can show with probability $1-\lambda$ that for every i, j, and G

$$\begin{aligned}&\frac{1}{T}\left|\sum_{t=1}^T \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t}\left[\mathbb{1}[h_t(x) \in B(i,j), G(x) = 1] \cdot \left((y - A_{i,j}^G)^k - h_t^{\overline{m}^k}(x)\right)\right]\right| \\ &\leq \frac{1}{T}\left|\sum_{t=1}^T \mathbb{1}[(\overline{\mu}_t, \overline{m}_t^k) \in B(i,j), G(x_t) = 1] \cdot \left((y_t - A_{i,j}^G)^k - \overline{m}_t^k\right)\right| + \sqrt{\frac{8\ln\left(\frac{8|\mathcal{G}|nn'}{\lambda}\right)}{T}} \\ &\qquad\qquad \text{(with probability } 1-\lambda/4 \text{ Lemma A.3)} \\ &\leq \frac{1}{T}\left|\sum_{t=1}^T \mathbb{1}[(\overline{\mu}_t, \overline{m}_t^k) \in B(i,j), G(x_t) = 1] \cdot \left((y_t - {A'}_{i,j}^G)^k - \overline{m}_t^k\right)\right| + k\sqrt{\frac{2\ln\left(\frac{8|\mathcal{G}|nn'}{\lambda}\right)}{T}} + \sqrt{\frac{8\ln\left(\frac{8|\mathcal{G}|nn'}{\lambda}\right)}{T}} \\ &\qquad\qquad \text{(with probability } 1-\lambda/4 \text{ Lemma A.2)} \\ &\leq (k+2)\left((4+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{8|\mathcal{G}|n\cdot n'}{\lambda}\right) + 2\epsilon}\right) + \frac{k}{2n} + k\sqrt{\frac{2\ln\left(\frac{8|\mathcal{G}|nn'}{\delta}\right)}{T}} + \sqrt{\frac{8\ln\left(\frac{8|\mathcal{G}|nn'}{\delta}\right)}{T}} \\ &\qquad\qquad \text{(with probability } 1-\lambda/2 \text{ Theorem 4.3)} \\ &\leq (k+3)\left((5+\epsilon')\sqrt{\frac{2}{T}\ln\left(\frac{8|\mathcal{G}|n\cdot n'}{\lambda}\right) + 2\epsilon}\right) + \frac{k}{2n},\end{aligned}$$

where the second inequality holds because $|a^k - b^k| \leq k|a-b|$ for any $a, b \in [0,1]$ and $T > 2\ln(\frac{8|\mathcal{G}|n\cdot n'}{\lambda})$.

Because the mean consistency holds with probability $1-\lambda$ and the moment consistency holds with probability $1-\lambda$, $h^{\text{mean, moment}}$ is (α, β, n, n')-mean-conditioned-moment multicalibrated with respect to $\mathcal{G}$ over $\mathcal{D}$ with probability $1-2\lambda$. ☐

A.2.3 Interval Prediction

Algorithm 11: Von Neumann's Batch Multivalid Predictor

INPUT: Training dataset $D = \{(x_t, y_t)\}_{t=1}^{T}$
Training: Run Algorithm 6 on the sequence of examples D to generate a transcript π_T.
Denote by $h_t(x)$ the (randomized) mapping from $\mathcal{X}$ to $[0,1] \times [0,1]$ that Algorithm 4 induces as a function of transcript π_{t-1} (the prefix of π_T of length $t-1$).
Prediction: On input x, sample $h^{\text{interval}}(x)$ by selecting $t \sim [T]$ uniformly at random, and then sampling from $h_t(x)$.

More explicitly, select $t \sim [T]$ uniformly at random and:
Observe x_t and compute $C_{t-1}^{\ell,u}(x_t)$ for each $(\ell, u) \in \mathcal{P}^{rn}_{\text{interval}}$ as in (11) conditioning on π_{t-1}.
Solve the Linear Program from Figure 2 using the Ellipsoid algorithm, with Algorithm 7 as a separation oracle, to obtain a solution $Q_t^L \in \Delta\mathcal{P}^{rn}_{\text{interval}}$.
Predict $h^{\text{interval}}(x) = (\ell, u)$ with probability $Q_t^L((\ell, u))$.

Theorem A.3. *Assume that $\mathcal{D}$ is a (ρ, rn)-smooth distribution. Let $D = \{(x_t, y_t)\}_{t=1}^{T}$ be a dataset drawn i.i.d. from $\mathcal{D}$. Let $\delta, \lambda > 0$. With probability $1 - \lambda$, Algorithm 11 produces a predictor $h^{interval}$ that is (α, n)-multivalid with respect to δ and $\mathcal{G}$ over $\mathcal{D}$ where*

$$\alpha = \rho + 6\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n^2}{\lambda}\right) + 2\epsilon}.$$

Proof. In order to show that h^{interval} is (α, n)-multivalid with respect to δ and $\mathcal{G}$ over $\mathcal{D}$, it is sufficient to show for all $G \in \mathcal{G}$ and $i \leq j \in [n]$

$$\left| \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h^{\text{interval}}} \left[\mathbb{1}[h^{\text{interval}}(x) \in B(i,j), G(x) = 1] \cdot \left(\text{Cover}(h^{\text{interval}}(x), x) - (1-\delta)\right)\right] \right| \leq \alpha.$$

We can calculate:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h^{\text{interval}}} \left[\mathbb{1}[h^{\text{interval}}(x) \in B(i,j), G(x) = 1] \cdot \left(\text{Cover}(h^{\text{interval}}(x), y) - (1-\delta)\right)\right] \\
&= \sum_{(x,y)} \sum_{t=1}^{T} \mathcal{D}[(x,y)] \cdot \Pr[h^{\text{interval}} = h_t] \cdot \Pr_{h_t}[h_t(x) \in B(i,j)] \cdot \mathbb{1}[G(x) = 1] \cdot (\text{Cover}(h_t(x), y) - (1-\delta)) \\
&= \frac{1}{T} \sum_{(x,y)} \sum_{t=1}^{T} \mathcal{D}[(x,y)] \cdot \Pr_{h_t}[h_t(x) \in B(i,j)] \cdot \mathbb{1}[G(x) = 1] \cdot (\text{Cover}(h_t(x), y) - (1-\delta)) \\
&= \frac{1}{T} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t} [\mathbb{1}[h_t(x) \in B(i,j), G(x) = 1] \cdot (\text{Cover}(h_t(x), y) - (1-\delta))]
\end{aligned}$$

Consider the following martingale sequence adapted to the filtration $\mathcal{F}_s = \sigma(\{(x_t, y_t), \overline{\mu}_t\}_{t=1}^{s})$:

$$\begin{aligned}
\tilde{Z}_s = Z_{s-1} &+ \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t} [\mathbb{1}[h_s(x) \in B(i,j), G(x) = 1] \cdot (\text{Cover}(h_s(x), y) - (1-\delta)) \,|\pi_{s-1}] \\
&- \mathbb{1}[(\overline{\ell}_s, \overline{u}_s) \in B(i,j), G(x_s) = 1] \cdot \left(\text{Cover}((\overline{\ell}_s, \overline{u}_s), y_s) - (1-\delta)\right).
\end{aligned}$$

Because $|Z_s - Z_{s-1}| \leq 2$, we can apply Azuma's inequality (Lemma D.2) to get that with probability $1 - \lambda/2$

over the randomness of π_T and D,

$$\left| \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t} [\mathbb{1}[h_t(x) \in B(i,j), G(x) = 1] \cdot (\text{Cover}(h_t(x), y) - (1-\delta))] \right.$$
$$\left. - \sum_{t=1}^{T} \mathbb{1}[(\overline{\ell}_t, \overline{u}_t) \in B(i,j), G(x_t) = 1] \cdot \left(\text{Cover}((\overline{\ell}_t, \overline{u}_t), y_t) - (1-\delta)\right) \right|$$
$$\leq 2\sqrt{2T \ln\left(\frac{4}{\lambda}\right)}.$$

Note that from Theorem 5.3 that with probability $1-\lambda/2$ over the randomness of π_T produced in training that for all $i \leq j \in [n], G \in \mathcal{G}$:

$$\left| \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}[(\overline{\ell}_t, \overline{u}_t) \in B(i,j), G(x_t) = 1] \cdot \left(\text{Cover}((\overline{\ell}_t, \overline{u}_t), y_t) - (1-\delta)\right) \right| \leq \rho + 4\sqrt{\frac{2}{T} \ln\left(\frac{4|\mathcal{G}|n^2}{\lambda}\right) + 2\epsilon}.$$

Therefore, taking the union bound for the above Azuma's inequality over all $i \leq j \in [n], G \in \mathcal{G}$, we have that with probability $1-\lambda$,

$$\left| \frac{1}{T} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D},h_t} [\mathbb{1}[h_t(x) \in B(i,j), G(x) = 1] \cdot (\text{Cover}(h_t(x), y) - (1-\delta))] \right| \leq \rho + 6\sqrt{\frac{2}{T} \ln\left(\frac{4|\mathcal{G}|n^2}{\lambda}\right) + 2\epsilon}$$

for every $i \leq j \in [n], G \in \mathcal{G}$.

□

B Unboundedly Many Groups, Bounded Group Membership

In this section, we briefly sketch how we can modify our results so that we can handle the case that there are a "large number" of groups (i.e. $|\mathcal{G}|$ is infinite or larger than 2^T — a range in which the bounds we prove in the main body are vacuous). In this scenario, we maintain the assumption that any given $x \in \mathcal{X}$ appears in at most d groups, i.e. that $|\mathcal{G}(x)| \leq d$ for all $x \in \mathcal{X}$. As we have already noted, in this scenario, our running time dependence on $|\mathcal{G}|$ can be replaced with d — here we show that we can do the same in our convergence bounds.

The first step is to redefine our surrogate loss function L. The way it was previously defined, L_0 was already a quantity at the scale of $|\mathcal{G}|$, and so it would be hopeless to use it for infinite collections of groups. But a small modification solves this problem:

Definition B.1 (Surrogate loss function)**.** *Fixing a transcript $\pi_s \in \Pi^*$ and a parameter $\eta \in [0, \frac{1}{2}]$, define a surrogate calibration loss function at day s as:*

$$L_s(\pi_s) = 1 + \sum_{\substack{G\in\mathcal{G},\\ i\in[n]}} \left(\exp(\eta V_s^{G,i}) + \exp(-\eta V_s^{G,i}) - 2\right).$$

When the transcript π_s is clear from context, we will sometimes simply write L_s.

Observe that this modified function satisfies $L_0 = 1$, independently of the size of $|\mathcal{G}|$, and still allows us to tightly upper bound our calibration loss:

Observation B.1. *For any transcript π_T, and any $\eta \in [0, \frac{1}{2}]$, we have that:*

$$\max_{G\in\mathcal{G},i\in[n]} \left|V_T^{G,i}\right| \leq \frac{1}{\eta} \ln(L_T + 2dT) \leq \max_{G\in\mathcal{G},i\in[n]} \left|V_T^{G,i}\right| + \frac{\ln(dT)}{\eta}.$$

This observation uses the fact that because (by assumption) $|\mathcal{G}(x_t)| \leq d$ for all t, after T time steps, there are at most dT quantities $V_T^{G,i}$ that are non-zero.

We can now provide a modified bound on $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1})$:

Lemma B.1. *For any transcript $\pi_s \in \Pi^*$, any $x_{s+1} \in \mathcal{X}$, and any $\overline{\mu}_{s+1} \in \mathcal{P}_{mean}$ such that $\overline{\mu}_{s+1} \in B(i)$ for some $i \in [n]$:*

$$\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1}) \leq \eta \left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}} [\tilde{y}_{s+1}] - \overline{\mu}_{s+1} \right) C_s^i(x_{s+1}) + 2\eta^2 L_s + 4d\eta^2,$$

where for each $i \in [n]$:

$$C_s^i(x_{s+1}) \equiv \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i}) - \exp(-\eta V_s^{G,i}).$$

Proof. Fix any transcript $\pi_s \in \Pi^*$ (which defines L_s), feature vector $x_{s+1} \in \mathcal{X}$, and $\overline{\mu}_{s+1}$ such that $\overline{\mu}_{s+1} \in B(i)$ for some $i \in [n]$. By direct calculation, we obtain:

$$\begin{aligned}
&\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu}_{s+1}) \\
&= \mathop{\mathbb{E}}_{\tilde{y}_{s+1}} \left[\sum_{G \in \mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i}) \left(\exp(\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1})) - 1 \right) + \exp(-\eta V_s^{G,i}) \left(\exp(-\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1})) - 1 \right) \right], \\
&\leq \mathop{\mathbb{E}}_{\tilde{y}_{s+1}} \left[\sum_{G \in \mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i}) \left(\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1}) + 2\eta^2 \right) + \exp(-\eta V_s^{G,i}) \left(-\eta(\tilde{y}_{s+1} - \overline{\mu}_{s+1}) + 2\eta^2 \right) \right], \\
&= \eta \left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}} [\tilde{y}_{s+1}] - \overline{\mu}_{s+1} \right) \sum_{G \in \mathcal{G}(x_{s+1})} \left(\exp(\eta V_s^{G,i}) - \exp(-\eta V_s^{G,i}) \right) + 2\eta^2 \sum_{G \in \mathcal{G}(x_{s+1})} \left(\exp(\eta V_s^{G,i}) + \exp(-\eta V_s^{G,i}) \right), \\
&\leq \eta \left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}} [\tilde{y}_{s+1}] - \overline{\mu}_{s+1} \right) \left(\sum_{G \in \mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,i}) - \exp(-\eta V_s^{G,i}) \right) + 2\eta^2 L_s + 4d\eta^2, \\
&= \eta \left(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}} [\tilde{y}_{s+1}] - \overline{\mu}_{s+1} \right) C_s^i(x_{s+1}) + 2\eta^2 L_s + 4d\eta^2.
\end{aligned}$$

Here, the first inequality follows from the fact that for $0 < |x| < \frac{1}{2}$, $\exp(x) \leq 1 + x + 2x^2$. ☐

We can use this to provide a modified bound to Lemma 3.2.

Lemma B.2. *For any transcript $\pi_s \in \Pi^*$, any $x_{s+1} \in \mathcal{X}$, and any $r \in \mathbb{N}$ there exists a distribution over predictions for the learner $Q_{s+1}^L \in \Delta\mathcal{P}^{rn}$, such that regardless of the adversary's choice of distribution of y_{s+1} over $\Delta\mathcal{Y}$, we have that:*

$$\mathop{\mathbb{E}}_{\overline{\mu} \sim Q_{s+1}^L} [\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu})] \leq L_s \left(\frac{\eta}{rn} + 2\eta^2 \right) + 2d.$$

Proof. As in the proof of Lemma 3.2, we construct a zero-sum game between the learner and the adversary. Fix the transcript π_s and the feature vector x_{s+1}. We define the utility of this game to be the upper bound we proved on $\Delta_{s+1}(\pi_s, x_{s+1}, \overline{\mu})$ in Lemma B.1. For each $\overline{\mu} \in \mathcal{P}^{rn}$ and each $y \in [0, 1]$, we let:

$$u(\overline{\mu}, y) = \eta \, (y - \overline{\mu}) \, C_s^{\overline{\mu}}(x_{s+1}) + 2\eta^2 L_s + 4d\eta^2.$$

We now establish the value of this game. Observe that for any strategy of the adversary (which fixes $\mathbb{E}[\tilde{y}]$), the learner can respond by playing $\overline{\mu}^* = \text{argmin}_{\overline{\mu} \in \mathcal{P}^{rn}} |\, \mathbb{E}[\tilde{y}] - \overline{\mu}|$, and that because of our discretization,

$\min |\mathbb{E}[\tilde{y}] - \overline{\mu}^*| \leq \frac{1}{rn}$. Therefore, the value of the game is at most:

$$\begin{aligned}\max_{y\in[0,1]} \min_{\overline{\mu}^* \in \mathcal{P}^{rn}} u(\overline{\mu}^*, y) &\leq \max_{\overline{\mu}\in\mathcal{P}^{rn}} \frac{\eta}{rn} \left|C_s^{\overline{\mu}}(x_{s+1})\right| + 2\eta^2 L_s + 4d\eta^2, \\ &\leq L_s\left(\frac{\eta}{rn} + 2\eta^2\right) + 2d.\end{aligned}$$

Here the latter inequality follows since $C_s^{\overline{\mu}}(x_{s+1}) \leq L_s + 2d$ for all $\overline{\mu} \in \mathcal{P}^{rn}$, by observation, and then since $\eta \in (0, \frac{1}{2})$ we have the bound. We can now apply the minimax theorem (Theorem 2.1) to conclude that there exists a fixed distribution $Q_{s+1}^L \in \mathcal{Q}^L$ for the learner that guarantees that simultaneously for *every* label $y \in [0,1]$ that might be chosen by the adversary:

$$\mathop{\mathbb{E}}_{\overline{\mu}\sim Q_{s+1}^L}[u(\overline{\mu}, y)] \leq L_s\left(\frac{\eta}{rn} + 2\eta^2\right) + 2d,$$

as desired. □

Corollary B.1. *For every $r \in \mathbb{N}$, $s \in [T]$, $\pi_s \in \Pi^*$, and $x_{s+1} \in \mathcal{X}$ (which fixes L_s and Q_{s+1}^L), and any distribution over $\mathcal{Y}$:*

$$\mathop{\mathbb{E}}_{\overline{\mu}_{s+1}^L \sim Q_{s+1}}[\tilde{L}_{s+1}|\pi_s] = L_s + \mathop{\mathbb{E}}_{\overline{\mu}_{s+1}\sim Q_{s+1}^L}[\Delta_{s+1}(\pi_{s+1}, x_{s+1}, \overline{\mu}_{s+1})] \leq L_s\left(1 + \frac{\eta}{rn} + 2\eta^2\right) + 2d.$$

Lemma B.2 shows that playing the minimax strategy of this zero-sum game (Algorithm 1) continues to provide a low value to the learner. We now show the counterpart of the first part of Theorem 3.1 for these modified bounds:

Theorem B.1. *Consider a nonnegative random process $\tilde{X}_t$ adapted to the filtration $\mathcal{F}_t = \sigma(\pi_t)$, where $\tilde{X}_0$ is constant a.s. Suppose we have that for any period t, and any π_{t-1}, $\mathbb{E}[\tilde{X}_t|\pi_{t-1}] \leq X_{t-1}(1 + \eta c + 2\eta^2) + 2d$ for some $\eta \in [0, \frac{1}{2}], c \in [0,1], d > 0$. Then we have that:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{X}_T] \leq (X_0 + 2dT)\exp\left(T\eta c + 2T\eta^2\right). \tag{13}$$

Proof. First, observe that:

$$\begin{aligned}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{X}_T] &= \mathop{\mathbb{E}}_{\tilde{\pi}_{T-1}}\left[\mathbb{E}[\tilde{X}_T|\pi_{T-1}]\right], \\ &\leq \mathop{\mathbb{E}}_{\tilde{\pi}_{T-1}}\left[\mathbb{E}[\left(1 + \eta c + 2\eta^2\right) X_{T-1} + 2d|\pi_{T-1}]\right] \\ &= \left(1 + \eta c + 2\eta^2\right) \mathop{\mathbb{E}}_{\tilde{\pi}_{T-1}}\left[\tilde{X}_{T-1}\right] + 2d, \\ &\vdots \\ &\leq X_0\left(1 + \eta c + 2\eta^2\right)^T + 2d\sum_{t=0}^{T-1}(1 + c\eta + 2\eta^2)^t, \\ &\leq X_0\left(1 + \eta c + 2\eta^2\right)^T + 2dT(1 + c\eta + 2\eta^2)^T, \\ &= (X_0 + 2dT)\exp\left(T\ln\left(1 + \eta c + 2\eta^2\right)\right), \\ &\leq (X_0 + 2dT)\exp\left(T\eta c + 2T\eta^2\right),\end{aligned}$$

where the last inequality holds because $\ln(1 + x) \leq x$ for any $x > -1$. This concludes the proof of (13). □

We are now ready to bound our multicalibration error. As a straightforward consequence of Corollary B.1 and Theorem B.1, we have the following Corollary.

Corollary B.2. *Against any adversary, Algorithm 1 instantiated with discretization parameter r results in surrogate loss satisfying:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T] \leq (1+2dT)\exp\left(\frac{T\eta}{rn}+2T\eta^2\right).$$

Proof. Note that the first part of Theorem B.1 applies to the process L with $L_0 = 1$ and $c = \frac{1}{rn}$. The bound follows by plugging these values into (13). □

Next, we can convert this into a bound on Algorithm 1's expected calibration error:

Theorem B.2. *When Algorithm 1 is run using n buckets for calibration, discretization $r \in \mathbb{N}$, and $\eta = \sqrt{\frac{\ln(1+2dT)}{2T}}$, then against any adversary, its sequence of mean predictions are (α, n)-multicalibrated with respect to $\mathcal{G}$, where:*

$$\mathbb{E}[\alpha] \leq \frac{1}{rn} + 2\sqrt{\frac{2\ln(1+4dT)}{T}}.$$

For $r = \frac{\sqrt{T}}{\epsilon n\sqrt{2\ln(1+4dT)}}$ this gives:

$$\mathbb{E}[\alpha] \leq (2+\epsilon)\sqrt{\frac{2}{T}\ln(1+4dT)}.$$

Here the expectation is taken over the randomness of the transcript π_T.

Proof. From Observation 3.1, it suffices to show that

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G\in\mathcal{G}, i\in[n]}|\tilde{V}_T^{G,i}|\right] \leq \frac{1}{rn} + 2\sqrt{\frac{2\ln(1+4dT)}{T}}.$$

We begin by computing a bound on the (exponential of) the expectation of this quantity:

$$\begin{aligned}
\exp\left(\eta \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}|\tilde{V}_T^{G,i}|\right]\right) &\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\exp\left(\eta\max_{G,i}|\tilde{V}_T^{G,i}|\right)\right], \\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}\exp\left(\eta|\tilde{V}_T^{G,i}|\right)\right], \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}\left(\exp\left(\eta\tilde{V}_T^{G,i}\right)+\exp\left(-\eta\tilde{V}_T^{G,i}\right)\right)\right], \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\sum_{\substack{G,i \\ G_T(i)\neq\phi}}\left(\exp\left(\eta\tilde{V}_T^{G,i}\right)+\exp\left(-\eta\tilde{V}_T^{G,i}\right)\right)\right], \\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T + 2dT], \\
&\leq (1+2dT)\exp\left(\frac{T\eta}{rn}+2T\eta^2\right)+2dT, \\
&\leq (1+4dT)\exp\left(\frac{T\eta}{rn}+2T\eta^2\right).
\end{aligned}$$

Here the first step is by Jensen's inequality and the second last one follows from Corollary B.2. Taking the logarithm of both sides and dividing by ηT, we have

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}|\tilde{V}_T^{G,i}|\right] \leq \frac{\ln(1+4dT)}{\eta T} + \frac{1}{rn} + 2\eta.$$

Choosing $\eta = \sqrt{\frac{\ln(1+4dT)}{2T}}$, we thus obtain the desired inequality

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i}|\tilde{V}_T^{G,i}|\right] \leq \frac{1}{rn} + 2\sqrt{\frac{2\ln(1+4dT)}{T}}.$$

□

The corresponding high-probability bounds are omitted for brevity. They have the analogous dependence on dT replacing $|\mathcal{G}|$. Similar bounds can be obtained for the case of moment-multicalibration and multivalid intervals with the same approach.

C Mean Conditioned Moment Multicalibrators Can Randomize Over Small Support

In Section 4.3, we derived a linear programming based algorithm for making mean conditioned moment multicalibrated predictors. Although we proved that we could reduce the pure strategy space of the learner from (r^2nn') to $4nn'$, a priori, the solutions we find via linear programming could have full support. Here we prove that this need not be the case — there always exists a basic feasible solution of the linear program that we solve that has support only over $k+1$ pure strategies for the learner.

Lemma C.1. *For any game with objective function (7), there exists a minimax strategy for the learner $\hat{Q}^L \in \hat{\mathcal{Q}}^L_{r,n,n'}$, such that $|support(\hat{Q}^L)| \leq k+1$.*

Proof. Suppose that Q^* is a minimax strategy for the learner.

Observe that the adversary's best response in this problem is straightforward: we have that $\psi_\ell = 1$ if $\sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q^*(\overline{\mu},\overline{m}^k) > 0$, that $\psi_\ell = 0$ if $\sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q^*(\overline{\mu},\overline{m}^k) < 0$, and otherwise the adversary is indifferent. Define

$$L_+ = \{\ell \in [k] : \sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q^*(\overline{\mu},\overline{m}^k) > 0\},$$
$$L_- = \{\ell \in [k] : \sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q^*(\overline{\mu},\overline{m}^k) < 0\},$$
$$L_= = \{\ell \in [k] : \sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q^*(\overline{\mu},\overline{m}^k) = 0\}.$$

Note that $L_+ \cup L_- \cup L_= = [k]$.

Since Q^* is a minimax strategy, it must solve the following linear program, which corresponds to minimizing the learner's objective value over all strategies Q which engender the same best response for the

adversary as Q^*:

$$\min_{Q \in \hat{\mathcal{Q}}^L_{r,n,n'}} \sum_{\overline{\mu},\overline{m}^k} Q(\overline{\mu},\overline{m}^k)\left(\overline{\mu} C_s^{\overline{\mu},\overline{m}^k} + \overline{m}^k D_s^{\overline{\mu},\overline{m}^k} - \hat{\mu}_i^k D_s^{\overline{\mu},\overline{m}^k}\right)$$

subject to:

$$\forall \ell \in L_+ : \sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q(\overline{\mu},\overline{m}^k) \geq 0,$$

$$\forall \ell \in L_- : \sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q(\overline{\mu},\overline{m}^k) \leq 0,$$

$$\forall \ell \in L_= : \sum_{\overline{\mu},\overline{m}^k} F_\ell^{\overline{\mu},\overline{m}^k} Q(\overline{\mu},\overline{m}^k) = 0,$$

$$\sum_{\overline{\mu},\overline{m}^k} Q(\overline{\mu},\overline{m}^k) = 1,$$

$$Q \geq 0.$$

Further, any solution to this LP must also be a minimax strategy for the learner. Observe that this has $k+1$ linear constraints. Any such linear program has a basic feasible solution: so there exists a solution $\hat{Q}^L$ (viewed as a vector) with exactly the number of non-zero entries as the number of binding constraints, i.e. $\leq k+1$, as desired.[8] This is exactly the statement of the Lemma. □

D Proofs from Section 3

Theorem 3.1. *Consider a nonnegative random process $\tilde{X}_t$ adapted to the filtration $\mathcal{F}_t = \sigma(\pi_t)$, where $\tilde{X}_0$ is constant a.s. Suppose we have that for any period t, and any π_{t-1}, $\mathbb{E}[\tilde{X}_t|\pi_{t-1}] \leq X_{t-1}(1+\eta c+2\eta^2)$ for some $\eta \in [0,\frac{1}{2}], c \in [0,1]$. Then we have that:*

$$\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{X}_T] \leq X_0 \exp\left(T\eta c + 2T\eta^2\right). \tag{3}$$

Further, define a process $\tilde{Z}_t$ adapted to the same filtration by $\tilde{Z}_t = Z_{t-1} + \ln \tilde{X}_t - \mathbb{E}[\ln(\tilde{X}_t)|\pi_{t-1}]$. Suppose that $|Z_t - Z_{t-1}| \leq 2\eta$, where $Z_0 = 0$ a.s. Then, with probability $1-\lambda$,

$$\ln(X_T(\pi_T)) \leq \ln(X_0) + T\left(\eta c + 2\eta^2\right) + \eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}. \tag{4}$$

[8] As an aside, we point out that this also implies the square submatrix with rows corresponding to binding constraints and corresponding to non-zero variables is of full rank. Textbook treatments that we are aware of consider either LPs with all inequality constraints or all equality constraints. So for completeness we include the following argument. Convert the LP above into a LP in standard form $\min c^T x$ s.t. $Ax = b, x \geq 0$ by adding/subtracting non-negative slack variables to the inequality constraints L_+, L_-. This is a system of $k+1$ linear equality constraints in $4nn' + |L_-| + |L_+| + 1$ variables. We know that there exists an optimal of this LP that is a Basic feasible solution (BFS) (see e.g. Theorem 4.7 of Vohra [2004]), i.e. an optimal solution with exactly $k+1$ non-zero variable with the corresponding $(k+1)\times(k+1)$ sub-matrix of A, denoted $\hat{A}$, of full rank. By observation, the number of non-zero Q's in this BFS must equal the number of constraints that bind at equality in the original LP (any non-zero slack variable will correspond to a slack constraint in the original). The sub-matrix of $\bar{A}$ corresponding to the non-zero Q's as columns and binding constraints of the original LP as rows must be of full rank, because these rows have all 0's in the columns corresponding to the slack variables in $\bar{A}$.

Proof. First, observe that:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{X}_T] &= \mathop{\mathbb{E}}_{\tilde{\pi}_{T-1}}\left[\mathbb{E}[\tilde{X}_T|\pi_{T-1}]\right], \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_{T-1}}\left[\mathbb{E}[\left(1+\eta c+2\eta^2\right)X_{T-1}|\pi_{T-1}]\right] \\
&= \left(1+\eta c+2\eta^2\right)\mathop{\mathbb{E}}_{\tilde{\pi}_{T-1}}\left[\tilde{X}_{T-1}\right], \\
&\quad\vdots \\
&\leq X_0\left(1+\eta c+2\eta^2\right)^T, \\
&= X_0\exp\left(T\ln\left(1+\eta c+2\eta^2\right)\right), \\
&\leq X_0\exp\left(T\eta c+2T\eta^2\right),
\end{aligned}$$

where the last inequality holds because $\ln(1+x)\leq x$ for any $x>-1$. This concludes the proof of (3).

Towards demonstrating the high-probability bound 4, we first show the following statement.

Lemma D.1. *For any π_T, we have*

$$\sum_{t=1}^{T}\left(\mathop{\mathbb{E}}_{\tilde{\pi}_t}\left[\ln(\tilde{X}_t)\middle|\pi_{t-1}\right]-\ln(X_{t-1}(\pi_{t-1}))\right)\leq T\left(\eta c+2\eta^2\right).$$

Proof. Fixing π_T and taking any $t\leq T$, we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{\tilde{\pi}_t}\left[\ln(\tilde{X}_t)|\pi_{t-1}\right] &\leq \ln\left(\mathop{\mathbb{E}}_{\tilde{\pi}_t}[\tilde{X}_t|\pi_{t-1}]\right), && \text{(Jensen's inequality)} \\
&\leq \ln(X_{t-1}(\pi_{t-1}))+\ln\left(1+c\eta+2\eta^2\right), && \text{(by assumption)} \\
&\leq \ln(X_{t-1}(\pi_{t-1}))+\left(c\eta+2\eta^2\right). && (\ln(1+x)\leq x \text{ for any } x>-1)
\end{aligned}$$

Summing over every round $t\in[T]$ gives us the result. □

Now observe that for any π_{t-1}, we have $\mathbb{E}[\tilde{Z}_t|\pi_{t-1}]=Z_{t-1}$, so the process $\tilde{Z}_t$ is a martingale. Further, its increments are bounded by assumption. Recall Azuma's inequality for martingales with bounded increments (see e.g. Dubhashi and Panconesi [2009]):

Lemma D.2 (Azuma's Inequality)**.** *For any martingale $\{\tilde{Z}_t\}_{t=1}^T$ with $|Z_t-Z_{t-1}|\leq c$ a.s., for all T it holds*

$$\Pr\left[\tilde{Z}_T-\tilde{Z}_0\geq\epsilon\right]\leq\exp\left(-\frac{\epsilon^2}{2c^2T}\right).$$

By assumption, we may apply Azuma's inequality with $c=2\eta$, and we obtain

$$\Pr_{\tilde{\pi}_T}\left[\sum_{t=1}^{T}\left(\ln(X_t(\pi_t))-\mathop{\mathbb{E}}_{\tilde{\pi}_t}[\ln X_t(\tilde{\pi}_t)|\pi_{t-1}]\right)\geq\epsilon\right]\leq\exp\left(-\frac{\epsilon^2}{8\eta^2T}\right).$$

So, with probability $1-\lambda$, it holds that

$$\begin{aligned}
&\sum_{t=1}^{T}\left(\ln(X_t(\pi_t))-\mathop{\mathbb{E}}_{\tilde{\pi}_t}[\ln X_t(\tilde{\pi}_t)|\pi_{t-1}]\right)\leq\eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)} \\
\implies &\ln(X_T(\pi_T))\leq\ln(X_0)+\left(\sum_{t=1}^{T}\mathop{\mathbb{E}}_{\tilde{\pi}_t}\left[\ln(X_t(\tilde{\pi}_t))|\pi_{t-1}\right]-\ln(X_{t-1}(\pi_{t-1}))\right)+\eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)} \\
\implies &\ln(X_T(\pi_T))\leq\ln(X_0)+T\left(\eta c+2\eta^2\right)+\eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)},
\end{aligned}$$

where the last inequality follows from Lemma D.1. □

Lemma 3.3. *At any round $t \in [T]$ and for any realized transcript π_t, $|Z_t - Z_{t-1}| \leq 2\eta$.*

Proof. Observe that

$$\begin{aligned}|Z_t - Z_{t-1}| &= |\ln(L_t(\pi_t)) - \mathbb{E}\left[\ln(L_t(\tilde{\pi}_t))|\pi_{t-1}\right]| \\ &= \left|\mathbb{E}\left[\ln\left(\frac{L_t(\pi_t)}{L_t(\tilde{\pi}_t)}\right)\middle|\pi_{t-1}\right]\right|\end{aligned}$$

Note that for any π_t,

$$L_t(\pi_t) = L_{t-1}(\pi_{t-1}) + \Delta_t(\pi_{t-1}, x_t, y_t, \overline{\mu}_t)$$

where:

$$\Delta_t(\pi_{t-1}, x_t, y_t, \overline{\mu}_t) = \sum_{\mathcal{G}(x_t)} \exp(\eta V_{t-1}^{G,B^{-1}(\overline{\mu}_t)}) \left(\exp(\eta(y_t - \overline{\mu}_t)) - 1\right) + \exp(-\eta V_{t-1}^{G,B^{-1}(\overline{\mu}_t)}) \left(\exp(-\eta(y_t - \overline{\mu}_t)) - 1\right).$$

Since $y_t - \overline{\mu}_t$ must lie in $[-1, 1]$, we have that:

$$(\exp(-\eta) - 1)L_{t-1}(\pi_{t-1}) \leq \Delta_t(\pi_{t-1}, x_t, y_t, \overline{\mu}_t) \leq (\exp(\eta) - 1)L_{t-1}(\pi_{t-1})$$

which implies:

$$\exp(-\eta)L_{t-1}(\pi_{t-1}) \leq L_t(\pi_t) \leq \exp(\eta)L_{t-1}(\pi_{t-1}).$$

Hence, for any two transcripts π_t, π_t' which are equal over the first $t-1$ periods, we have

$$\left|\ln\left(\frac{L_t(\pi_t)}{L_t(\pi_t')}\right)\right| \leq \ln\left(\frac{\exp(\eta)}{\exp(-\eta)}\right) = 2\eta.$$

Therefore, $\left|\mathbb{E}\left[\ln\left(\frac{L_t(\pi_t)}{L_t(\tilde{\pi}_t)}\right)\middle|\pi_{t-1}\right]\right| \leq 2\eta$ as desired. □

E Proofs from Section 4

Theorem 4.1. *When Algorithm 3 is run using bucketing coarseness parameters n and n', discretization parameter $r \in \mathbb{N}$, and $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n \cdot n')}{2T}} \in (0, 1/2)$, then against any adversary, its sequence of mean-moment predictions is (α, β, n, n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$, where $\beta = (k+1)\alpha + \frac{k}{2n}$ and:*

$$\mathbb{E}[\alpha] \leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n \cdot n')}{T}}.$$

For $r = \frac{\sqrt{T}(n+n')}{\varepsilon n \cdot n' \cdot \sqrt{2\ln(4|\mathcal{G}|n \cdot n')}}$, this gives:

$$\mathbb{E}[\alpha] \leq (2 + \varepsilon)\sqrt{\frac{2}{T}\ln(4|\mathcal{G}|n \cdot n')}.$$

Here the expectation is taken over the randomness of the transcript π_T.

Proof. From Observation 4.1, it suffices to show that:

$$\begin{aligned}\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G \in \mathcal{G}, i \in [n], j \in [n']} |\tilde{V}_T^{G,i,j}|\right] &\leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n \cdot n')}{T}}, \\ \frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G \in \mathcal{G}, i \in [n], j \in [n']} |\tilde{M}_T^{G,i,j}|\right] &\leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n \cdot n')}{T}}.\end{aligned}$$

We begin by computing a bound on the (exponential of) the expectation of the first quantity:

$$\begin{aligned}
\exp\left(\eta \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,i,j}|\tilde{V}_T^{G,i,j}|]\right) &\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\exp\left(\eta \max_{G,i,j}|\tilde{V}_T^{G,i,j}|\right)\right], \\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i,j}\exp\left(\eta|\tilde{V}_T^{G,i,j}|\right)\right], \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,i,j}\left(\exp\left(\eta\tilde{V}_T^{G,i,j}\right)+\exp\left(-\eta\tilde{V}_T^{G,i,j}\right)\right)\right], \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\sum_{G,i,j}\left(\exp\left(\eta\tilde{V}_T^{G,i,j}\right)+\exp\left(-\eta\tilde{V}_T^{G,i,j}\right)+\exp\left(\eta\tilde{M}_T^{G,i,j}\right)+\exp\left(-\eta\tilde{M}_T^{G,i,j}\right)\right)\right], \\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T], \\
&\leq 4|\mathcal{G}|n\cdot n'\cdot\exp\left(\frac{T\eta}{rn}+\frac{T\eta}{rn'}+2T\eta^2\right).
\end{aligned}$$

Here the first inequality follows from Jensen's inequality and the last one follows from Corollary 4.2. Taking the log of both sides and dividing by ηT we obtain

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,i}|\tilde{V}_T^{G,i}|] \leq \frac{\ln(4|\mathcal{G}|n\cdot n')}{\eta T}+\frac{1}{rn}+\frac{1}{rn'}+2\eta.$$

Choosing $\eta=\sqrt{\frac{\ln(4|\mathcal{G}|n\cdot n')}{2T}}$, we have

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,i}|\tilde{V}_T^{G,i}|] \leq \frac{1}{rn}+\frac{1}{rn'}+2\sqrt{\frac{2\ln(4|\mathcal{G}|n\cdot n')}{T}}.$$

Repeating the same steps, we get an identical bound for $\frac{1}{T}\mathbb{E}_{\tilde{\pi}_T}[\max_{G\in\mathcal{G},i\in[n],j\in[n']}|\tilde{M}_T^{G,i,j}|]$. □

Now, given $\tilde{L}$, define $\tilde{Z}$ analogously to the second part of Theorem 3.1. Next, we can show that the increments of $\tilde{Z}$ thus defined, at any round t, can be bounded.

Lemma E.1. *At any round $t\in[T]$ and for any realized transcript π_t, $|Z_t-Z_{t-1}|\leq 2\eta$.*

Proof. Observe that

$$\begin{aligned}
|Z_t-Z_{t-1}| &= |\ln(L_t(\pi_t))-\mathbb{E}\left[\ln(L_t(\tilde{\pi}_t))|\pi_{t-1}\right]| \\
&= \left|\mathbb{E}\left[\ln\left(\frac{L_t(\pi_t)}{L_t(\tilde{\pi}_t)}\right)\middle|\pi_{t-1}\right]\right|
\end{aligned}$$

Note that for any π_t,

$$L_t(\pi_t)=L_{t-1}(\pi_{t-1})+\Delta_t(\pi_{t-1},x_t,y_t,\overline{\mu}_t,\overline{m}_t^k)$$

where:

$$\begin{aligned}
&\Delta_t(\pi_{t-1},x_t,y_t,\overline{\mu}_t,\overline{m}_t^k) \\
=\ &\sum_{\mathcal{G}(x_t)}\exp(\eta V_{t-1}^{G,B^{-1}(\overline{\mu}_t),B^{-1}(\overline{m}_t^k)})\left(\exp(\eta(y_t-\overline{\mu}_t))-1\right)+\exp(-\eta V_{t-1}^{G,B^{-1}(\overline{\mu}_t),B^{-1}(\overline{m}_t^k)})\left(\exp(-\eta(y_t-\overline{\mu}_t))-1\right), \\
&+\sum_{\mathcal{G}(x_t)}\exp(\eta M_{t-1}^{G,B^{-1}(\overline{\mu}_t),B^{-1}(\overline{m}_t^k)})\left(\exp(\eta((y_t-\hat{\mu}_{\overline{\mu}_t})^k-\overline{m}_t^k))-1\right) \\
&+\exp(-\eta M_{t-1}^{G,B^{-1}(\overline{\mu}_t),B^{-1}(\overline{m}_t^k)})\left(\exp(-\eta((y_t-\hat{\mu}_{\overline{\mu}_t})^k-\overline{m}_t^k))-1\right).
\end{aligned}$$

Since $(y_t - \overline{\mu}_t)$ and $((y_t - \hat{\mu}_{\overline{\mu}_t})^k - \overline{m}_t^k)$ must lie in $[-1, 1]$, we have that:

$$(\exp(-\eta) - 1)L_{t-1}(\pi_{t-1}) \leq \Delta_t(\pi_{t-1}, x_t, y_t, \overline{\mu}_t, \overline{m}_t^k) \leq (\exp(\eta) - 1)L_{t-1}(\pi_{t-1})$$

which implies:

$$\exp(-\eta)L_{t-1}(\pi_{t-1}) \leq L_t(\pi_t) \leq \exp(\eta)L_{t-1}(\pi_{t-1}).$$

Therefore, for any two π_t, π_t' such that the corresponding transcripts for the first $t-1$ periods is the same, we have

$$\left|\ln\left(\frac{L_t(\pi_t)}{L_t(\pi_t')}\right)\right| \leq \ln\left(\frac{\exp(\eta)}{\exp(-\eta)}\right) = 2\eta.$$

Therefore we have $\left|\mathbb{E}\left[\ln\left(\frac{L_t(\pi_t)}{L_t(\bar{\pi}_t)}\right)\middle|\pi_{t-1}\right]\right| \leq 2\eta$ as desired. □

Theorem 4.2. *When Algorithm 3 is run using bucketing coarseness parameters n and n', discretization $r \in \mathbb{N}$ and $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n \cdot n')}{2T}} \in (0, 1/2)$, then against any adversary, with probability $1-\lambda$ over the randomness of the transcript, its sequence of predictions is (α, β, n, n')-mean-conditioned moment multicalibrated with respect to $\mathcal{G}$ for $\beta = (k+1)\alpha + \frac{k}{2n}$ and:*

$$\alpha \leq \frac{1}{rn} + \frac{1}{rn'} + 4\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n \cdot n'}{\lambda}\right)}.$$

For $r = \frac{\sqrt{T}(n+n')}{\epsilon n \cdot n'\sqrt{2\ln(4|\mathcal{G}|n \cdot n'/\lambda)}}$, this gives:

$$\alpha \leq (4 + \epsilon)\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n \cdot n'}{\lambda}\right)}.$$

Proof. By Lemma E.1, the second part of Theorem 3.1 applies, and plugging in $L_0 = 4|\mathcal{G}|n \cdot n'$ and $c = \frac{1}{rn} + \frac{1}{rn'}$, we have that, with probability $(1-\lambda)$ over the randomness of the transcript:

$$\ln(L_T(\pi_T)) \leq \ln(4|\mathcal{G}|n \cdot n) + T\left(\frac{\eta}{rn} + \frac{\eta}{rn'} + 2\eta^2\right) + \eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}.$$

Now, note that

$$\begin{aligned}
\exp\left(\eta \max_{G,i,j} |V_T^{G,i,j}|\right) &= \max_{G,i,j} \exp\left(\eta |V_T^{G,i,j}|\right), \\
&\leq \max_{G,i,j} \left(\exp\left(\eta V_T^{G,i,j}\right) + \exp\left(-\eta V_T^{G,i,j}\right)\right), \\
&\leq \sum_{G,i,j} \left(\exp\left(\eta V_T^{G,i,j}\right) + \exp\left(-\eta V_T^{G,i,j}\right) + \exp\left(\eta M_T^{G,i,j}\right) + \exp\left(-\eta M_T^{G,i,j}\right)\right), \\
&= L_T(\pi_T).
\end{aligned}$$

By an analogous argument we have that $\exp\left(\eta \max_{G,i,j} |M_T^{G,i,j}|\right) \leq L_T(\pi_T)$.Taking log on both sides and dividing both sides by ηT, we get

$$\frac{1}{T}\max_{G,i} |V_T^{G,i,j}| \leq \frac{1}{\eta T}\ln(L_T(\pi_T)) \leq \frac{\ln(4|\mathcal{G}|n \cdot n')}{\eta T} + \frac{1}{rn} + \frac{1}{rn'} + 2\eta + \sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}}.$$

Choosing $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n\cdot n')}{2T}}$, we obtain:

$$\begin{aligned}\frac{1}{T}\max_{G,i,j}|V_T^{G,i,j}| &\leq \frac{1}{rn}+\frac{1}{rn'}+2\sqrt{\frac{2\ln(4|\mathcal{G}|n\cdot n')}{T}}+\sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}}\\ &\leq \frac{1}{rn}+\frac{1}{rn'}+4\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right)},\end{aligned}$$

and, by an analogous argument,

$$\frac{1}{T}\max_{G,i,j}|M_T^{G,i,j}| \leq \frac{1}{rn}+\frac{1}{rn'}+4\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right)},$$

as desired. □

Lemma 4.5. *Consider a linear program of the following form, with variables $x \in \mathbb{R}^m$, $\gamma \in \mathbb{R}$ for some m:*

$$\textit{Minimize } \gamma, \quad \textit{subject to:} \quad Ax \leq \gamma \mathbf{1}^m,\ x\cdot \mathbf{1}^m = 1,\ x \geq 0.$$

Here, $\mathbf{1}^m \in \mathbb{R}^m$ is the all-ones vector, and $A = (a_{ji})$ is a finite matrix with real entries.

Take any $\epsilon > 0$. Modify the above linear program by replacing matrix A with matrix $\tilde{A} = (\tilde{a}_{ji})$, where each $\tilde{a}_{ji}$ is a rational number within $\pm\frac{\epsilon}{2}$ from a_{ji}, obtained by truncating a_{ji} to $O(\log\frac{1}{\epsilon})$ bits of precision. Then, any optimal solution $(x^{*,r}, \gamma^{*,r})$ of the resulting rational linear program is an ϵ-approximately optimal feasible solution of the original linear program.

Proof. Let (x^*, γ^*) be the optimal solution of the original LP. Consider the constraint of the original (resp. rational) LP associated with any row j of matrix A (resp. $\tilde{A}$). This constraint is written as $\sum_i a_{ji}x_i \leq \gamma$ in the original LP, and $\sum_i \tilde{a}_{ji}x_i \leq \gamma$ in the rational LP. Here and below, i ranges over $[m]$. Now, we have that

$$\sum_i \tilde{a}_{ji}x_i^* \leq \sum_i \left(a_{ji}+\frac{\epsilon}{2}\right)x_i^* = \sum_i a_{ji}x_i^* + \frac{\epsilon}{2}\sum_i x_i^* \leq \gamma^* + \frac{\epsilon}{2}\sum_i x_i^* = \gamma^* + \frac{\epsilon}{2}.$$

Since this holds for any row j of the matrix, then setting $x = x^*$ achieves value at most $\gamma^* + \frac{\epsilon}{2}$ with respect to the rational LP.

Conversely, consider an optimal solution $(x^{*,r}, \gamma^{*,r})$ of the rational LP — by the above, we immediately have $\gamma^{*,r} \leq \gamma^* + \frac{\epsilon}{2}$. We claim it achieves value at most $\gamma^* + \epsilon$ with respect to the original LP. Indeed, for any matrix row j,

$$\sum_i a_{ji}x_i^{*,r} \leq \sum_i \left(\tilde{a}_{ji}+\frac{\epsilon}{2}\right)x_i^{*,r} = \sum_i \tilde{a}_{ji}x_i^{*,r} + \frac{\epsilon}{2}\sum_i x_i^{*,r} = \sum_i \tilde{a}_{ji}x_i^{*,r} + \frac{\epsilon}{2} \leq \gamma^{*,r} + \frac{\epsilon}{2} \leq \left(\gamma^* + \frac{\epsilon}{2}\right) + \frac{\epsilon}{2} = \gamma^* + \epsilon.$$

Therefore, by solving the rational LP, we obtain an ϵ-approximate solution to the original LP, as desired. □

Lemma 4.6. *Algorithm 4 achieves the multivalidity guarantees specified in Theorem 4.3.*

Proof. We briefly argue that the additive ϵ-approximation to the (shifted and rescaled) value of the game results in the claimed dependence of the multivalidity guarantees on ϵ. When the learner achieves an ϵ approximation to the value of the game at each round, the statement of Corollary 4.1 becomes:

$$\mathop{\mathbb{E}}_{Q_{s+1}^L}[\tilde{L}_{s+1}|\pi_s] \leq L_s\left(1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)+\eta\epsilon \leq L_s\left(1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)+\epsilon.$$

Indeed, recall that the linear program that we solve at each round solves for the value of the game that has been shifted by $2\eta^2 L_s$ *and divided by* η. For the second inequality, recall that $\eta < 1$.

Now, using the telescoping argument from the first part of the proof of Theorem 3.1, we obtain

$$\begin{aligned}
\exp\left(\eta \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|]\right) \leq & 4|\mathcal{G}|n \cdot n' \left((1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)^T + \epsilon \sum_{t=0}^{T-1} \left(1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)^t, \\
\leq & 4|\mathcal{G}|n \cdot n' \left((1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)^T + \epsilon T \left(1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)^T, \\
= & (4|\mathcal{G}|n \cdot n' + \epsilon T) \exp\left(T \ln\left(1+\frac{\eta}{rn}+\frac{\eta}{rn'}+2\eta^2\right)\right), \\
\leq & (4|\mathcal{G}|n \cdot n' + \epsilon T) \exp\left(\frac{T\eta}{rn}+\frac{T\eta}{rn'}+2T\eta^2\right),
\end{aligned}$$

Taking logs and dividing by ηT, we get

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|] \leq \frac{\ln(4|\mathcal{G}|n \cdot n' + \epsilon T)}{\eta T} + \frac{1}{rn} + \frac{1}{rn'} + 2\eta.$$

Setting the two terms involving η equal, we have:

$$\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n \cdot n' + \epsilon T)}{2T}}.$$

For this choice of η, we obtain the following *in-expectation* multivalidity guarantee (and the same guarantee for the M's):

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|] \leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n \cdot n' + \epsilon T)}{T}}.$$

Now, setting $\epsilon = \frac{\epsilon'}{T}$ for any desired $\epsilon' > 0$, we obtain the guarantee (and same for the M's) that

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|] \leq \frac{1}{rn} + \frac{1}{rn'} + 2\sqrt{\frac{2\ln(4|\mathcal{G}|n \cdot n' + \epsilon')}{T}} \quad \text{if we set } \eta = \sqrt{\frac{\ln(4|\mathcal{G}|n \cdot n' + \epsilon')}{2T}},$$

and the resulting runtime will be polynomial in T and $\log\frac{1}{\epsilon}$ and thus polynomial in T and $\log\frac{1}{\epsilon'}$.

Now, we show the *high-probability* multivalidity guarantee. In the proof of Theorem 3.1, the statement of Lemma D.1 changes to:

Lemma E.2. *For any π_T, we have*

$$\sum_{t=1}^{T}\left(\mathop{\mathbb{E}}_{\tilde{\pi}_t}\left[\ln(\tilde{X}_t)\middle|\pi_{t-1}\right] - \ln(X_{t-1}(\pi_{t-1}))\right) \leq T\left(\eta c + 2\eta^2 + \epsilon\right).$$

Proof. Fixing π_T and taking any $t \leq T$, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\tilde{\pi}_t}\left[\ln(\tilde{X}_t)|\pi_{t-1}\right] \leq \ln\left(\mathop{\mathbb{E}}_{\tilde{\pi}_t}[\tilde{X}_t|\pi_{t-1}]\right), && \text{(Jensen's inequality)} \\
&\leq \ln\left(X_{t-1}(\pi_{t-1}) \cdot \left(1+c\eta+2\eta^2\right)+\epsilon\right), && \text{(since we computed an } \epsilon\text{-approximation)} \\
&\leq \ln\left(X_{t-1}(\pi_{t-1}) \cdot \left(1+c\eta+2\eta^2\right)\right) + \frac{\epsilon}{X_{t-1}(\pi_{t-1}) \cdot (1+c\eta+2\eta^2)}, && (\ln(x+y) \leq \ln(x) + \frac{y}{x} \text{ for } x,y \geq 0) \\
&\leq \ln(X_{t-1}(\pi_{t-1})) + \ln\left(1+c\eta+2\eta^2\right)+\epsilon, && \text{(since the loss satisfies } X_{t-1}(\pi_{t-1}) \geq 1) \\
&\leq \ln(X_{t-1}(\pi_{t-1})) + \left(c\eta+2\eta^2+\epsilon\right). && (\ln(1+x) \leq x \text{ for any } x > -1)
\end{aligned}$$

Summing over every round $t \in [T]$ gives us the result. ∎

Thus, the statement of the second part of Theorem 3.1 becomes that with probability $1-\lambda$,

$$\ln(X_T(\pi_T)) \leq \ln(X_0) + T\left(\eta c + 2\eta^2 + \epsilon\right) + \eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}.$$

Now, applying it to the setting at hand, we obtain:

$$\ln(L_T(\pi_T)) \leq \ln(4|\mathcal{G}|n\cdot n') + T\left(\eta\left(\frac{1}{rn}+\frac{1}{rn'}\right) + 2\eta^2 + \epsilon\right) + \eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}.$$

Thus, taking log on both sides and dividing both sides by ηT, we get

$$\frac{1}{T}\max_{G,i,j}|V_T^{G,(i,j)}| \leq \frac{1}{\eta T}\ln(L_T(\pi_T)) \leq \frac{\ln(4|\mathcal{G}|n\cdot n')}{\eta T} + \frac{1}{rn} + \frac{1}{rn'} + 2\eta + \frac{\epsilon}{\eta} + \sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}}.$$

Choosing $\eta = \sqrt{\frac{\ln(4|\mathcal{G}|n\cdot n')+\epsilon T}{2T}}$, we obtain (and the same holds for the M's):

$$\begin{aligned}\frac{1}{T}\max_{G,i,j}|V_T^{G,i,j}| &\leq \frac{1}{rn}+\frac{1}{rn'} + 2\sqrt{\frac{2(\ln(4|\mathcal{G}|n\cdot n')+\epsilon T)}{T}} + \sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}} \\ &\leq \frac{1}{rn}+\frac{1}{rn'} + 4\sqrt{\frac{2}{T}\ln\left(\frac{4|\mathcal{G}|n\cdot n'}{\lambda}\right)} + 2\epsilon,\end{aligned}$$

as desired. □

F Proofs from Section 5

Lemma 5.1. *For every transcript $\pi_s \in \Pi^*$, every $x_{s+1} \in \mathcal{X}$, and every $(\overline{\ell}_{s+1}, \overline{u}_{s+1}) \in B_n(i,j)$ we have that:*

$$\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1})) \leq \left(\eta(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1})])\right) C_s^{i,j}(x_{s+1}) + 2\eta^2 L_s,$$

where for each $i \leq j \in [n]$, we have defined

$$C_s^{i,j}(x_{s+1}) \equiv \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)}) - \exp(-\eta V_s^{G,(i,j)}).$$

When x_{s+1} is clear from context, for notational economy, we will elide it and simply write $C_s^{i,j}$.

Proof. We calculate:

$$\begin{aligned}
&\Delta_{s+1}(\pi_s, x_{s+1}, (\overline{\ell}_{s+1}, \overline{u}_{s+1})) \\
= &\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\left[\sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)})\left(\exp(\eta v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1})) - 1\right) + \exp(-\eta V_s^{G,(i,j)})\left(\exp(-\eta v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1}) - 1\right)\right] \\
\leq &\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}\left[\sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)})\left(\eta v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1}) + 2\eta^2\right) + \exp(-\eta V_s^{G,(i,j)})\left(-\eta v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1}) + 2\eta^2\right)\right] \\
= &\eta(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1})])C_s^{i,j} + 2\eta^2 \sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)}) + \exp(-\eta V_s^{G,(i,j)}) \\
\leq &\eta(\mathop{\mathbb{E}}_{\tilde{y}_{s+1}}[v_\delta((\overline{\ell}_{s+1}, \overline{u}_{s+1}), \tilde{y}_{s+1})])C_s^{i,j} + 2\eta^2 L_s,
\end{aligned}$$

as desired. Here the first inequality follows from the fact that for $0 < |x| < \frac{1}{2}$, $\exp(x) \leq 1 + x + 2x^2$, the following equality from organizing terms and the final inequality by noting that $\sum_{\mathcal{G}(x_{s+1})} \exp(\eta V_s^{G,(i,j)}) + \exp(-\eta V_s^{G,(i,j)}) \leq L_s$ by definition of L. ☐

Theorem 5.1. *When Algorithm 5 is run using n buckets, discretization parameter r and $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)}{2T}} \in (0, 1/2)$, then against any adversary constrained to playing (ρ, rn)-smooth distributions, its sequence of interval predictions is α-multivalid with respect to $\mathcal{G}$ in expectation over the randomness of the transcript π_T, where:*

$$\mathbb{E}[\alpha] \leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2)}{T}}.$$

Proof. From Observation 5.1, it suffices to show that $\frac{1}{T}\mathbb{E}_{\pi_T}[\max |V_T^{G,(i,j)}|] \leq \alpha$.

We begin by computing a bound on the (exponential of) the expectation of this quantity:

$$\begin{aligned}
\exp\left(\eta \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|]\right) &\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\exp\left(\eta \max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|\right)\right] \\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,(i,j)} \exp\left(\eta |\tilde{V}_T^{G,(i,j)}|\right)\right] \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\max_{G,(i,j)} \left(\exp\left(\eta \tilde{V}_T^{G,(i,j)}\right) + \exp\left(-\eta V_T^{G,(i,j)}\right)\right)\right] \\
&\leq \mathop{\mathbb{E}}_{\tilde{\pi}_T}\left[\sum_{G,(i,j)} \left(\exp\left(\eta \tilde{V}_T^{G,(i,j)}\right) + \exp\left(-\eta \tilde{V}_T^{G,(i,j)}\right)\right)\right] \\
&= \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\tilde{L}_T(\tilde{\pi}_T)] \\
&\leq 2|\mathcal{G}|n^2 \exp\left(T\eta\rho + 2T\eta^2\right).
\end{aligned}$$

Here the first inequality follows from Jensen's inequality and the last one follows from Lemma 5.4. Taking the log of both sides and dividing by ηT we obtain:

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|] \leq \frac{\ln(2|\mathcal{G}|n^2)}{\eta T} + \rho + 2\eta.$$

Choosing $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)}{2T}}$ we obtain:

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)} |\tilde{V}_T^{G,(i,j)}|] \leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2)}{T}},$$

as desired. ☐

Now, given $\tilde{L}$, define $\tilde{Z}$ analogously to the second part of Theorem 3.1. Next, we can show that the increments of $\tilde{Z}$ thusly defined, at any round t, can be bounded.

Lemma F.1. *At any round $t \in [T]$ and for any realized transcript π_t, $|Z_t - Z_{t-1}| \leq 2\eta$.*

Proof. Observe that

$$\begin{aligned}
|Z_t - Z_{t-1}| &= |\ln(L_t(\pi_t)) - \mathbb{E}[\ln(L_t(\tilde{\pi}_t))|\pi_{t-1}]| \\
&= \left|\mathbb{E}\left[\ln\left(\frac{L_t(\pi_t)}{L_t(\tilde{\pi}_t)}\right)\middle|\pi_{t-1}\right]\right|
\end{aligned}$$

Note that for any π_t,

$$L_t(\pi_t) = L_{t-1}(\pi_{t-1}) + \Delta_t(\pi_{t-1}, x_t, y_t, (\ell_t, \mu_t))$$

where:

$$\begin{aligned}
&\Delta_t(\pi_{t-1}, x_t, y_t, (\ell_t, u_t)) \\
&= \sum_{\mathcal{G}(x_t)} \exp(\eta V_{t-1}^{G, B_n^{-1}(\ell_t, u_t)}) \left(\exp(\eta v_\delta((\ell_t, u_t), y_t)) - 1\right) + \exp(-\eta V_{t-1}^{G, B_n^{-1}(\ell_t, u_t)}) \left(\exp(-\eta v_\delta((\ell_t, u_t), y_t) - 1\right).
\end{aligned}$$

Since $v_\delta((\ell_t, u_t), y_t)$ must lie in $[-1, 1]$ (actually $[-(1-\delta), \delta]$), we have that:

$$(\exp(-\eta) - 1) L_{t-1}(\pi_{t-1}) \leq \Delta_t(\pi_{t-1}, x_t, y_t, (\ell_t, u_t)) \leq (\exp(\eta) - 1) L_{t-1}(\pi_{t-1})$$

which implies:

$$\exp(-\eta) L_{t-1}(\pi_{t-1}) \leq L_t(\pi_t) \leq \exp(\eta) L_{t-1}(\pi_{t-1}).$$

Therefore, for any two π_t, π_t' such that the corresponding transcripts for the first $t-1$ periods are the same, we have

$$\left| \ln\left(\frac{L_t(\pi_t)}{L_t(\pi_t')}\right) \right| \leq \ln\left(\frac{\exp(\eta)}{\exp(-\eta)}\right) = 2\eta.$$

Therefore we have $\left| \mathbb{E}\left[\ln\left(\frac{L_t(\pi_t)}{L_t(\tilde{\pi}_t)}\right) \middle| \pi_{t-1} \right] \right| \leq 2\eta$ as desired. □

Theorem 5.2. *When Algorithm 5 is run using n buckets, discretization parameter r and $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)}{2T}} \in (0, 1/2)$, then against any adversary who is constrained to playing (ρ, rn)-smooth distributions, its sequence of interval predictions is α-multivalid with respect to $\mathcal{G}$ with probability $1-\lambda$ over the randomness of the transcript π_T:*

$$\alpha \leq \rho + 4\sqrt{\frac{2}{T} \ln\left(\frac{2|\mathcal{G}|n^2}{\lambda}\right)}.$$

Proof. By Lemma F.1, the second part of Theorem 3.1 applies, and plugging in $L_0 = 2|\mathcal{G}|n^2$ and $c = \rho$, we have that, with probability $(1-\lambda)$ over the randomness of the transcript:

$$\ln(L_T(\pi_T)) \leq \ln(2|\mathcal{G}|n^2) + T\left(\eta\rho + 2\eta^2\right) + \eta\sqrt{8T \ln\left(\frac{1}{\lambda}\right)}.$$

Now, note that

$$\begin{aligned}
\exp\left(\eta \max_{G,i,j} |V_T^{G,(i,j)}|\right) &= \max_{G,i,j} \exp\left(\eta |V_T^{G,(i,j)}|\right), \\
&\leq \max_{G,i,j} \left(\exp\left(\eta V_T^{G,(i,j)}\right) + \exp\left(-\eta V_T^{G,(i,j)}\right)\right), \\
&\leq \sum_{G,i,j} \left(\exp\left(\eta V_T^{G,(i,j)}\right) + \exp\left(-\eta V_T^{G,(i,j)}\right)\right), \\
&= L_T(\pi_T).
\end{aligned}$$

Taking log on both sides and dividing both sides by ηT, we get

$$\frac{1}{T} \max_{G,i,j} |V_T^{G,(i,j)}| \leq \frac{1}{\eta T} \ln(L_T(\pi_T)) \leq \frac{\ln(2|\mathcal{G}|n^2)}{\eta T} + \rho + 2\eta + \sqrt{\frac{8 \ln\left(\frac{1}{\lambda}\right)}{T}}.$$

Choosing $\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2)}{2T}}$, we obtain

$$\begin{aligned}\frac{1}{T}\max_{G,i,j}|V_T^{G,i,j}| &\leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2)}{T}} + \sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}} \\ &\leq \rho + 4\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n^2}{\lambda}\right)},\end{aligned}$$

as desired. □

Lemma 5.7. *Algorithm 6 achieves the multivalidity guarantees stated in Theorem 5.3.*

Proof. We briefly argue that the additive ϵ-approximation to the (shifted and rescaled) value of the game results in the claimed dependence of the multivalidity guarantees on ϵ. When the learner achieves an ϵ approximation to the value of the game at each round, the statement of Corollary 5.1 becomes:

$$\mathop{\mathbb{E}}_{(\ell,u)\sim Q_{s+1}^L}[\tilde{L}_{s+1}|\pi_s] \leq L_s\left(1+\eta\rho+2\eta^2\right)+\eta\epsilon \leq L_s\left(1+\eta\rho+2\eta^2\right)+\epsilon.$$

Indeed, recall that the linear program that we solve at each round solves for the value of the game that has been shifted by $2\eta^2 L_s$ *and divided by* η. For the second inequality, recall that $\eta < 1$.

Now, using the telescoping argument from the first part of the proof of Theorem 3.1, we obtain

$$\begin{aligned}\exp\left(\eta \mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)}|\tilde{V}_T^{G,(i,j)}|]\right) \leq& 2|\mathcal{G}|n^2\left(1+\eta\rho+2\eta^2\right)^T + \epsilon\sum_{t=0}^{T-1}(1+\eta\rho+2\eta^2)^t, \\ \leq& 2|\mathcal{G}|n^2\left(1+\eta\rho+2\eta^2\right)^T + \epsilon T(1+\eta\rho+2\eta^2)^T, \\ =& (2|\mathcal{G}|n^2+\epsilon T)\exp\left(T\ln\left(1+\eta\rho+2\eta^2\right)\right), \\ \leq& (2|\mathcal{G}|n^2+\epsilon T)\exp\left(T\eta\rho+2T\eta^2\right),\end{aligned}$$

Taking logs and dividing by ηT, we get

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)}|\tilde{V}_T^{G,(i,j)}|] \leq \frac{\ln(2|\mathcal{G}|n^2+\epsilon T)}{\eta T} + \rho + 2\eta.$$

Setting the two terms involving η equal, we have:

$$\eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2+\epsilon T)}{2T}}.$$

For this choice of η, we obtain the following *in-expectation* multivalidity guarantee:

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)}|\tilde{V}_T^{G,(i,j)}|] \leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2+\epsilon T)}{T}}.$$

Now, setting $\epsilon = \frac{\epsilon'}{T}$ for any desired $\epsilon' > 0$, we obtain the guarantee that

$$\frac{1}{T}\mathop{\mathbb{E}}_{\tilde{\pi}_T}[\max_{G,(i,j)}|\tilde{V}_T^{G,(i,j)}|] \leq \rho + 2\sqrt{\frac{2\ln(2|\mathcal{G}|n^2+\epsilon')}{T}} \quad \text{if we set } \eta = \sqrt{\frac{\ln(2|\mathcal{G}|n^2+\epsilon')}{2T}},$$

and the resulting runtime will be polynomial in T and $\log\frac{1}{\epsilon}$ and thus polynomial in T and $\log\frac{1}{\epsilon'}$.

Now, we show the *high-probability* multivalidity guarantee. In the proof of Theorem 3.1, the statement of Lemma D.1 changes to:

Lemma E.2. *For any π_T, we have*

$$\sum_{t=1}^{T}\left(\mathop{\mathbb{E}}_{\tilde{\pi}_t}\left[\ln(\tilde{X}_t)\middle|\pi_{t-1}\right]-\ln(X_{t-1}(\pi_{t-1}))\right)\leq T\left(\eta c+2\eta^2+\epsilon\right).$$

We show this updated claim in the proof of Lemma 4.6 of Section 4.3.
Thus, the statement of the second part of Theorem 3.1 becomes that with probability $1-\lambda$,

$$\ln(X_T(\pi_T))\leq\ln(X_0)+T\left(\eta c+2\eta^2+\epsilon\right)+\eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}.$$

Now, applying it to the setting at hand, we obtain:

$$\ln(L_T(\pi_T))\leq\ln(2|\mathcal{G}|n^2)+T\left(\eta\rho+2\eta^2+\epsilon\right)+\eta\sqrt{8T\ln\left(\frac{1}{\lambda}\right)}.$$

Thus, taking log on both sides and dividing both sides by ηT, we get

$$\frac{1}{T}\max_{G,i,j}|V_T^{G,(i,j)}|\leq\frac{1}{\eta T}\ln(L_T(\pi_T))\leq\frac{\ln(2|\mathcal{G}|n^2)}{\eta T}+\rho+2\eta+\frac{\epsilon}{\eta}+\sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}}.$$

Choosing $\eta=\sqrt{\frac{\ln(2|\mathcal{G}|n^2)+\epsilon T}{2T}}$, we obtain:

$$\begin{aligned}\frac{1}{T}\max_{G,i,j}|V_T^{G,i,j}|&\leq\rho+2\sqrt{\frac{2(\ln(2|\mathcal{G}|n^2)+\epsilon T)}{T}}+\sqrt{\frac{8\ln\left(\frac{1}{\lambda}\right)}{T}}\\&\leq\rho+4\sqrt{\frac{2}{T}\ln\left(\frac{2|\mathcal{G}|n^2}{\lambda}\right)+2\epsilon},\end{aligned}$$

as desired. □