Distribution-free Risk-controlling Prediction Sets

Stephen Bates [*1,2] **Anastasios Angelapolous** [*2] **Lihua Lei** [*3] **Jitendra Malik** [2] **Michael I. Jordan** [1,2]

Abstract

While improving prediction accuracy has been the focus of machine learning in recent years, this alone does not suffice for reliable decision-making. Deploying learning systems in consequential settings also requires calibrating and communicating the uncertainty of predictions. To convey instance-wise uncertainty for prediction tasks, we show how to generate set-valued predictions from a black-box predictor that control the expected loss on future test points at a user-specified level. Our approach provides explicit finite-sample guarantees for any dataset by using a holdout set to calibrate the size of the prediction sets. This framework enables simple, distribution-free, rigorous error control for many tasks, and we demonstrate it in five large-scale machine learning problems: (1) classification problems where some mistakes are more costly than others; (2) multi-label classification, where each observation has multiple associated labels; (3) classification problems where the labels have a hierarchical structure; (4) image segmentation, where we wish to predict a set of pixels containing an object of interest; and (5) protein structure prediction. Lastly, we discuss extensions to uncertainty quantification for ranking, metric learning and distributionally robust learning.

1. Introduction

Black-box predictive algorithms have begun to be deployed in many real-world decision-making settings. Problematically, however, these algorithms are rarely accompanied by reliable uncertainty quantification. Algorithm developers often depend on the standard training/validation/test paradigm to make assertions of accuracy, stopping short of any further attempt to indicate that an algorithm's predictions should be treated with skepticism. Thus, prediction failures will often be silent ones, which is particularly alarming in high-consequence settings.

While one reasonable response to this problem involves retreating from black-box prediction, such a retreat raises many unresolved problems, and it is clear that black-box prediction will be with us for some time to come. A second response is to modify black-box prediction procedures so that they provide reliable uncertainty quantification, thereby supporting a variety of post-prediction activities, including risk-sensitive decision-making, audits, and protocols for model improvement.

We introduce a method for modifying a black-box predictor to return a set of plausible responses that limits the frequency of costly errors to a level chosen by the user. Returning a set of responses is a useful way to represent uncertainty, since such sets can be readily constructed from any existing predictor and, moreover, they are often interpretable. We call our proposed technique *risk-controlling prediction sets* (RCPS). The idea is to produce prediction sets that provide distribution-free, finite-sample control of a general loss.

Formally, for a test point with features $X \in \mathcal{X}$, a response $Y \in \mathcal{Y}$, we consider set-valued predictors $\mathcal{T}(X) : \mathcal{X} \to \mathcal{Y}'$ where $\mathcal{Y}'$ is some space of sets; we take $\mathcal{Y}' = 2^{\mathcal{Y}}$ for most of this work. We then have a loss function on set-valued predictions $L : \mathcal{Y} \times \mathcal{Y}' \to \mathbb{R}$ that encodes our notion of consequence, and seek a predictor $\mathcal{T}$, that controls the risk $R(\mathcal{T}) = \mathbb{E}[L(Y, \mathcal{T}(X))]$. Our goal in this work is to create set-valued predictors from training data that have risk that is below some desired level γ, with high probability. Specifically, we seek the following:

Definition 1 (Risk-controlling prediction sets) *Let $\mathcal{T}$ be a random function taking values in the space of functions $\mathcal{X} \to \mathcal{Y}'$ (e.g., a functional estimator trained on data). We say that $\mathcal{T}$ is a* (γ, δ)-risk-controlling prediction set *if, with probability at least $1 - \delta$, we have $R(\mathcal{T}) \leq \gamma$.*

The error level (γ, δ) is chosen in advance by the user. The reader should think of 10% as a representative value of δ; the choice of γ will vary with the choice of loss function.

[*]Equal contribution [1]Department of Statistics, University of California, Berkeley [2]Department of EECS, University of California, Berkeley [3]Department of Statistics, Stanford University. Correspondence to: Stephen Bates <stephenbates@cs.berkeley.edu>.

Presented at the ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning.,

Related work

Prediction sets have a long history in statistics, going back at least to tolerance regions in the 1940s (Wilks, 1941; 1942; Wald, 1943; Tukey, 1947). Tolerance regions are sets that contain a desired fraction of the population distribution with high probability. For example, one may ask for a region that contains 90% of future test points with probability 99% (over the training data). See (Krishnamoorthy & Mathew, 2009) for an overview of tolerance regions. Recently, tolerance regions have been instantiated to form prediction sets for deep learning models (Park et al., 2020; 2021). In parallel, conformal prediction (Vovk et al., 1999; 2005) has been recognized as an attractive way of producing predictive sets with finite-sample guarantees. A particularly convenient form of conformal prediction, known as *split conformal prediction* (Papadopoulos et al., 2002; Lei et al., 2015), uses data splitting to generate prediction sets in a computationally efficient way; see also (Vovk, 2015; Barber et al., 2019a) for generalizations that re-use data for improved statistical efficiency. Conformal prediction is a generic approach, and much recent work has focused on designing specific conformal procedures to have good performance according to metrics such as small set sizes (Sadinle et al., 2019), approximate coverage in all regions of feature space (Barber et al., 2019b; Romano et al., 2019; Izbicki et al., 2019; Romano et al., 2020; Cauchois et al., 2020b; Guan, 2020; Angelopoulos et al., 2021), and errors balanced across classes (Lei, 2014; Sadinle et al., 2019; Hechtlinger et al., 2018; Guan & Tibshirani, 2019). Further extensions of conformal prediction address topics such as distribution estimation (Vovk et al., 2019), causal inference (Lei & Candès, 2020), and handling or testing distribution shift (Tibshirani et al., 2019; Cauchois et al., 2020a; Hu & Lei, 2020). As an alternative to conformal prediction and tolerance regions, there is also a set of techniques that approach the tradeoff between small sets and high coverage by defining a utility function balancing these two considerations and finding the set-valued predictor that maximizes this utility (Grycko, 1993; del Coz et al., 2009; Mortier et al., 2020). The present work concerns the construction of tolerance regions with a user-specified coverage guarantee, and we do not pursue this latter formulation here.

In the current work, we expand the notion of tolerance regions to apply to a wider class of losses for set-valued predictors. Our development is inspired by the nested set interpretation of conformal prediction articulated in (Gupta et al., 2020), and our proposed algorithm is somewhat similar to split conformal prediction. Unlike conformal prediction, however, we pursue the high-probability error guarantees of tolerance regions and thus rely on entirely different proof techniques—see (Vovk, 2012) for a discussion of their relationship. As one concrete instance of this framework, we introduce a family of set-valued predictors that generalizes those of (Sadinle et al., 2019) to produce small set-valued predictions in a wide range of settings.

Our contribution

The central contribution of this work is a procedure to calibrate prediction sets to have finite-sample control of any loss satisfying a certain monotonicity requirement. The calibration procedure applies to any set-valued predictor, but we also show how to take any standard (non-set-valued) predictor and turn it into a set-valued predictor that works well with our calibration procedure. Our algorithm includes the construction of tolerance regions as special case, but applies to many other problems; this work explicitly considers classification with different penalties for different misclassification events, multi-label classification, classification with hierarchically structured classes, image segmentation, prediction problems where the response is a 3D structure, ranking, and metric learning.

2. Upper Confidence Bound Calibration

This section introduces our proposed method to calibrate any set-valued predictor so that it is guaranteed to have risk below a user-specified level, i.e., so that it satisfies Definition 1.

2.1. Setting and notation

Let $(X_i, Y_i)_{i=1,\ldots,m}$ be an independent and identically distributed (i.i.d.) set of variables, where the features vectors X_i take values in $\mathcal{X}$ and the response Y_i take values in $\mathcal{Y}$. To begin, split our data into a *training set* and a *calibration set*. Formally, let $\{\mathcal{I}_{\text{train}}, \mathcal{I}_{\text{cal}}\}$ form a partition of $\{1, \ldots, m\}$, and let $n = |\mathcal{I}_{\text{cal}}|$. Without loss of generality, we take $\mathcal{I}_{cal} = \{1, \ldots, n\}$. We allow the researcher to fit a predictive model on the training set $\mathcal{I}_{\text{train}}$ using an arbitrary procedure, calling the result $\hat{f}$, a function from $\mathcal{X}$ to some space $\mathcal{Z}$. The remainder of this paper shows how to subsequently create set-valued predictors from $\hat{f}$ that control a certain statistical error notion, regardless of the quality of the initial model fit or the distribution of the data. For this task, we will only use the calibration points $(X_1, Y_1), \ldots, (X_n, Y_n)$.

Next, let $\mathcal{T} : \mathcal{X} \to \mathcal{Y}'$ be a set-valued function (a *tolerance region*) that maps a feature vector to a set-valued prediction. This function would typically be constructed from the predictive model, $\hat{f}$, which was fit on the training data. We will describe one possible construction in detail in Section C. We further suppose we have a collection of such set-valued predictors indexed by a one-dimensional parameter λ taking values in a closed set $\Lambda \subset \mathbb{R} \cup \{\pm\infty\}$ that are *nested*, meaning that larger values of λ lead to larger sets:

$$\lambda_1 < \lambda_2 \implies \mathcal{T}_{\lambda_1}(x) \subset \mathcal{T}_{\lambda_2}(x). \tag{1}$$

To capture a notion of error, let $L(y, \mathcal{S}) : \mathcal{Y} \times \mathcal{Y}' \to \mathbb{R}_{\geq 0}$ be a *loss function* on prediction sets. For example, we could take $L(y, \mathcal{S}) = \perp$, which is the loss function corresponding to classical tolerance regions. We require that the loss function respects the following nesting property:

$$\mathcal{S} \subset \mathcal{S}' \implies L(y, \mathcal{S}) \geq L(y, \mathcal{S}'). \tag{2}$$

That is, larger sets lead to smaller loss. We then define the *risk* of a set-valued predictor $\mathcal{T}$ to be

$$R(\mathcal{T}) = \mathbb{E}[L(Y, \mathcal{T}(X))].$$

Since we will primarily be considering the risk of the tolerance functions from the family $\{\mathcal{T}_\lambda\}_{\lambda \in \Lambda}$, we will use the notational shorthand $R(\lambda)$ to mean $R(\mathcal{T}_\lambda)$. We further assume that there exists an element $\lambda_{\max} \in \Lambda$ such that $R(\lambda_{\max}) = 0$.

2.2. The procedure

Recalling Definition 1, our goal is to find a set function whose risk is less than some user-specified threshold γ. To do this, we search across the collection of functions $\{\mathcal{T}_\lambda\}_{\lambda \in \mathcal{T}}$ and estimate their risk on data not used for model training, $\mathcal{I}_{cal}$. We then show that by choosing the value of λ in a certain way, we can guarantee that the procedure has risk less than γ with high probability.

We assume that we have access to a pointwise upper confidence bound (UCB) for the risk function for each λ:

$$P\big(R(\lambda) \leq \underbrace{\widehat{R}^+(\lambda)}_{\text{UCB}}\big) \geq 1 - \delta, \tag{3}$$

where $\widehat{R}^+(\lambda)$ may depend on $(X_1, Y_1), \ldots, (X_n, Y_n)$. We will present a generic strategy to obtain such bounds by inverting a concentration inequality as well as concrete bounds for various settings in Section B. We choose $\hat{\lambda}$ as the smallest value of λ such that the entire confidence region to the right of λ falls below the target risk level γ:

$$\hat{\lambda} \triangleq \inf\left\{\lambda \in \Lambda : \widehat{R}^+(\lambda') < \gamma, \ \forall \lambda' \geq \lambda\right\}. \tag{4}$$

See Figure 1 for a visualization.

This choice of λ results in a set-valued predictor that controls the risk with high probability:

Theorem 1 (Validity of UCB calibration) *Let $(X_i, Y_i)_{i=1,\ldots,n}$ be an i.i.d. sample, let $L(\cdot, \cdot)$ be a loss satisfying the monotonicity condition in (2), and let $\{\mathcal{T}_\lambda\}_{\lambda \in \Lambda}$ be a collection of set predictors satisfying the nesting property in (1). Suppose (3) holds pointwise for each λ, and that $R(\lambda)$ is continuous. Then for $\hat{\lambda}$ chosen as in (4),*

$$P\left(R(\mathcal{T}_{\hat{\lambda}}) \leq \gamma\right) \geq 1 - \delta.$$

That is, $\mathcal{T}_{\hat{\lambda}}$ is a (γ, δ)-RCPS.

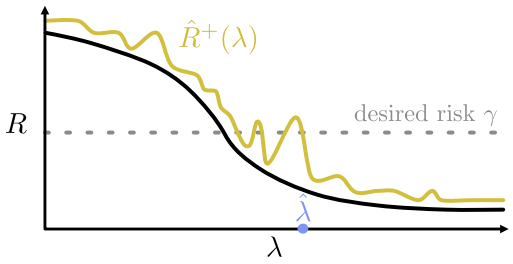

Figure 1. **Visualization of UCB calibration.**

All proofs are presented in Appendix D. Note that we are able to turn a pointwise convergence result into a result on the validity of a data-driven choice of λ. This is due to the monotonicity of the risk function; without the monotonicity, we would need a uniform convergence result on the empirical risk in order to get a similar guarantee. Next, we will show how to get the required concentration in (3) for cases of interest, so that we can carry out the UCB calibration algorithm. Later, in Section 3, we will introduce several concrete loss functions and empirically evaluate the performance of the UCB calibration algorithms in a variety of prediction tasks.

Remark 1 *Upper confidence bound calibration holds in more generality than the concrete instantiation above. The result holds for any monotone $R(\lambda)$ with a pointwise upper confidence bound $\hat{R}^+(\lambda)$. We present the general statement in Appendix D.*

Remark 2 *The above result also implies that UCB calibration gives an RCPS even if the data used to fit the initial preditive model comes from a different distribution. The only requirement is that the calibration data and the test data come from the same distribution.*

Remark 3 *We assumed that $R(\cdot)$ is continuous for simplicity, but this condition can be removed with minor modifications. The upper confidence bound is not assumed to be continuous.*

We show how to derive valid upper confidence bounds with concentration inequalities in Appendix B

3. Examples

Next, we apply our proposed method to five prediction problems. For each task, we introduce a relevant loss function and set-valued predictor, and then evaluate the performance of UCB calibration. The reader can reproduce our experiments using our public GitHub repository.

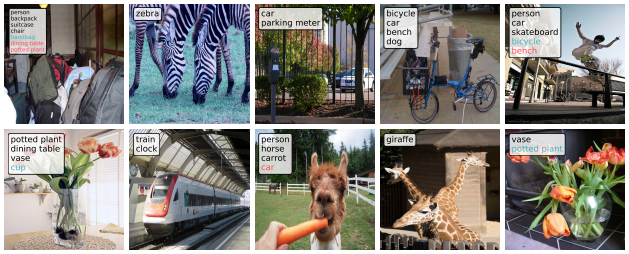

Figure 2. **Multi-label prediction set examples on MS COCO.** Black classes are correctly identified (true positives), blue ones are spurious (false positives), and red ones are missed (false negatives).

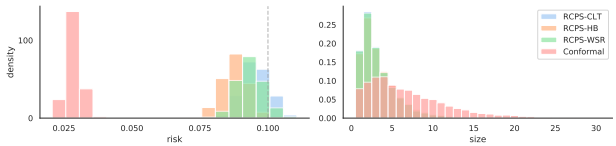

Figure 3. **Multi-label prediction set results on MS COCO.** The risk and set sizes are plotted as histograms over 1000 different random splits of MS COCO, with parameters $\gamma = 0.1$ and $\delta = 0.1$. We also include a conformal baseline. For details see Section 3.1.

3.1. Multi-label classification

Next, we consider the multi-label classification setting where each observation may have multiple corresponding correct labels; i.e., the response y is a subset of $\{1, ..., K\}$. Here, we seek to return prediction sets that control the loss

$$L(y, \mathcal{S}) = 1 - \frac{|y \cap \mathcal{S}|}{|y|}$$

at level γ. That is, we want to capture at least a $1 - \gamma$ proportion of the correct labels for each observation, on average. In this case, our nested sets

$$\mathcal{T}_\lambda(x) = \{z \in \{1, ..., K\} : \hat{\pi}_x(z) > -\lambda\}$$

depend on a classifier $\hat{\pi}_x$ that does not assume classes are exclusive, so their conditional probabilities generally do not sum to 1. Note that in this example we choose the output space $\mathcal{Y}'$ to be $\mathcal{Y} = 2^{\{1,...,K\}}$ (rather than $2^{\mathcal{Y}}$ as was done our previous example), since here $\mathcal{Y}$ is already a suitable space of sets.

To evaluate our method, we use the Microsoft Common Objects in Context (MS COCO) dataset, a large-scale, eighty-category dataset of natural images in realistic and often complicated contexts (Lin et al., 2014). We use TResNet as the base model, since it has the state-of-the-art classification performance on MS COCO at the time of writing (Ridnik et al., 2020). The standard procedure for multi-label estimation in computer vision involves training a convolutional neural network to output the vector of class probabilities, and then thresholding the probabilities in an ad-hoc manner return a set-valued prediction. Our method follows this general approach, but rigorously chooses the threshold so that the risk is controlled at a user-specified level γ, which we take to be 10%. To set the threshold, we choose $\hat{\lambda}$ as in Theorem B.3 using 4,000 calibration points, and then we evaluate the risk on an additional test set of 1,000 points. In Figure 2 we report on our our method's performance on ten randomly selected images from MS COCO, and in Figure 3 we quantitatively summarize the performance of our prediction sets. Our method controls the risk and gives sets with reasonable sizes.

In this setting, it is also possible to consider a conformal prediction baseline. To frame this problem in a way such that conformal prediction can be used, we follow (Cauchois et al., 2020b) and say that a test point is covered correctly if $y \subset T(x)$ and miscovered otherwise. That is, a point is covered only if the prediction set contains all true labels. The conformal baseline then uses the same set of set-valued predictors as above, but chooses the threshold as in (Cauchois et al., 2020b) so that there is probability $1 - \gamma$ that all of the labels per image are correctly predicted. In Figure 2, we find that the conformal baseline returns larger prediction sets. The reason is that the notion of coverage used by conformal prediction is more strict, requiring that all classes are covered. By contrast, the RCPS method can incorporate less brittle loss functions, such as the false negative rate.

3.2. Further examples

We present many more examples in Appendix A

4. Discussion

Risk-controlling prediction sets are a new way to represent uncertainty in predictive models. Since they apply to any existing model without retraining, they are straightforward to use in many situations. Our approach is closely related to that of split conformal prediction, but is more flexible in two ways. First, our approach can incorporate many loss functions, whereas conformal prediction controls the coverage—i.e, binary risk. The multilabel classification setting of Section 3.1 is one example where RCPS enables the use of a more natural loss function: the false negative rate. Second, risk-controlling prediction sets apply whenever one has access to a concentration result, whereas conformal prediction relies on exchangeability, a particular combinatorial structure. To summarize, in contrast to the standard train/validation/test split paradigm which only estimates global uncertainty (in the form of overall prediction accuracy), RCPS allow the user to automatically return *valid instance-wise uncertainty estimates* for many prediction tasks.

References

Angelopoulos, A. N., Bates, S., Malik, J., and Jordan, M. I. Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=eNdiU_DbM9.

Bahadur, R. R. and Savage, L. J. The nonexistence of certain statistical procedures in nonparametric problems. *Annals of Mathematical Statistics*, 27(4):1115–1122, 1956. doi: 10.1214/aoms/1177728077. URL https://doi.org/10.1214/aoms/1177728077.

Barber, R. F., Candès, E. J., Ramdas, A., and Tibshirani, R. J. Predictive inference with the jackknife+. *arXiv:1905.02928*, 2019a.

Barber, R. F., Candès, E. J., Ramdas, A., and Tibshirani, R. J. The limits of distribution-free conditional predictive inference. *arXiv:1903.04684*, 2019b.

Bentkus, V. On Hoeffding's inequalities. *Annals of Probability*, 32(2):1650–1673, 2004.

Bernal, J., Sánchez, J., and Vilarino, F. Towards automatic polyp detection with a polyp appearance model. *Pattern Recognition*, 45(9):3166–3182, 2012.

Bernstein, S. On a modification of Chebyshev's inequality and of the error formula of Laplace. *Ann. Sci. Inst. Sav. Ukraine, Sect. Math*, 1(4):38–49, 1924.

Borgli, H., Thambawita, V., Smedsrud, P. H., Hicks, S., Jha, D., Eskeland, S. L., Randel, K. R., Pogorelov, K., Lux, M., Nguyen, D. T. D., et al. Hyperkvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. *Scientific Data*, 7(1):1–14, 2020.

Brown, L. D., Cai, T., and DasGupta, A. Interval estimation for a binomial proportion. *Statistical science*, pp. 101–117, 2001.

Cauchois, M., Gupta, S., Ali, A., and Duchi, J. C. Robust validation: Confident predictions even when distributions shift. *arXiv:2008.04267*, 2020a.

Cauchois, M., Gupta, S., and Duchi, J. Knowing what you know: valid and validated confidence sets in multiclass and multilabel prediction. *arXiv:2004.10181*, 2020b.

del Coz, J. J., Díez, J., and Bahamonde, A. Learning nondeterministic classifiers. *Journal of Machine Learning Research*, 10(79):2273–2293, 2009. URL http://jmlr.org/papers/v10/delcoz09a.html.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, 2009.

Fan, D.-P., Ji, G.-P., Zhou, T., Chen, G., Fu, H., Shen, J., and Shao, L. Pranet: Parallel reverse attention network for polyp segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 263–273, 2020.

Fellbaum, C. Wordnet. *The Encyclopedia of Applied Linguistics*, 2012.

Grycko, E. Classification with set-valued decision functions. In *Information and Classification*, pp. 218–224, 1993. ISBN 978-3-642-50974-2.

Guan, L. Conformal prediction with localization. *arXiv:1908.08558*, 2020.

Guan, L. and Tibshirani, R. Prediction and outlier detection in classification problems. *arXiv:1905.04396*, 2019.

Gupta, C., Kuchibhotla, A. K., and Ramdas, A. K. Nested conformal prediction and quantile out-of-bag ensemble methods. *arXiv:1910.10562*, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hechtlinger, Y., Poczos, B., and Wasserman, L. Cautious deep learning. *arXiv:1805.09460*, 2018.

Hirschberg, D. S., Chandra, A. K., and Sarwate, D. V. Computing connected components on parallel computers. *Communications of the ACM*, 22(8):461–464, 1979.

Hoeffding, W. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. ISSN 01621459. URL http://www.jstor.org/stable/2282952.

Hu, X. and Lei, J. A distribution-free test of covariate shift using conformal prediction. *arXiv:2010.07147*, 2020.

Izbicki, R., Shimizu, G. T., and Stern, R. B. Flexible distribution-free conditional predictive bands using density estimators. *arXiv:1910.05575*, 2019.

Krishnamoorthy, K. and Mathew, T. *Statistical Tolerance Regions: Theory, Applications, and Computation*. Wiley, 2009. ISBN 9780470473894. URL https://books.google.com/books?id=1jQh0miU6PQC.

Lei, J. Classification with confidence. *Biometrika*, 101 (4):755–769, 10 2014. ISSN 0006-3444. doi: 10.1093/biomet/asu038. URL https://doi.org/10.1093/biomet/asu038.

Lei, J., Rinaldo, A., and Wasserman, L. A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, 74:29–43, 2015. doi: 10.1007/s10472-013-9366-6.

Lei, L. and Candès, E. J. Conformal inference of counterfactuals and individual treatment effects. *arXiv:2006.06138*, 2020.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Marcel, S. and Rodriguez, Y. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, pp. 1485–1488, 2010.

Maurer, A. Concentration inequalities for functions of independent variables. *Random Structures & Algorithms*, 29 (2):121–138, 2006.

Maurer, A. and Pontil, M. Empirical Bernstein bounds and sample variance penalization. *arXiv:0907.3740*, 2009.

Mortier, T., Wydmuch, M., Dembczyński, K., Hüllermeier, E., and Waegeman, W. Efficient set-valued prediction in multi-class classification. *arXiv:1906.08129*, 2020.

Papadopoulos, H., Proedrou, K., Vovk, V., and Gammerman, A. Inductive confidence machines for regression. In *Machine Learning: European Conference on Machine Learning*, pp. 345–356, 2002.

Park, S., Bastani, O., Matni, N., and Lee, I. PAC confidence sets for deep neural networks via calibrated prediction. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BJxVI04YvB.

Park, S., Li, S., Bastani, O., and Lee, I. PAC confidence predictions for deep neural network classifiers. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Qk-Wq5AIjpq.

Pinelis, I. and Utev, S. Exact exponential bounds for sums of independent random variables. *Theory of Probability and Its Applications*, 34:384–390, 1989. doi: 10.1137/1134032. (original text in Russian).

Pogorelov, K., Randel, K. R., Griwodz, C., Eskeland, S. L., de Lange, T., Johansen, D., Spampinato, C., Dang-Nguyen, D.-T., Lux, M., Schmidt, P. T., et al. Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection. In *Proceedings of the 8th ACM on Multimedia Systems Conference*, pp. 164–169, 2017.

Ridnik, T., Lawen, H., Noy, A., and Friedman, I. Tresnet: High performance gpu-dedicated architecture. *arXiv:2003.13630*, 2020.

Robinson, K. and Whelan, P. F. Efficient morphological reconstruction: a downhill filter. *Pattern Recognition Letters*, 25(15):1759–1767, 2004.

Romano, Y., Patterson, E., and Candès, E. Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, volume 32, pp. 3543–3553. 2019.

Romano, Y., Sesia, M., and Candès, E. J. Classification with valid and adaptive coverage. *arXiv:2006.02544*, 2020.

Sadinle, M., Lei, J., and Wasserman, L. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114:223 – 234, 2019.

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Zidek, A., Nelson, A. W., Bridgland, A., et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792): 706–710, 2020.

Silva, J., Histace, A., Romain, O., Dray, X., and Granado, B. Toward embedded detection of polyps in WCE images for early diagnosis of colorectal cancer. *International Journal of Computer Assisted Radiology and Surgery*, 9 (2):283–293, 2014.

Tibshirani, R. J., Foygel Barber, R., Candes, E., and Ramdas, A. Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems 32*, pp. 2530–2540. 2019.

Tukey, J. W. Non-parametric estimation ii. statistically equivalent blocks and tolerance regions–the continuous case. *Annals of Mathematical Statistics*, 18(4):529–539, 1947. doi: 10.1214/aoms/1177730343. URL https://doi.org/10.1214/aoms/1177730343.

Vovk, V. Conditional validity of inductive conformal predictors. In *Proceedings of the Asian Conference on Machine Learning*, volume 25, pp. 475–490, 2012.

Vovk, V. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):9–28, 2015.

Vovk, V., Gammerman, A., and Saunders, C. Machine-learning applications of algorithmic randomness. In *International Conference on Machine Learning*, pp. 444–453, 1999.

Vovk, V., Gammerman, A., and Shafer, G. *Algorithmic Learning in a Random World*. Springer, 2005. doi: 10.1007/b106715.

Vovk, V., Petej, I., Toccaceli, P., and Gammerman, A. Conformal calibrators. *arXiv:1902.06579*, 2019.

Wald, A. An extension of wilks' method for setting tolerance limits. *Annals of Mathematical Statistics*, 14(1):45–55, 1943. doi: 10.1214/aoms/1177731491. URL https://doi.org/10.1214/aoms/1177731491.

Waudby-Smith, I. and Ramdas, A. Variance-adaptive confidence sequences by betting. *arXiv preprint arXiv:2010.09686*, 2020.

Wilks, S. S. Determination of sample sizes for setting tolerance limits. *Annals of Mathematical Statistics*, 12(1):91–96, 1941. doi: 10.1214/aoms/1177731788. URL https://doi.org/10.1214/aoms/1177731788.

Wilks, S. S. Statistical prediction with special reference to the problem of tolerance limits. *Annals of Mathematical Statistics*, 13(4):400–409, 1942. doi: 10.1214/aoms/1177731537. URL https://doi.org/10.1214/aoms/1177731537.

Zhang, L., Yin, B., Wang, C., Jiang, S., Wang, H., Yuan, Y. A., and Wei, D. Structural insights into enzymatic activity and substrate specificity determination by a single amino acid in nitrilase from syechocystis sp. pcc6803. *Journal of structural biology*, 188(2):93–101, 2014.

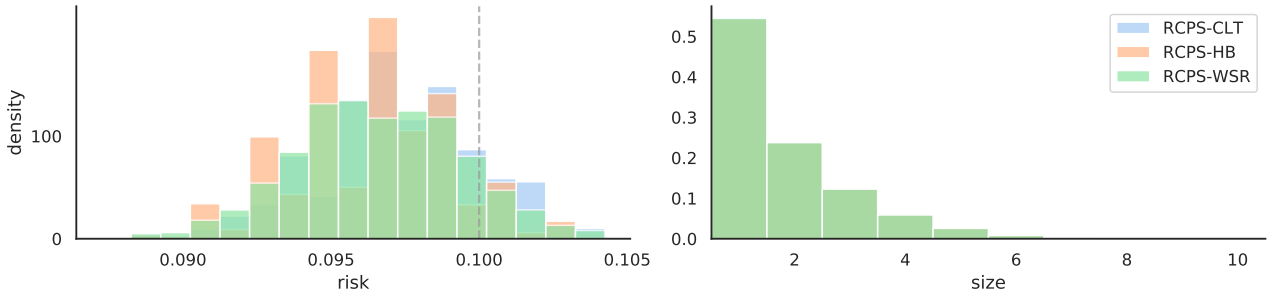

Figure 4. **Prediction set results on Imagenet.** The risk and set sizes for an RCPS are plotted as histograms over 100 different random splits of Imagenet, with parameters $\gamma = 0.1$ and $\delta = 0.1$. For details see Section A.1. The set sizes for all three methods overlap.

A. Examples, continued

This section lays out more examples of risk-controlling prediciton sets.

A.1. Classification with a class-varying loss

Suppose each observation has a single correct label y, and each label incurs a different, fixed loss if it is not correctly predicted:

$$L(y, \mathcal{S}) = L_y \mathbb{I}_{\{y \notin \mathcal{S}\}}.$$

This was the setting of our oracle result in Section C.2, and the medical diagnostic setting from the introduction also has this form. We would like to predict a set of labels that controls this loss. Towards that end, we define the family of nested sets

$$\mathcal{T}_\lambda(x) = \{y : \hat{\pi}_x(y) > -\lambda\},$$

where $\hat{\pi}_x : \mathcal{Y} \to [0, 1]$ represents a classifier, usually designed to estimate $P(Y|X)$. This family of nested sets simply returns the set of classes whose estimated conditional probability exceeds the value $-\lambda$, as in Figure 17. (The negative on λ comes from the definition of nesting, which asks sets to grow as λ grows.)

Here, we conduct an experiment on Imagenet—the gold-standard computer vision classification dataset—comprised of one thousand classes of natural images (Deng et al., 2009). For this experiment, we assign the loss L_y of class $y \in \{1, ..., 1000\}$ as $L(y) \overset{i.i.d.}{\sim} \text{Unif}(0, 1)$. We use a pretrained ResNet-152 from the `torchvision` repository as the base model $\hat{\pi}_x$ (Marcel & Rodriguez, 2010; He et al., 2016). We then choose $\hat{\lambda}$ as in Theorem B.3. Figure 4 summarizes the performance of our prediction sets over 100 random splits of Imagenet-Val with 30,000 points used for calibration and the remaining 20,000 used for evaluation. The RCPS procedure controls the risk at the correct level and the sets have reasonable sizes.

A.2. Hierarchical classification

Next, we discuss the application of RCPS to prediction problems where there exists a hierarchy on K labels. Here, we have a response variable $y \in \{1, \ldots, K\}$ with the structure on the labels encoded as a tree with nodes V and edges E with a designated root node, finite depth D, and K leaves, one for each label. To represent uncertainty while respecting the hierarchical structure, we seek to predict a node $\hat{y} \in V$ that is as precise as possible, provided that that is an ancestor of y. Note that with our tree structure, each $v \in V$ can be interpreted as a subset of $\{1, \ldots, K\}$ by taking the set of all the leaf-node descendants of v, so this setting is a special case of the set-valued prediction studied in this work.

We now turn to a loss function for this hierarchical label structure. Let $d : V \times V \to \mathbb{Z}$ be the function that returns the length of the shortest path between two nodes, let $\mathcal{A} : V \to 2^V$ be the function that returns the ancestors of its argument, and let $\mathcal{P} : V \to 2^V$ be the function that returns the set of leaf nodes that are descendants of its argument. Further define a hierarchical distance

$$d_H(v, u) = \inf_{a \in \mathcal{A}(v)} \{d(a, u)\}.$$

For a set of nodes $\mathcal{S} \in 2^V$, we then define the set-valued loss

$$L(y, \mathcal{S}) = \inf_{s \in \mathcal{S}} \{d_H(y, s)\}/D.$$

Figure 5. **Hierarchical predictions.** We show randomly selected examples of hierarchical prediction sets on Imagenet where the point prediction is incorrect but the prediction sets cover the true label. The black label is the ground truth class, the blue label is our prediction, and the red label is the top-1 output of a ResNet-18. Our prediction is an ancestor in the WordNet hierarchy of both the true class and the model's top-1 prediction. See the rightmost panel for an example subtree from the WordNet hierarchy.

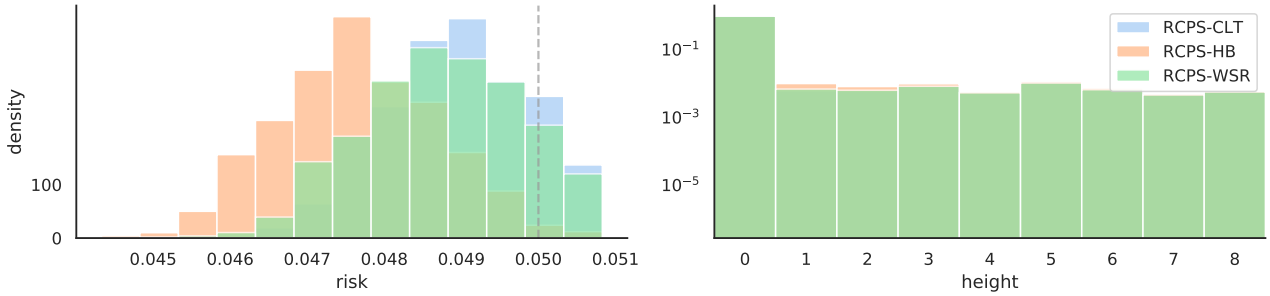

Figure 6. **The risk and height of RCPS for hierarchical classification.** We show histograms of risk and height (distance from the leaf node) over 100 different random splits of the Imagenet dataset, with parameters $\gamma = 0.05$ and $\delta = 0.1$. For details see Section A.2.

This loss returns zero if y is a child of any element in $\mathcal{S}$, and otherwise returns the minimum distance between any element of $\mathcal{S}$ and any ancestor of y, scaled by the depth D.

Lastly, we develop set-valued predictors that respect the hierarchical structure. Define a model $\hat{f} : \mathcal{X} \to [0,1]^K$ that outputs an estimated probability for each class. For any $x \in \mathcal{X}$, let $\hat{y}(x) = \arg\max_k \hat{f}(x)_k$ be the class with highest estimated probability. We also let $g(v,x) = \sum_{k \in \mathcal{P}(v)} \hat{f}(x)_k$ be the sum of scores of leaves descended from v. Then, we choose our family of set-valued predictors as:

$$\mathcal{T}_\lambda(x) = \underset{\{a \in \mathcal{A}(\hat{y}(x)) \,:\, g(a,x) \geq -\lambda\}}{cap} \mathcal{P}(a).$$

In words, we return the leaf nodes of the smallest subtree that includes $\hat{y}(x)$ that has estimated probability mass of at least $-\lambda$. This subtree has a unique root $v \in V$, so can equivalently view $\mathcal{T}_\lambda(x)$ as returning the node v.

We return to the Imagenet dataset for our empirical evaluations. The Imagenet labels form a subset of the WordNet hierarchy (Fellbaum, 2012), and we parsed them to form the tree. The maximum depth of the WordNet hierarchy is $D = 14$. Similarly to Section A.1, we used a pretrained ResNet-18 from the `torchvision` repository as the base model for Algorithm 1, and chose $\hat{\lambda}$ as in Theorem B.3. Figure 5 shows several examples of our hierarchical predictions on this dataset, and Figure 6 summarizes the performance of the predictor. As before, we find that RCPS controls the risk at the desired level, and the predictions are generally relatively precise (i.e., of low depth in the tree).

A.3. Image segmentation

In the binary segmentation setting, we are given an $d_1 \times d_2 \times c$-dimensional image $x \in \mathbb{R}^{d_1 \times d_2 \times c}$ and seek to predict a set of object pixels $y \subseteq \mathcal{G}$, where $\mathcal{G} = \{(i,j) : 1 \leq i \leq d_1, 1 \leq j \leq d_2\}$. Intuitively, y is a set of pixels that differentiates

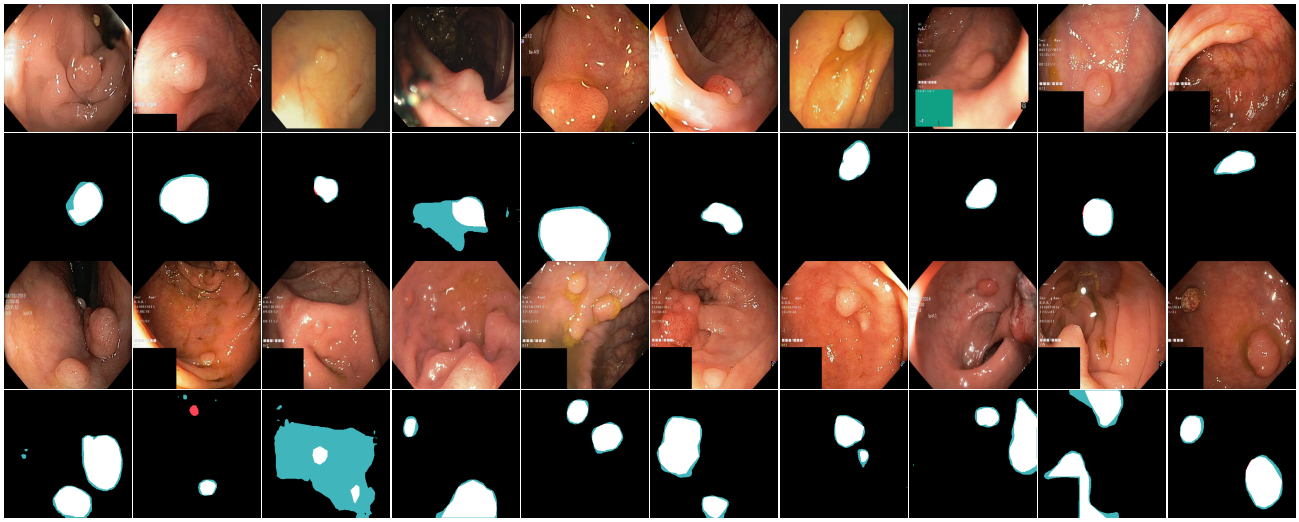

Figure 7. **Polyp segmentations.** We show examples of polyps along with prediction sets that capture 90% of the true polyp pixels per polyp per image, generated with our method using the CLT bound. White pixels are correctly identified polyp pixels (true positives), blue ones are spurious (false positives), and red ones are missed (false negatives). The top two rows show examples with a single polyp per image, and the second two rows show examples with two polyps per image.

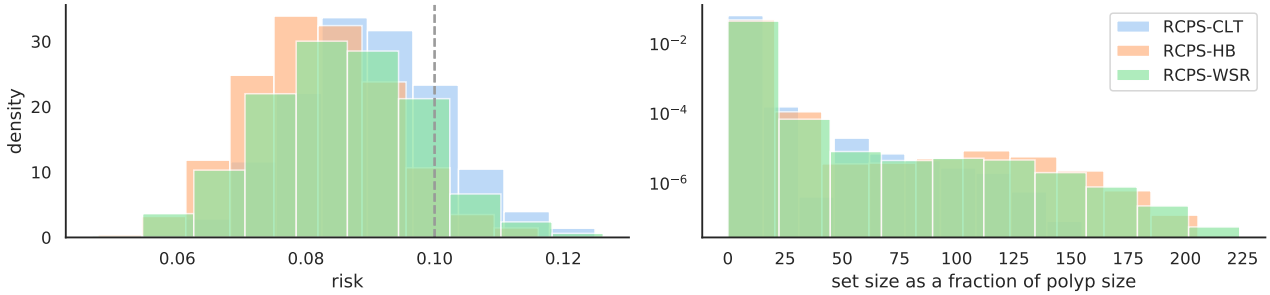

Figure 8. **Polyp segmentation results.** The risk and normalized set size are plotted as histograms over different random splits of the polyp dataset, with parameters $\gamma = 0.1$ and $\delta = 0.1$. For details see Section A.3.

objects of interest from the backdrop of the image.

Using the technique in Section 3.1, one may easily return prediction sets that capture at least a $1-\gamma$ proportion of the object pixels from each image with high probability. However, if there are multiple objects in the image, we may want to ensure our algorithm does not miss an entire, distinct object. Therefore, we target a different goal: returning prediction sets that capture a $1-\gamma$ fraction of the object pixels *from each object* with high probability. Specifically, consider $h : \mathcal{Y} \to 2^{\mathcal{Y}}$ to be an 8-connectivity connected components function (Hirschberg et al., 1979). Then $h(y)$ is a set of distinct regions of object pixels in the input image. For example, in the bottom right image of Figure 7, $h(y)$ would return two subsets of $\mathcal{G}$, one for each connected component. With this notation, we want to predict sets of pixels $\mathcal{S} \subseteq \mathcal{G}$ that control the proportion of missed pixels per object:

$$L(y, \mathcal{S}) = \frac{\sum_{y' \in h(y)} |y' \setminus \mathcal{S}|/|y'|}{|h(y)|}.$$

With this loss, if there are regions of different sizes, we would still incur a large loss for missing an entire small region, so this loss better captures our goal in image segmentation.

Having defined our loss, we now turn to our set construction. Standard object segmentation involves a model $\hat{f} : \mathbb{R}^{d_1 \times d_2} \to [0,1]^{d_1 \times d_2}$ that outputs approximate scores (e.g., after a sigmoid function) for each pixel in the image, then binarizes these scores with some threshold. To further our goal of per-object validity, in this experiment we additionally detect local peaks in the raw scores via morphological operations and connected components analysis, then re-normalize the connected regions by their maximum value. We will refer to this renormalization function as $r : [0,1]^{d_1 \times d_2} \to [0,1]^{d_1 \times d_2}$, and describe it

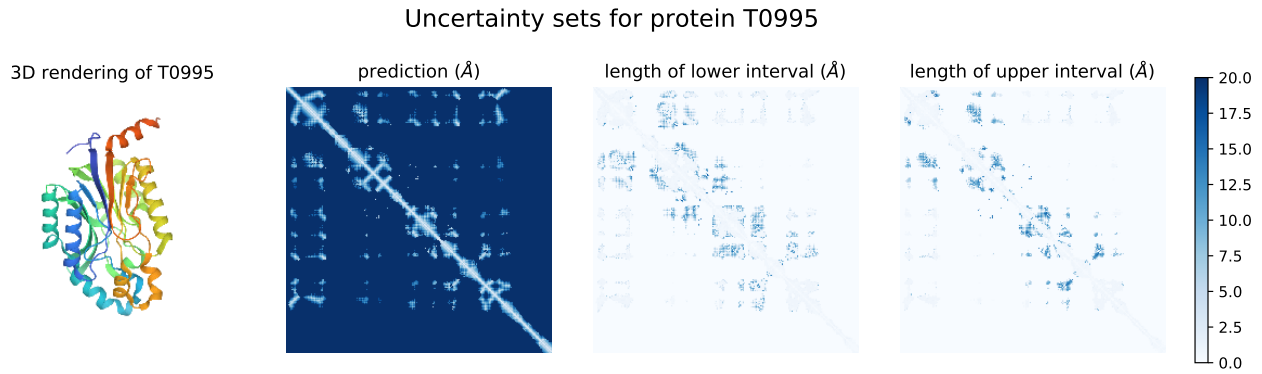

Figure 9. **Protein distograms.** We show AlphaFold's predicted distances between residues of protein T0995 along with prediction sets at $\gamma = 2\mathring{A}$ and $\delta = 0.1$. The prediction set for the whole protein is the union of distance intervals for each pair of residues, and the right two panels report the distance from the point prediction to the lower and upper endpoints for each of these intervals.

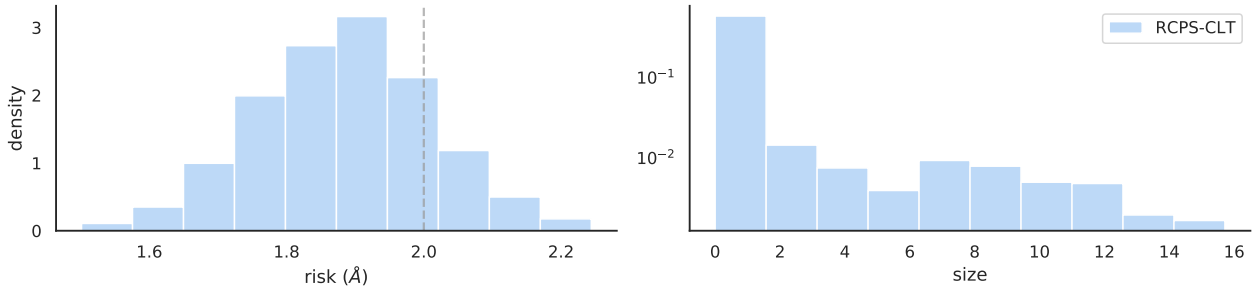

Figure 10. **Protein folding results.** The risk in $\mathring{A}$ and interval size (pooling all entries of each distogram) in $\mathring{A}$ are plotted as histograms, repeating for many random splits of the CASP-13 test-set.

precisely in Appendix G. We choose our family of set-valued predictors as

$$\mathcal{T}_\lambda = \{(i,j) : r(\hat{f}(x))_{i,j} \geq -\lambda\},$$

and then select $\hat{\lambda}$ as in Theorem B.3 or as in Theorem B.5.

We evaluated our method with an experiment combining several open-source polyp segmentation datasets: Kvasir (Pogorelov et al., 2017), Hyper-Kvasir (Borgli et al., 2020), CVC-ColonDB and CVC-ClinicDB (Bernal et al., 2012), and ETIS-Larib (Silva et al., 2014). Together, these datasets include 1,781 examples of segmented polyps, and in each experiment we use 1,000 examples for calibration and the remainder as a test set. We used PraNet (Fan et al., 2020) as our base segmentation model. In Figure 7 we report on our method's performance on 20 randomly selected images from the polyp datasets that contain at least two polyps, and in Figure 8 we summarize the quantitative performance of our prediction sets. RCPS again control the risk at the desired level, and the average prediction set size size is comparable to the average polyp size.

A.4. Protein folding

We finish the section by demonstrating RCPS for protein folding prediction, inspired by the recent success of AlphaFold. *Proteins* are biomolecules comprising one or more long chains of amino acids; when amino acids form a chemical bond to form a protein, they eject a water molecule and become amino acid *residues*. Each amino acid residue has a common amine-carboxyl backbone and a different *side chain* with electrical and chemical properties that together determine the 3D conformation of the whole protein, and thus its function. The so-called *protein folding problem* is to predict a protein's three dimensional structure from a list of its residues. A critical step in AlphaFold's protein folding pipeline involves predicting the distance between the β*-carbons* (the second-closest carbon to the side-chain) of each residue. These distances are then used to determine the protein's 3D structure. We express uncertainty directly on the distances between β-carbons.

Concretely, consider the alphabet $\Sigma = \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$, where each letter is the common abbreviation for an amino acid (for example, A denotes Alanine). The feature space consists of all possible

words over Σ, commonly denoted as $\mathcal{X} = \Sigma^*$. The label space $\mathcal{Y}$ is the set of all symmetric matrices with positive elements of any side length. In an example $(x, y) \in \mathcal{X} \times \mathcal{Y}$, the entry $y_{i,j}$ defines the distance in 3D space of residues x_i and x_j; hence, $y \in \mathbb{R}^{|x|\times|x|}$, and $y_{i,j} = y_{j,i}$. We seek to predict sets $\mathcal{S}$ that control the ℓ_1 projective distance from y to $\mathcal{S}$:

$$L(y, \mathcal{S}) = \inf_{s\in\mathcal{S}} \left\{ \frac{1}{|x|^2} \sum_{i,j} |y_{i,j} - s_{i,j}| \right\}.$$

Now we turn to the set construction, which we specialize to the AlphaFold pipeline. Because the AlphaFoldv2 codebase was not released at the time this paper was written, we use AlphaFoldv1 here (Senior et al., 2020). For a residue chain $x \in \mathcal{X}$, consider a variadic function $h(x) \in [0, 1]^{|x|\times|x|\times K}$, where K is a positive integer and $\sum_k h(x)_{i,j,k} = 1$ for all fixed choices of i and j. The function h represents a probability distribution over distances $d_1, ..., d_K$ for each distance between residues as a histogram; the output of h is referred to as a *distogram*. Given a distogram, we construct the family of set valued predictors

$$\mathcal{T}_\lambda(x) = \prod_{0\le i,j\le|x|} \{d_k : h(x)_{i,j,k} \ge -\lambda\}$$

and choose $\hat{\lambda}$ as in (4), as usual.

We evaluated our set construction algorithm on the 71 test points from the CASP-13 challenge on which DeepMind released the output of their model. In the AlphaFoldv1 pipeline, $K = 64$ and $d_1, ..., d_k = 2\mathring{A}, ..., 20\mathring{A}$. Since the data prepossessing pipepline was not released, no ground truth distance data is available. Instead, we generated semi-synthetic data points by sampling once from the distogram corresponding to each protein. We choose parameters $\gamma = 2\mathring{A}$ and $\delta = 0.1$, and, due to the small sample size (35 calibration and 36 test points), we only report results using the CLT bound, because the exact concentration results are hopelessly conservative with only 35 calibration points.

Figure 9 shows an example of our prediction sets on protein T0995 (PDB identifier 3WUY) (Zhang et al., 2014). Figure 10 shows the quantitative performance of the CLT, which nearly controls the risk. The strong performance of the CLT in this small-sample regime is encouraging and suggests that our methodology can be applied to problems even with small calibration sets.

B. Concentration Inequalities for the Upper Confidence Bound

In this section, we develop upper confidence bounds as in (3) under different conditions on the loss function, which will allow us to use the UCB calibration procedure for a variety of prediction tasks. In addition, for settings for which no finite-sample bound is available, we give an asymptotically valid upper confidence bound. Software implementing the upper confidence bounds is available in this project's public GitHub repository along with code to exactly reproduce our experimental results.

B.1. Bounded losses

We begin with the case where our loss is bounded above, and without loss of generality we take the bound to be one. We will present several upper confidence bounds and compare them in numerical experiments. The confidence bound of Waudby-Smith and Ramdas (Waudby-Smith & Ramdas, 2020) is the clear winner, and ultimately we recommend this bound for use in all cases with bounded loss.

B.1.1. Illustrative case: the simplified Hoeffding bound

It is natural to construct an upper confidence bound for $R(\lambda)$ based on the empirical risk, the average loss of the set-valued predictor $\mathcal{T}_\lambda$ on the calibration set:

$$\widehat{R}(\lambda) \triangleq \frac{1}{n} \sum_{i=1}^{n} L\left(Y_i, \mathcal{T}_\lambda(X_i)\right).$$

As a warm-up, recall the following simple version of Hoeffding's inequality:

Proposition B.1 (Hoeffding's inequality, simple version (Hoeffding, 1963)) *Suppose the loss is bounded above by one. Then,*

$$P\left(\widehat{R}(\lambda) - R(\lambda) \le -x\right) \le \exp\{-2nx^2\}.$$

This implies an upper confidence bound

$$\widehat{R}^{+}_{\text{sHoef}}(\lambda) = \widehat{R}(\lambda) + \sqrt{\frac{1}{2n}\log\left(\frac{1}{\delta}\right)}. \tag{5}$$

Applying Theorem 1 with

$$\begin{aligned}\hat{\lambda} = \hat{\lambda}^{\text{sHoef}} &\triangleq \inf\left\{\lambda \in \Lambda : \widehat{R}^{+}_{\text{sHoef}}(\lambda') < \gamma,\ \forall \lambda' \geq \lambda\right\} \\ &= \inf\left\{\lambda \in \Lambda : \widehat{R}(\lambda) < \gamma - \sqrt{\frac{1}{2n}\log\left(\frac{1}{\delta}\right)}\right\},\end{aligned} \tag{6}$$

we can generate an RCPS, which we record formally below.

Theorem B.1 (RCPS from Hoeffding's inequality) *In the setting of Theorem 1, assume additionally that the loss is bounded by one. Then, $\mathcal{T}_{\hat{\lambda}^{\text{sHoef}}}$ is a (γ, δ)-RCPS.*

In view of (6), UCB calibration with this version of Hoeffding's bound results in a procedure that is simple to state—one selects the smallest set size such that the empirical risk on the calibration set is below $\gamma - \sqrt{\log(1/\delta)/2n}$. This result is only presented for illustration purposes, however. Much tighter concentration results are available, so in practice we recommend using the better bounds described next.

B.1.2. Hoeffding–Bentkus bound

In general, an upper confidence bound can be obtained if the lower tail probability of $\widehat{R}(\lambda)$ can be controlled, in the following sense:

Proposition B.2 *Suppose $g(t; R)$ is a nondecreasing function in $t \in \mathbb{R}$ for every R:*

$$P(\widehat{R}(\lambda) \leq t) \leq g(t; R(\lambda)).$$

Then, $\widehat{R}^{+}(\lambda) = \sup\left\{R : g(\widehat{R}(\lambda); R) \geq \delta\right\}$ satisfies (3).

This result shows how a tail probability bound can be inverted to yield an upper confidence bound. Put another way, $g(\widehat{R}(\lambda); R)$ is a conservative p-value for testing the one-sided null hypothesis $H_0 : R(\lambda) \geq R$. From this perspective, Proposition B.2 is simply a restatement of the duality between p-values and confidence intervals.

The previous discussion of the simple Hoeffding bound is a special case of this proposition, but stronger results are possible. The rest of this section develops a sharper tail bound that builds on two stronger concentration inequalities.

We begin with a tighter version of Hoeffding's inequality.

Proposition B.3 (Hoeffding's inequality, tighter version (Hoeffding, 1963)) *Suppose the loss is bounded above by one. Then for any $t < R(\lambda)$,*

$$P\left(\widehat{R}(\lambda) \leq t\right) \leq \exp\{-nh_1(t; R(\lambda))\},$$

where $h_1(t; R) = t\log(t/R) + (1-t)\log((1-t)/(1-R))$.

The weaker Hoeffding inequality is implied by Proposition B.3 using the fact that $h_1(t; R) \geq 2(t-R)^2$. Another strong inequality is the Bentkus inequality, which implies that the Binomial distribution is the worst case up to a small constant. The Bentkus inequality is nearly tight if the loss function is binary, in which case $n\widehat{R}(\lambda)$ is binomial.

Proposition B.4 (Bentkus inequality (Bentkus, 2004)) *Suppose the loss is bounded above by one. Then,*

$$P\left(\widehat{R}(\lambda) \leq t\right) \leq eP\left(\text{Binom}(n, R(\lambda)) \leq \lceil nt \rceil\right),$$

where $\text{Binom}(n, p)$ denotes a binomial random variable with sample size n and success probability p.

Putting Proposition B.3 and B.4 together, we obtain a lower tail probability bound for $\widehat{R}(\lambda)$:

$$g^{\mathrm{HB}}(t; R(\lambda)) \triangleq \min(\exp\{-nh_1(t; R(\lambda))\},\ eP\left(\mathrm{Binom}(n, R(\lambda)) \leq \lceil nt \rceil\right)).$$

By Proposition B.2, we obtain a $(1-\delta)$ upper confidence bound for $R(\lambda)$ as

$$\widehat{R}^{+}_{\mathrm{HB}}(\lambda) = \sup\{R : g^{\mathrm{HB}}(\widehat{R}(\lambda); R) \geq \delta\}. \tag{7}$$

We obtain $\hat{\lambda}^{\mathrm{HB}}$ from $\widehat{R}^{+}_{\mathrm{HB}}(\lambda)$ as in (4) and conclude the following:

Theorem B.2 (RCPS from the Hoeffding–Bentkus bound) *In the setting of Theorem 1, assume additionally that the loss is bounded by one. Then, $\mathcal{T}_{\hat{\lambda}^{\mathrm{HB}}}$ is a (γ, δ)-RCPS.*

Remark 4 *The Bentkus inequality is closely related to an exact confidence region for the mean of a bimonial distribution. In the special where the loss takes values only in $\{0, 1\}$, this exact binomial result gives the most precise upper confidence bound and should always be used; see Appendix E.*

B.1.3. Waudby-Smith–Ramdas bound

Although the Hoeffding–Bentkus bound is nearly tight for binary loss functions, for non-binary loss functions, it can be very loose because it does not adapt to the variance of $L(Y_i, \mathcal{T}_\lambda(X_i))$. As an example, consider the extreme case where $\mathrm{Var}(L(Y_i, \mathcal{T}_\lambda(X_i))) = 0$, then $\widehat{R}(\lambda) = R(\lambda)$ almost surely, and hence $\widehat{R}^{+}(\lambda)$ can be set as $\widehat{R}(\lambda)$. In general, when $\mathrm{Var}(L(Y_i, \mathcal{T}_\lambda(X_i)))$ is small, the tail probability bound can be much tighter than that given by the Hoeffding–Bentkus bound. We next present a bound that is adaptive to the variance and improves upon the previous result in most settings.

The most well-known concentration result incorporating the variance is Bernstein's inequality (Bernstein, 1924). To accommodate the case where the variance is unknown and must be estimated, (Maurer & Pontil, 2009) proposed an empirical Bernstein inequality which replaces the variance by the empirical variance estimate. This implies the following upper confidence bound for $R(\lambda)$:

$$\widehat{R}^{+}_{\mathrm{eBern}}(\lambda) = \widehat{R}(\lambda) + \hat{\sigma}(\lambda)\sqrt{\frac{2\log(2/\delta)}{n}} + \frac{7\log(2/\delta)}{3(n-1)}, \quad \text{where } \hat{\sigma}^2(\lambda) = \frac{1}{n-1}\sum_{i=1}^{n}(L(Y_i, T_\lambda(X_i)) - \widehat{R}(\lambda))^2. \tag{8}$$

However, the constants in the empirical Bernstein inequality are not tight, and improvements are possible.

As an alternative bound that adapts to the unknown variance, (Waudby-Smith & Ramdas, 2020) recently proposed the *hedged capital confidence interval* for the mean of bounded random variables, drastically improving upon the empirical Bernstein inequality. Unlike all aforementioned bounds, it is not based on inverting a tail probability bound for $\widehat{R}(\lambda)$, but instead builds on tools from online inference and martingale analysis. For our purposes, we consider an one-sided variant of their result, which we refer to as the Waudby-Smith–Ramdas (WSR) bound.

Proposition B.5 (Waudby-Smith–Ramdas bound (Waudby-Smith & Ramdas, 2020)) *Let $L_i(\lambda) = L(Y_i, T_\lambda(X_i))$ and*

$$\hat{\mu}_i(\lambda) = \frac{1/2 + \sum_{j=1}^{i} L_j(\lambda)}{1+i},\ \hat{\sigma}_i^2(\lambda) = \frac{1/4 + \sum_{j=1}^{i}(L_j(\lambda) - \hat{\mu}_j(\lambda))^2}{1+i},\ \nu_i(\lambda) = \min\left\{1, \sqrt{\frac{2\log(1/\alpha)}{n\hat{\sigma}_{i-1}^2(\lambda)}}\right\}.$$

Further let

$$\mathcal{K}_i(R; \lambda) = \prod_{j=1}^{i}\{1 - \nu_j(\lambda)(L_j(\lambda) - R)\}, \quad \widehat{R}^{+}_{\mathrm{WSR}}(\lambda) = \inf\left\{R \geq 0 : \max_{i=1,\ldots,n} \mathcal{K}_i(R; \lambda) > \frac{1}{\delta}\right\}.$$

Then $\widehat{R}^{+}_{\mathrm{WSR}}(\lambda)$ is a $(1-\delta)$ upper confidence bound for $R(\lambda)$.

Since the result is a small modification of the one stated in (Waudby-Smith & Ramdas, 2020), for completeness we present a proof in Appendix D. As before, we then set $\hat{\lambda}^{\mathrm{WSR}}$ as in (4) to obtain the following corollary:

Theorem B.3 (RCPS from the Waudby-Smith–Ramdas bound) *In the setting of Theorem 1, assume additionally that the loss is bounded by 1. Then, $\mathcal{T}_{\hat{\lambda}^{\mathrm{WSR}}}$ is a (γ, δ)-RCPS.*

B.1.4. NUMERICAL EXPERIMENTS FOR BOUNDED LOSSES

We now evaluate the aforementioned bounds on random samples from a variety of distributions on $[0, 1]$. As an additional point of comparison, we also consider a bound based on the central limit theorem (CLT) that does not have finite-sample guarantees, formally defined later in Section B.3. In particular, given a distribution F for the loss $L(Y, \mathcal{T}_\lambda(X))$, we sample $L_1, \ldots, L_n \overset{\text{i.i.d.}}{\sim} F$ and compute the $(1 - \delta)$ upper confidence bound of the mean for $n \in \{\lfloor 10^r \rfloor : r = 2, 2.5, 3, 3.5, 4\}$ and $\delta \in \{0.1, 0.01, 0.001\}$. We present the results for $\delta = 0.1$ here and report on other choices of δs in Appendix F. Based on one million replicates of each setting, we report the coverage and the median gap between the UCB and true mean; the former measures the validity and the latter measures the power.

We consider the Bernoulli distribution, $F = \text{Ber}(\mu)$, and the Beta distribution, $F = \text{Beta}(a, b)$ with $b = a(1/\mu - 1)$. Note that both distributions have mean μ. Since a user would generally be interested in setting γ in $[0.001, 0.1]$ in practice, we set $\mu \in \{0.1, 0.01, 0.001\}$. To account for different levels of variability, we set $a \in \{0.1, 1, 10\}$ for the Beta distribution, with a larger a yielding a tighter concentration around the mean. We summarize the results in Figure 11. First, we observe that the CLT does not always have correct coverage, especially when the true mean is small, unless the sample size is large. Accordingly, we recommend the bounds with finite-sample guarantees in this case. Next, as shown in Figure 12, the WSR bound outperforms the others for all Beta distributions and has a similar performance to the HB bound for Bernoulli distributions. It is not surprising that the HB bound performs well for binary variables since the Bentkus inequality is nearly tight here. Based on these observations, we recommend the WSR bound for any non-binary bounded loss. When the loss is binary, one should use the exact result based on quantiles of the binomial distribution; see Appendix E.

B.2. Unbounded losses

We now consider the more challenging case of unbounded losses. As a motivating example, consider the Euclidean distance of a point to its closest point in the prediction set as a loss:

$$L\left(y, \mathcal{S}\right) = \inf\{||y - y'||_2 \ : \ y' \in \mathcal{S})\}.$$

Based on the well-known results of (Bahadur & Savage, 1956), we can show that it is impossible to derive a nontrivial upper confidence bound for the mean of nonnegative random variables in finite samples without any other restrictions—see Proposition D.1 in Appendix D. As a result, we must restrict our attention to distributions that satisfy some regularity conditions. One reasonable approach is to consider distributions satisfying a bound on the coefficient of variation, and we turn our attention to such distributions next.

B.2.1. THE PINELIS–UTEV INEQUALITY

For nonnegative random variables with bounded coeffecient of variation, the Pinelis–Utev inequality gives a tail bound as follows:

Proposition B.6 (Pinelis and Utev (Pinelis & Utev, 1989), Theorem 7) *Let $c_v(\lambda) = \sigma(\lambda)/R(\lambda)$ denote the coefficient of variation. Then for any $t \in (0, R(\lambda)]$,*

$$P(\widehat{R}(\lambda) \leq t) \leq \exp\left\{-\frac{n}{c_v^2(\lambda) + 1}\left[1 + \frac{t}{R(\lambda)}\log\left(\frac{t}{eR(\lambda)}\right)\right]\right\}.$$

By Proposition B.2, this implies an upper confidence bound of $R(\lambda)$:

$$\widehat{R}^+_{\text{PU}}(\lambda) = \sup\left\{R : \exp\left\{-\frac{n}{c_v^2(\lambda) + 1}\left[1 + \frac{\widehat{R}(\lambda)}{R}\log\left(\frac{\widehat{R}(\lambda)}{eR}\right)\right]\right\} \geq \delta\right\}. \tag{9}$$

This result shows that a nontrivial upper confidence bound can be derived if $c_v(\lambda)$ is known. When $c_v(\lambda)$ is unknown, we can treat it as a sensitivity parameter or estimate it based on the sample moments. Using this inequality and plugging in an upper bound c_v for $c_v(\lambda)$, we define $\hat{\lambda}^{\text{PU}}$ with the UCB calibration procedure (i.e, as in (4)) to get the following guarantee:

Theorem B.4 (RCPS from Pinelis–Utev inequality) *In the setting of Theorem 1, suppose in addition that for each $\lambda \in \Lambda$, $c_v(\lambda) \leq c_v$ for some constant c_v. Then, $\mathcal{T}_{\hat{\lambda}^{\text{PU}}}$ is a (γ, δ)-RCPS.*

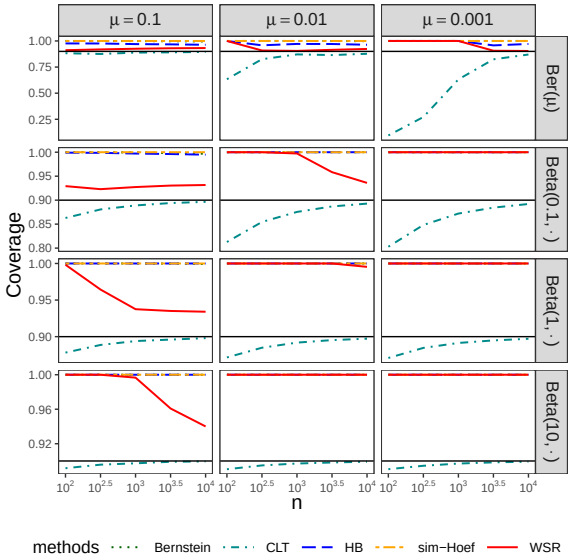

Figure 11. Coverage $P(\widehat{R}(\lambda) \geq R(\lambda))$

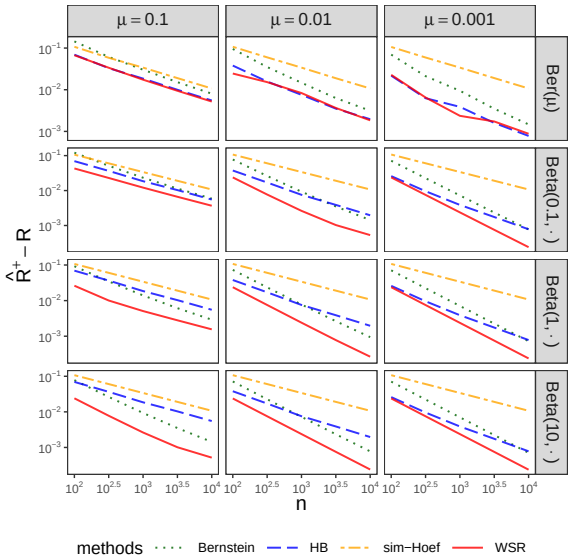

Figure 12. Median of $\widehat{R}^{+}(\lambda) - R(\lambda)$

Figure 13. **Numerical evaluations of concentration results for bounded losses**. We show the simple Hoeffding bound (5), HB bound (7), empirical Bernstein bound (8), CLT bound (10), and WSR bound (Proposition B.5) with sample size n. Each row corresponds to a type of distribution and each column corresponds to a value of the mean. The CLT bound is excluded in (b) because it does not achieve the target coverage in most of the cases.

B.2.2. NUMERICAL COMPARISONS OF UPPER CONFIDENCE BOUNDS

Next, we numerically study the unbounded case with two competing bounds—the PU bound with c_v estimated by the ratio between the standard error and the average, and a bound based on the CLT described explicitly later in Section B.3 (which does not have finite-sample coverage guarantees). We consider three types of distributions—the Gamma distribution $\Gamma(a, 1)$, the square-t distribution $t^2(v)$ (the distribution of the square of a t-distributed variable with degree of freedom v), and the log-normal distribution $\mathrm{LN}(\mu, \sigma)$ (the distribution of $\exp(Z)$ where $Z \sim N(\mu, \sigma)$). For each distribution, we consider a light-tailed and a heavy-tailed setting, and normalize the distributions to have mean $\mu = 1$. The parameter settings are summarized in Table 1.

Conducting our experiments as in the bounded case, we present the coverage and median gap with $\delta = 0.1$ in Figure 16. From Figure 14, we see that the CLT bound nearly achieves the desired coverage for light-tailed distributions but drastically undercovers for heavy-tailed distributions. By contrast, the PU bound has valid coverage in these settings. From Figure 15, we see that the CLT bound is much tighter in all cases, though the gap bewteen two bounds shrink as the sample size grows. Therefore, we recommend the CLT bound when the losses are believed to be light-tailed and the sample size is moderately large, and the PU bound otherwise.

	Gamma	Squared-t	Log-normal
light-tailed	$a = 1$	$v = 100$	$(\mu, \sigma) = (-0.125, 0.5)$
heavy-tailed	$a = 0.05$	$v = 4$	$(\mu, \sigma) = (-2, 2)$

Table 1. Distributions considered for the unbounded case.

B.3. Asymptotic results

When no finite-sample result is available, we can still use the UCB calibration procedure to get prediction sets with asymptotic validity. Suppose the loss $L(Y, \mathcal{T}_\lambda(X))$ has a finite second moment for each λ. Then, since the losses for each λ are i.i.d., we can apply the CLT to get

$$\lim_{n\to\infty} P\left(\frac{\sqrt{n}(\widehat{R}(\lambda) - R(\lambda))}{\hat{\sigma}(\lambda)} \leq -t\right) \leq \Phi\left(-t\right),$$

where Φ denotes the standard normal cumulative distribution function (CDF). This yields an asymptotic upper confidence bound for $R(\lambda)$:

$$\widehat{R}^+_{\mathrm{CLT}}(\lambda) = \widehat{R}(\lambda) + \frac{\Phi^{-1}(1-\delta)\hat{\sigma}(\lambda)}{\sqrt{n}}. \tag{10}$$

Let $\hat{\lambda}^{\mathrm{CLT}} = \inf\{\lambda \in \Lambda : \widehat{R}^+_{\mathrm{CLT}}(\lambda') < \gamma,\ \forall \lambda' \geq \lambda\}$. Then, $\mathcal{T}_{\hat{\lambda}^{\mathrm{CLT}}}$ is an asymptotic RCPS, as stated next.

Theorem B.5 (Asymptotically valid RCPS) *In the setting of Theorem 1, assume additionally that $L(Y, \mathcal{T}_\lambda(X))$ has a finite second moment for each λ. Then,*

$$\limsup_{n\to\infty} P\left(R(\mathcal{T}_{\hat{\lambda}^{\mathrm{CLT}}}) > \gamma\right) \leq \delta.$$

As a technical remark, note this result requires only a pointwise CLT for each $\lambda \in \Lambda$, analogously to the finite-sample version presented in Theorem 1. Since this asymptotic guarantee holds for many realistic choices of loss function and data-generating distribution, this approximate version of UCB calibration greatly extends the reach of our proposed method.

B.4. How large should the calibration set be?

The numerical results presented previously give rough guidance as to the required size of the calibration set. While UCB calibration is always guaranteed to control the risk by Theorem 1, if the calibration set is too small then the sets may be larger than necessary. Since our procedure finds the last point where the UCB $\widehat{R}^+(\lambda)$ is above the desired level γ, it will produce sets that are nearly as small as possible when $\widehat{R}^+(\lambda)$ is close to the true risk $R(\lambda)$. As a rule of thumb, we say that we have a sufficient number of calibration points when $\widehat{R}^+(\lambda)$ is within 10% of $R(\lambda)$. The sample size required will vary with the problem setting, but use this heuristic to analyze our simulation results to get a few representative values.

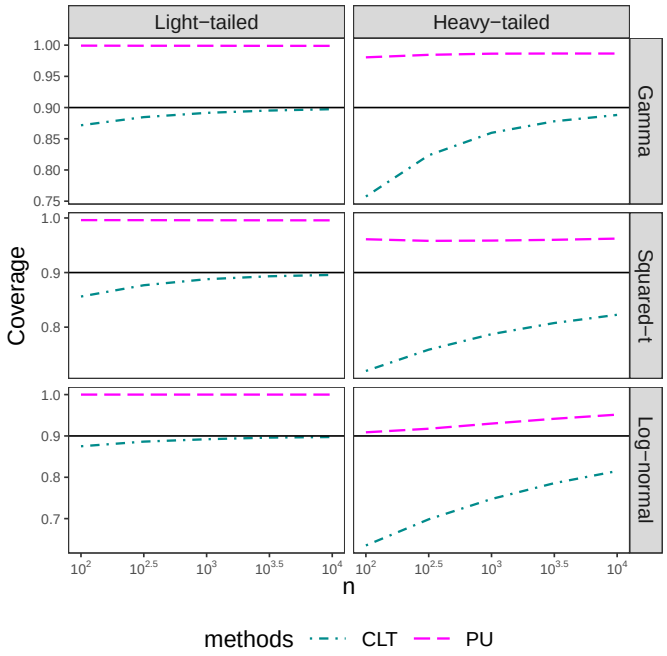

Figure 14. Coverage $P(\widehat{R}(\lambda) \geq R(\lambda))$

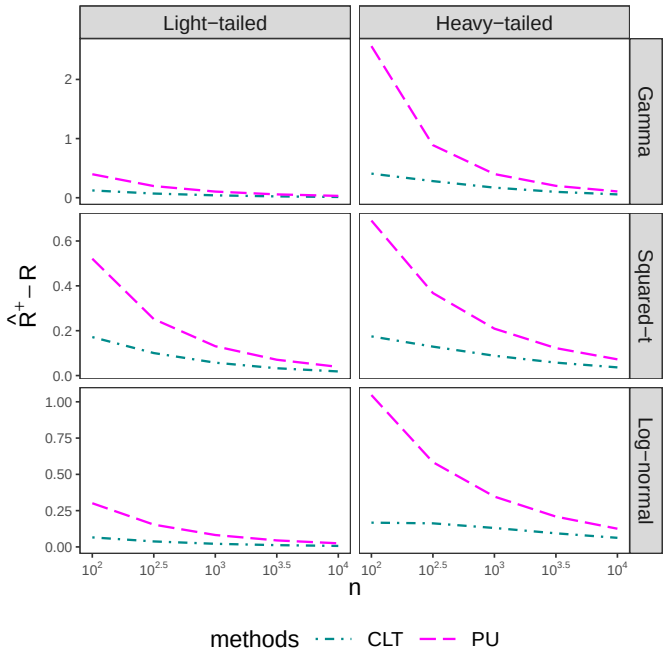

Figure 15. Median of $\widehat{R}^{+}(\lambda) - R(\lambda)$

Figure 16. **Numerical evaluations of the PU bound.** We compare the bound from (9) with the estimated coefficient of variation and the CLT bound (10), with sample size n from each distribution in Table 1. Each row corresponds to a type of distribution and each column corresponds to a value of the mean.

Algorithm 1 Greedy Sets

Input: λ, risk density estimate $\hat{\rho}_x$, step size $d\zeta$

0: **procedure** GREEDYSETS($\lambda, \hat{\rho}_x$)
0: $\mathcal{T} \leftarrow \emptyset$
0: $\zeta \leftarrow$ a large number (e.g., B in the bounded case)
0: **while** $\zeta > -\lambda$ **do**
0: $\zeta \leftarrow \zeta - d\zeta$
0: $\mathcal{T} \leftarrow \mathcal{T} \cup \left\{y' \in \mathcal{T}^c : \hat{\rho}_x(y', \mathcal{T}) > \zeta\right\}$
0: **return** $\mathcal{T}$

Output: The nested set with parameter λ at x: $\mathcal{T}_\lambda(x)$ =0

Figure 12 reports on the bounded loss case. The left column shows that when we seek to control the risk at the relatively loose $\gamma = 0.1$ level, around $1,000$ calibration points suffice; the middle panel shows that when we seek to control the risk at level $\gamma = 0.01$, a few thousand calibration points suffice; and the right column shows that for the strict risk level $\gamma = 0.001$, about $10,000$ calibration points suffice. The required number of samples will increase slightly if we ask for a higher confidence level (i.e., smaller δ), but the dependence on δ is minimal since the bounds will roughly scale as $\log(1/\delta)$—this scaling can be seen explicitly in the simple Hoeffding bound (5). Examining the unbounded loss examples presented in Figure 15, we see that about $1,000$ calibration points suffice for the student-t and log-normal examples, but that about $10,000$ calibration points are needed for the Gamma example. In summary, $1,000$ to $10,000$ calibration points are sufficient to generate prediction sets that are not too conservative, i.e., sets that have risk that are not far below the desired level γ.

C. Generating the Set-Valued Predictors

In this section, we describe one possible construction of the nested prediction sets $\mathcal{T}_\lambda(x)$ from a given predictor $\hat{f}$. Any collection of the sets can be used to control the risk by Theorem 1, but some may produce larger sets than others. Here, we present one choice and show that it is approximately optimal for an important class of losses.

In the following subsections, we denote the infinitesimal risk of a continuous response y with respect to a set $\mathcal{S} \subseteq \mathcal{Y}$ as its *conditional risk density*,

$$\rho_x(y, \mathcal{S}) = L(y, \mathcal{S}) p_{Y|X=x}(y).$$

We will present the results for the case where y is continuous, but the same algorithm and theoretical result hold in the discrete case if we instead take $\rho_x(y, \mathcal{S}) = L(y, \mathcal{S}) P(Y = y | X = x)$.

C.1. A greedy procedure

We now describe a construction of the tolerance functions $\mathcal{T}_\lambda$ based on the estimated conditional risk density. We assume that our predictor is $\hat{p}_x(y)$, an estimate of $p_{Y|X=x}(y)$, and we let $\hat{\rho}_x(y, \mathcal{S}) = L(y, \mathcal{S}) \hat{p}_x(y)$. Algorithm 1 indexes a family of sets $\mathcal{T}_\lambda$ nested in $\lambda \leq 0$ by iteratively including the riskiest portions of $\mathcal{Y}$, then re-computing the risk densities of the remaining elements. The general greedy procedure is computationally convenient; moreover, it is approximately optimal for a large class of useful loss functions, as we will prove soon.

Remark 5 *Algorithm 1 is greedy because it only considers the next $d\zeta$ portion of risk to choose which element to add to the current set. One can imagine versions of this algorithm which look ahead several steps. Such schemes may be tractable in some cases, but are generally much more computationally expensive.*

C.2. Optimality properties of the greedy procedure

Next, we outline a setting where our greedy algorithm is optimal. Suppose our loss function has the simple form $L(y, \mathcal{S}) = L_y \perp$, for constants L_y. This assumption on L describes the case where every y has a different, fixed loss if it is not present in the prediction set. In this case, the sets returned by Algorithm 1 have the form

$$\mathcal{T}_\lambda(x) = \{y' : \hat{\rho}_x(y', \emptyset) \geq \zeta(\lambda)\}.$$

That is, we return the set of response variables with risk density above some threshold; see Figure 17 for a visualization.

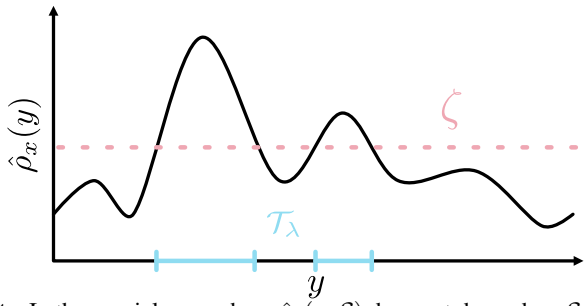

Figure 17. **Optimal prediction sets.** In the special case where $\hat{\rho}_x(y, \mathcal{S})$ does not depend on $\mathcal{S}$, $\mathcal{T}_\lambda(x)$ from Algorithm 1 is made up of the $y \in \mathcal{Y}$ whose conditional risk density exceeds a threshold ζ.

Now, imagine that we know the exact conditional probability density, $p_{Y|X=x}(y)$, and therefore the exact $\rho_x(y, \mathcal{S})$. The prediction sets produced by Algorithm 1 then have the smallest average size among all procedure that control the risk, as stated next.

Theorem C.1 (Optimality of the greedy sets) *In the setting above, let $\mathcal{T}' : \mathcal{X} \to \mathcal{Y}'$ be any set-valued predictor such that $R(\mathcal{T}') \leq R(\mathcal{T}_\lambda)$, where $\mathcal{T}_\lambda$ is given by Algorithm 1. Then,*

$$\mathbb{E}[|\mathcal{T}_\lambda(X)|] \leq \mathbb{E}[|\mathcal{T}'(X)|].$$

Here, $|\cdot|$ denotes the set size: Lebesgue measure for continuous variables and counting measure for discrete variables. This result is a generalization of a result of (Sadinle et al., 2019) to our risk-control setting. While we do not exactly know the risk density in practice and must instead use a plug-in estimate, this result gives us confidence that our set construction is a sensible one. The choice of the parameterization of the nested sets is the analogue to the choice of the score function in the more specialized setting of conformal prediction (Gupta et al., 2020), and it is known in that case that there are many choices that each have their own advantages and disadvantages. See (Sadinle et al., 2019; Romano et al., 2019) for further discussion of this point in that context.

C.3. Optimality in a more general setting

Next, we characterize the set-valued predictor that leads to the smallest sets for a wider class of losses. Suppose our loss takes the form

$$L(y; \mathcal{S}) = \int_{z \in \mathcal{S}^c} \ell(y, z) d\mu(z),$$

for some nonnegative ℓ and a finite measure μ. The function ℓ measures the cost of not including z in the prediction set when true response is y. For instance, $\ell(y, z) = L_y \mathbb{I}(y = z)$ and μ is the counting measure in the case considered above. Then the optimal $\mathcal{T}_\lambda$ is given by

$$\mathcal{T}_\lambda(x) = \{z : \mathbb{E}[\ell(Y; z) \mid X = x] \geq -\lambda\}, \tag{11}$$

for $\lambda \in \Lambda \subset (-\infty, 0]$, as stated next.

Theorem C.2 (Optimality of set predictors, generalized form) *In the setting above, let $\mathcal{T}' : \mathcal{X} \to \mathcal{Y}'$ be any set-valued predictor such that $R(\mathcal{T}') \leq R(\mathcal{T}_\lambda)$, where $\mathcal{T}_\lambda$ is given by (11). Then,*

$$\mathbb{E}[|\mathcal{T}_\lambda(X)|] \leq \mathbb{E}[|\mathcal{T}'(X)|].$$

For the case considered in Section C.2, $\mathbb{E}[\ell(Y; z) \mid X = x] = L_z p(z \mid x)$, so we see Theorem C.2 includes Theorem C.1 as a special case. As before, in practice we must estimate the distribution of Y given X from data, so we would not typically be able to implement this predictor exactly. Moreover, even if we perfectly knew the distribution of Y_i given $X = x$, the sets in (11) may not be easy to compute. Nonetheless, it is encouraging that we can understand the optimal set predictor for this important set of losses.

D. Proofs

Theorem D.1 (Validity of UCB calibration, abstract form) *Let $R : \Lambda \to \mathbb{R}$ be a continuous monotone nonincreasing function such that $R(\lambda) \leq \gamma$ for some $\lambda \in \Lambda$. Suppose $\widehat{R}^+(\lambda)$ is a random variable for each $\lambda \in \Lambda$ such that (3) holds pointwise. Then for $\hat{\lambda}$ chosen as in (4),*

$$P\left(R(\lambda) \leq \gamma\right) \geq 1 - \delta.$$

[Proof of Theorem D.1] Consider the smallest λ that controls the risk:

$$\lambda^* \triangleq \inf\{\lambda \in \Lambda : R(\lambda) \leq \gamma\}.$$

Suppose $R(\hat{\lambda}) > \gamma$. By the definition of λ^* and the monotonicity and continuity of $R(\cdot)$, this implies $\hat{\lambda} < \lambda^*$. By the definition of $\hat{\lambda}$, this further implies that $\widehat{R}^+(\lambda^*) < \gamma$. But since $R(\lambda^*) = \gamma$ (by continuity) and by the coverage property in (3), this happens with probability at most δ.

[Proof of Theorem 1] This follows from Theorem D.1.

[Proof of Proposition B.2] Let G denote the CDF of $\widehat{R}(\lambda)$. If $R(\lambda) > \widehat{R}^+(\lambda)$, then by definition, $g(\widehat{R}(\lambda); R(\lambda)) < \delta$. As a result,

$$P(R(\lambda) > \widehat{R}^+(\lambda)) \leq P(g(\widehat{R}(\lambda); R(\lambda)) < \delta) \leq P(G(\widehat{R}(\lambda)) < \delta).$$

Let $G^{-1}(\delta) = \sup\{x : G(x) \leq \delta\}$. Then

$$P(G(\widehat{R}(\lambda)) < \delta) \leq P(\widehat{R}(\lambda) < G^{-1}(\delta)) \leq \delta.$$

This implies that $P(R(\lambda) > \widehat{R}^+(\lambda)) \leq \delta$ and completes the proof.

[Proof of Proposition B.5] This proof is essentially a restatement of the proof of Theorem 4 in (Waudby-Smith & Ramdas, 2020). We present it here for completeness. Let $\mathcal{K}_i = \mathcal{K}_i(R(\lambda); \lambda)$, $\mathcal{F}_0$ be the trivial sigma-field, and $\mathcal{F}_i$ be the sigma-field generated by $(L_1(\lambda), \ldots, L_i(\lambda))$. Then $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \ldots \subset \mathcal{F}_n$ is a filtration. By definition, $\nu_i(\lambda) \in \mathcal{F}_{i-1}$ is a predictable sequence and $\mathcal{K}_i \in \mathcal{F}_i$. Since $\mathbb{E}[L_i(\lambda)] = R(\lambda)$,

$$\mathbb{E}[\mathcal{K}_i \mid \mathcal{F}_{i-1}] = \mathcal{K}_{i-1}\mathbb{E}[1 - \nu_i(\lambda)(L_i(\lambda) - R(\lambda)) \mid \mathcal{F}_{i-1}] = \mathcal{K}_{i-1}.$$

In addition, since $\nu_i \in [0, 1]$ and $(L_i(\lambda) - R(\lambda)) \in [-1, 1]$, each component $1 - \nu_i(\lambda)(L_i(\lambda) - R(\lambda)) \geq 0$. Thus, $\{\mathcal{K}_i : i = 1, \ldots, n\}$ is a non-negative martingale with respect to the filtration $\{\mathcal{F}_i : i = 1, \ldots, n\}$. By Ville's inequality,

$$P\left(\max_{i=1,\ldots,n} \mathcal{K}_i \geq \frac{1}{\delta}\right) \leq \delta.$$

On the other hand, since $\nu_i \geq 0$, $\mathcal{K}_i(R; \lambda)$ is increasing in R almost surely for every i. By definition of $\widehat{R}^+_{\mathrm{WSR}}(\lambda)$, if $\widehat{R}^+_{\mathrm{WSR}}(\lambda) < R(\lambda)$, then $P(\max_{i=1,\ldots,n} \mathcal{K}_i \geq 1/\delta)$. Therefore,

$$P\left(\widehat{R}^+_{\mathrm{WSR}}(\lambda) < R(\lambda)\right) \leq P\left(\max_i \mathcal{K}_i \geq \frac{1}{\delta}\right) \leq \delta.$$

This proves that $\widehat{R}^+_{\mathrm{WSR}}(\lambda)$ is a valid upper confidence bound of $R(\lambda)$.

[Proof of Theorem B.5] Define λ^* as in the proof of Theorem D.1. Suppose $R(\hat{\lambda}^{\mathrm{CLT}}) > \gamma$ By the definition of λ^* and the monotonicity and continuity of $R(\cdot)$, this implies $\hat{\lambda}^{\mathrm{CLT}} < \lambda^*$. By the definition of $\hat{\lambda}^{\mathrm{CLT}}$, this further implies that $\widehat{R}^+(\lambda^*) < \gamma$. But

$$\limsup_n P(\widehat{R}^+(\lambda^*) < \gamma) = \delta,$$

by the CLT, which implies the desired result.

[Proof of Theorem C.1] Suppose $R(\mathcal{T}') \leq R(\mathcal{T}_\lambda)$. Write $\rho_x(y)$ for $\rho_x(y; \emptyset)$. Then,

$$\int_{\mathcal{X}} \int_{\mathcal{T}'(x)} \rho_x(y) dy \, dP(x) \geq \int_{\mathcal{X}} \int_{\mathcal{T}_\lambda(x)} \rho_x(y) dy \, dP(x).$$

This further implies

$$\int_{\mathcal{X}} \int_{\mathcal{T}'(x)\setminus\mathcal{T}_\lambda(x)} \rho_x(y)dy\, dP(x) \geq \int_{\mathcal{X}} \int_{\mathcal{T}_\lambda(x)\setminus\mathcal{T}'(x)} \rho_x(y)dy\, dP(x).$$

For $y \in (\mathcal{T}'(x) \setminus \mathcal{T}_\lambda(x))$, we have $\rho_x(y) < \zeta$, whereas for $y \in (\mathcal{T}_\lambda(x) \setminus \mathcal{T}'(x))$ we have $\rho_x(y) \geq \zeta$. Therefore,

$$\int_{\mathcal{X}} \int_{\mathcal{T}'(x)\setminus\mathcal{T}_\lambda(x)} 1dy\, dP(x) \geq \int_{\mathcal{X}} \int_{\mathcal{T}_\lambda(x)\setminus\mathcal{T}'(x)} 1dy\, dP(x),$$

which implies the desired result.

[Proof of Theorem C.2] The proof is similar to that of Theorem C.1. If $R(\mathcal{T}') \leq R(\mathcal{T}_\lambda)$, then

$$\begin{aligned}
&\mathbb{E}\left[\mathbb{E}\left[L(Y;\mathcal{T}'(X)) \mid X\right]\right] \leq \mathbb{E}\left[\mathbb{E}\left[L(Y;\mathcal{T}_\lambda(X)) \mid X\right]\right] \\
\Longrightarrow &\mathbb{E}\left[\mathbb{E}\left[\int_{z\in\mathcal{T}'^c(X)} \ell(Y;z)d\mu(z) \mid X\right]\right] \leq \mathbb{E}\left[\mathbb{E}\left[\int_{z\in\mathcal{T}_\lambda^c(X)} \ell(Y;z)d\mu(z) \mid X\right]\right] \\
\Longrightarrow &\mathbb{E}\left[\mathbb{E}\left[\int_{z\in\mathcal{T}'(X)} \ell(Y;z)d\mu(z) \mid X\right]\right] \geq \mathbb{E}\left[\mathbb{E}\left[\int_{z\in\mathcal{T}_\lambda(X)} \ell(Y;z)d\mu(z) \mid X\right]\right] \\
\Longrightarrow &\mathbb{E}\left[\int_{z\in\mathcal{T}'(X)} \mathbb{E}\left[\ell(Y;z) \mid X\right] d\mu(z)\right] \geq \mathbb{E}\left[\int_{z\in\mathcal{T}_\lambda(X)} \mathbb{E}\left[\ell(Y;z) \mid X\right] d\mu(z)\right] \\
\Longrightarrow &\mathbb{E}\left[\int_{z\in\mathcal{T}'(X)\setminus\mathcal{T}_\lambda(X)} \mathbb{E}\left[\ell(Y;z) \mid X\right] d\mu(z)\right] \geq \mathbb{E}\left[\int_{z\in\mathcal{T}_\lambda(X)\setminus\mathcal{T}'(X)} \mathbb{E}\left[\ell(Y;z) \mid X\right] d\mu(z)\right] \\
\Longrightarrow &\mathbb{E}\left[\int_{z\in\mathcal{T}'(X)\setminus\mathcal{T}_\lambda(X)} -\lambda d\mu(z)\right] \geq \mathbb{E}\left[\int_{z\in\mathcal{T}_\lambda(X)\setminus\mathcal{T}'(X)} -\lambda d\mu(z)\right] \\
\Longrightarrow &\mathbb{E}[|\mathcal{T}'(X) \setminus \mathcal{T}_\lambda(X)|] \geq \mathbb{E}[|\mathcal{T}_\lambda(X) \setminus \mathcal{T}'(X)|] \\
\Longrightarrow &\mathbb{E}[|\mathcal{T}'(X)|] \geq \mathbb{E}[|\mathcal{T}_\lambda(X)|].
\end{aligned}$$

[Proof of Proposition **??**] Let $Z_i = (X_i, Y_i)$ and $\phi(Z_i, Z_j) = L(Y_i, Y_j, \mathcal{T}_\lambda(X_i, X_j))$. First, we apply a representation of U-statistics due to (Hoeffding, 1963) that shows many tail inequalities for sums of i.i.d. random variables hold for U-statistics of order two with an effective sample size $\lfloor n/2 \rfloor$. Specificially, let $m = \lfloor n/2 \rfloor$ and $\pi : \{1, \ldots, n\} \mapsto \{1, \ldots, n\}$ be a uniform random permutation. For each π, define

$$\widehat{R}_\pi(\lambda) = \frac{1}{m} \sum_{j=1}^{m} \phi\left(Z_{\pi(2j-1)}, Z_{\pi(2j)}\right).$$

Note that the summands in $\widehat{R}_\pi(\lambda)$ are independent given π. Then it is not hard to see that

$$\widehat{R}(\lambda) = \mathbb{E}_\pi[\widehat{R}_\pi(\lambda)],$$

where $\mathbb{E}_\pi$ denotes the expectation with respect to π while conditioning on $Z_1, \ldots, Z_n$. By Jensen's inequality, for any convex function ψ,

$$\mathbb{E}[\psi(\widehat{R}(\lambda))] = \mathbb{E}[\phi(\mathbb{E}_\pi[\widehat{R}_\pi(\lambda)])] \leq \mathbb{E}[\mathbb{E}_\pi\psi(\widehat{R}_\pi(\lambda))] = \mathbb{E}_\pi[\mathbb{E}\psi(\widehat{R}_\pi(\lambda))].$$

Since $\widehat{R}_\pi(\lambda)$ has identical distributions for all π,

$$\mathbb{E}[\psi(\widehat{R}(\lambda))] = \mathbb{E}[\psi(\widehat{R}_{\mathrm{id}}(\lambda))] \tag{12}$$

where id is the permutation that maps each element to itself.

For sums of i.i.d. random variables, the Hoeffding's inequality (Proposition B.3) is derived by setting $\psi(z) = \exp\{\nu z\}$ (Hoeffding, 1963), and the Bentkus inequality (Proposition B.4) is derived by setting $\psi(z) = (z - \nu)_+$. Therefore, the same tail probability bounds hold for $\widehat{R}_{\mathrm{id}}(\lambda)$ and thus $\widehat{R}(\lambda)$ by (12). This proves the first two bounds.

To prove the third bound, we apply the technique of (Maurer, 2006) on self-bounding functions of iid random variables. Write $\widehat{R}(\lambda)$ as $U(Z_1, \ldots, Z_n)$ and let

$$U_i = \inf_{z_i} U(Z_1, \ldots, Z_{i-1}, z_i, Z_{i+1}, \ldots, Z_n).$$

Note that U_i is independent of Z_i. Since $\phi(\cdot) \geq 0$, we have

$$0 \leq U - U_i \leq \frac{2}{n(n-1)} \sum_{i \neq j} \phi(Z_i, Z_j).$$

Since $\phi(Z_i, Z_j) \leq 1$,

$$\frac{n}{2}(U - U_i) \leq 1,$$

and

$$\sum_{i=1}^{n} (U - U_i) \leq \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{i \neq j} \phi(Z_i, Z_j) = 2U.$$

If we let $W = (n/2)U$ and $W_i = (n/2)U_i$, then

$$W - W_i \leq 1, \quad \sum_{i=1}^{n} (W - W_i)^2 \leq 2W.$$

In the proof of Theorem 13, (Maurer, 2006) shows that for any $\nu > 0$,

$$\log \mathbb{E}[\exp\{\nu(\mathbb{E}[W] - W)\}] \leq \frac{2\nu G(\nu)}{1 + 2G(\nu)} \mathbb{E}[W].$$

By Markov's inequality, for any $t \in (0, \mathbb{E}[U])$,

$$\begin{aligned} P\left(U \leq t\right) &= P\left(\mathbb{E}[W] - W \geq \mathbb{E}[W] - \frac{n}{2}t\right) \\ &\leq \exp\left\{\min_{\nu > 0} \nu \left(-\mathbb{E}[W] + \frac{n}{2}t + \frac{2G(\nu)}{1 + 2G(\nu)} \mathbb{E}[W]\right)\right\} \\ &= \exp\left\{\min_{\nu > 0} \frac{n\nu}{2} \left(t - \frac{1}{1 + 2G(\nu)} \mathbb{E}[U]\right)\right\}. \end{aligned}$$

The proof is completed by replacing U by $\widehat{R}(\lambda)$ and $\mathbb{E}[U]$ by $R(\lambda)$.

Proposition D.1 (Impossibility of valid UCB for unbounded losses in finite samples) *Let $\mathcal{F}$ be the class of all distributions supported on $[0, \infty)$ with finite mean, and $\mu(F)$ be the mean of the distribution F. Let $\hat{\mu}^+$ be any function of $Z_1, \ldots, Z_n \overset{\text{i.i.d.}}{\sim} F$ such that $P(\hat{\mu}^+ \geq \mu(F)) \geq 1 - \delta$ for any n and $F \in \mathcal{F}$. Then $P(\hat{\mu}^+ = \infty) \geq 1 - \delta$.*

[Proof of Proposition D.1] It is clear that $\mathcal{F}$ satisfies the conditions (i), (ii), and (iii) in (Bahadur & Savage, 1956). For any such $\hat{\mu}^+$, $[0, \hat{\mu}^+]$ is a $(1 - \delta)$ confidence interval of $\mu(F)$. By their Corollary 2, we know that for any $\mu \in \{\mu(F) : F \in \mathcal{F}\}$ and $F \in \mathcal{F}$

$$P_F(\mu \in [0, \hat{\mu}^+]) \geq 1 - \delta \iff P_F(\mu \leq \hat{\mu}^+) \geq 1 - \delta.$$

The proof is completed by letting $\mu \to \infty$.

[Proof of Theorem **??**] This follows from Theorem D.1

[Proof of Theorem **??**] This follows from Theorem D.1

E. An Exact Bound for Binary Loss

When the loss takes values in $\{0, 1\}$, for a fixed λ the loss at each point is a Bernoulli random variable, and the risk is simply the mean of this random variable. In this case, we can give a tight upper confidence bound by simple extracting the relevant quantile of a binomial distribution; see (Brown et al., 2001) for other exact or approximate upper confidence bounds. Explicitly, we have

$$P\left(\widehat{R}(\lambda) \leq t\right) = P\left(\text{Binom}(n, R(\lambda)) \leq \lceil nt \rceil\right),$$

which is the same expression as in the Bentkus bound, improved by a factor of e. From this, we obtain a lower tail probability bound for $\widehat{R}(\lambda)$:

$$g^{\text{bin}}(t; R(\lambda)) \triangleq P\left(\text{Binom}(n, R(\lambda)) \leq \lceil nt \rceil\right).$$

By Proposition B.2, we obtain a $(1 - \delta)$ upper confidence bound for $R(\lambda)$ as

$$\widehat{R}^{+}_{\text{bin}}(\lambda) = \sup\{R : g^{\text{bin}}(\widehat{R}(\lambda); R) \geq \delta\}.$$

We obtain $\hat{\lambda}^{\text{bin}}$ by inverting the above bound computationally, yielding the following corollary:

Theorem E.1 (RCPS for binary variables) *In the setting of Theorem 1, assume additionally that the loss takes values in {0,1}. Then, $\mathcal{T}_{\hat{\lambda}^{\text{bin}}}$ is a (γ, δ)-RCPS.*

The binary loss case case results in a classical tolerance region, as discussed previously in (Vovk, 2012) and (Park et al., 2020).

F. Further Comparisons of Upper Confidence Bounds

We present additional plots comparing the upper confidence bounds with $\delta = 0.01$ and $\delta = 0.001$. The counterparts of Figure 13 for bounded cases are presented in Figure 20 and 23, and the counterparts of Figure 16 for unbounded cases are presented in Figure 26 and 29.

To further compare the HB bound and WSR bound for the binary loss case, in Figure 30 we present the fraction of samples on which the HB bound or the WSR bound is the winner among the four bounds, excluding the CLT bound due to the undercoverage. The HB bound is more likely to be tighter than the WSR bound, especially when the mean μ or the level δ is small. Moreover, the symmetry between two curves in each panel is due to the fact that the simple Hoeffding bound and empirical Bernstein bound never win. These results show that the WSR better is not uniformly better than the HB bound, although it is still the best all-around choice for bounded losses.

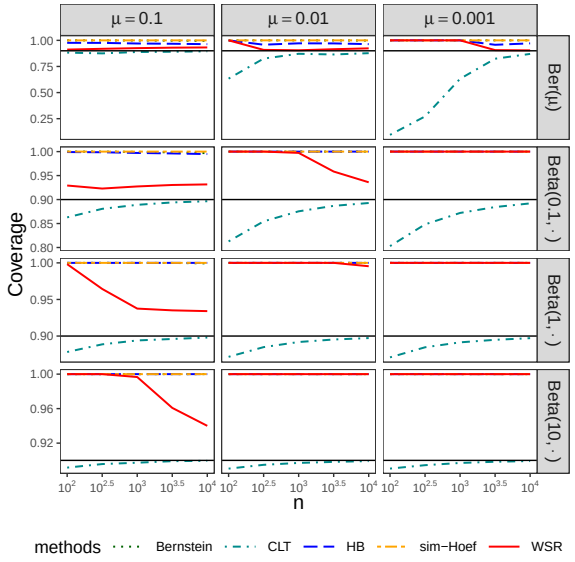

Figure 18. Coverage $P(\widehat{R}(\lambda) \geq R(\lambda))$

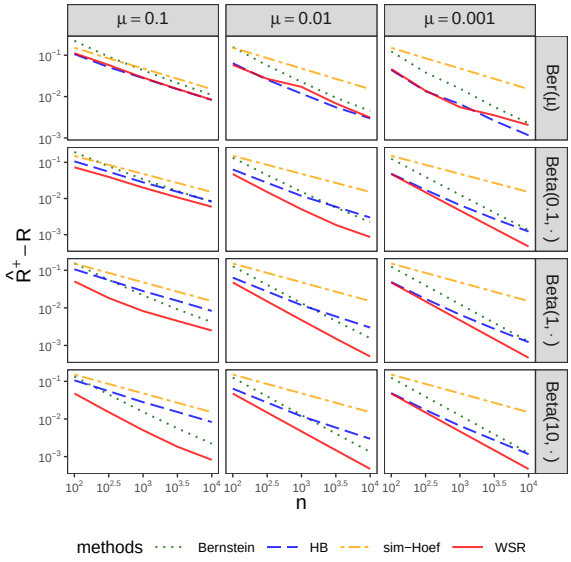

Figure 19. Median of $\widehat{R}(\lambda) - R(\lambda)$

Figure 20. Numerical evaluations of the simple Hoeffding bound (5), HB bound (7), empirical Bernstein bound (8), CLT bound (10), and WSR bound (Proposition B.5) on a million independent samples of size n with $\delta = 0.01$. Each row corresponds to a type of distribution and each column corresponds to a value of the mean. The CLT bound is excluded in (b) because it does not achieve the target coverage in most of the cases.

G. Adaptive Score Renormalization for Polyp Segmentation

This section describes in detail the construction of our predictor in the polyp segmentation example in Section A.3. In order to construct a good set predictor from the raw predictor, we draw on techniques from the classical literature on image processing to detect and emphasize local peaks in the raw scores. In particular, we construct a renormalization function $r : [0,1]^{m \times n} \to [0,1]^{m \times n}$, which is a composition of a set of morphological operations. We will now list a set of operations whose composition will define r.

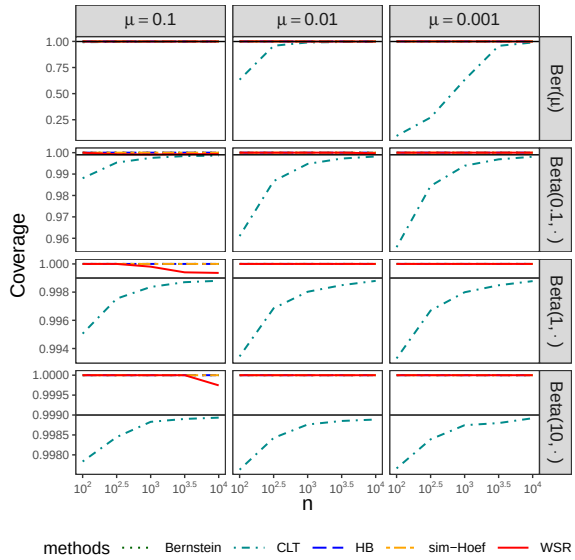

Figure 21. Coverage $P(\widehat{R}(\lambda) \geq R(\lambda))$

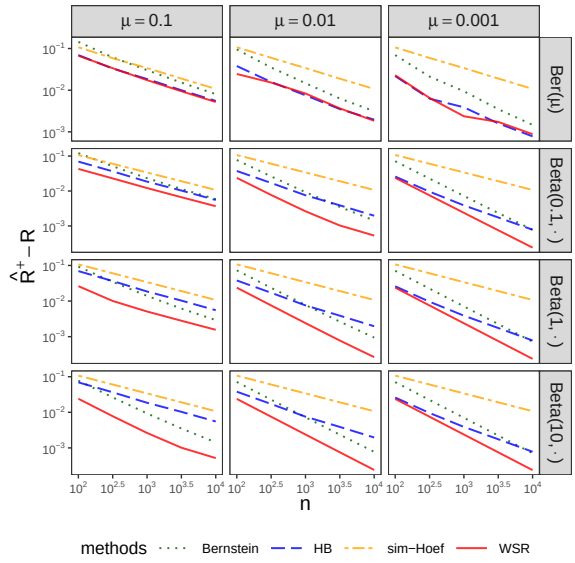

Figure 22. Median of $\widehat{R}(\lambda) - R(\lambda)$

Figure 23. Numerical evaluations of the simple Hoeffding bound (5), HB bound (7), empirical Bernstein bound (8), CLT bound (10), and WSR bound (Proposition B.5) on a million independent samples of size n with $\delta = 0.001$. Each row corresponds to a type of distribution and each column corresponds to a value of the mean. The CLT bound is excluded in (b) because it does not achieve the target coverage in most of the cases.

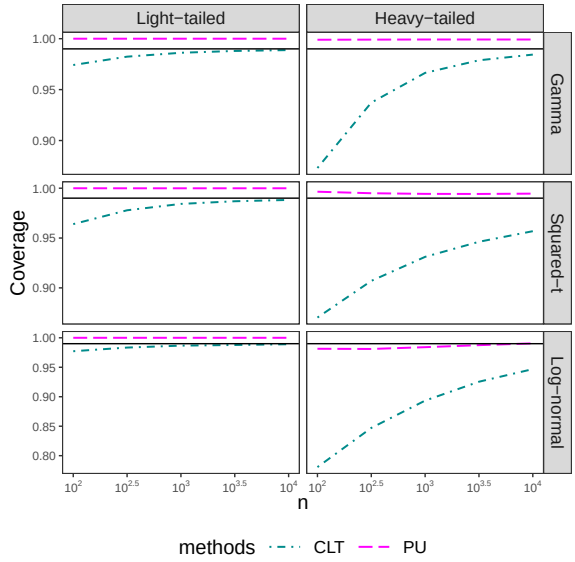

Figure 24. Coverage $P(\widehat{R}(\lambda) \geq R(\lambda))$

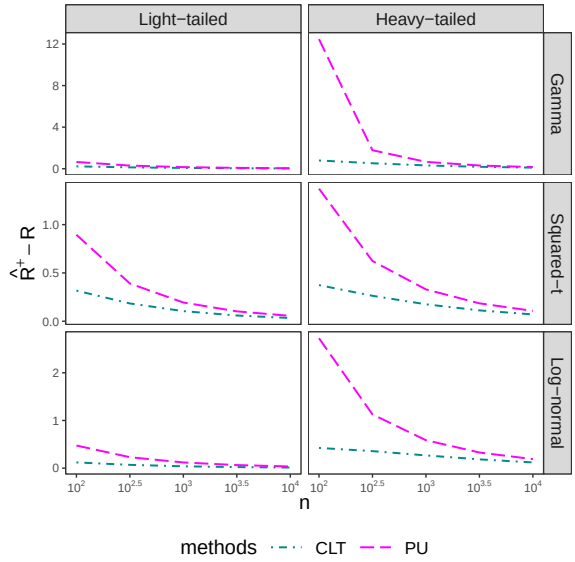

Figure 25. Median of $\widehat{R}(\lambda) - R(\lambda)$

Figure 26. Numerical evaluations of the PU bound (9) with the estimated coefficient of variation and the CLT bound (10), on a million independent samples of size n from each distribution in Table 1 with $\delta = 0.01$. Each row corresponds to a type of distribution and each column corresponds to a value of the mean.

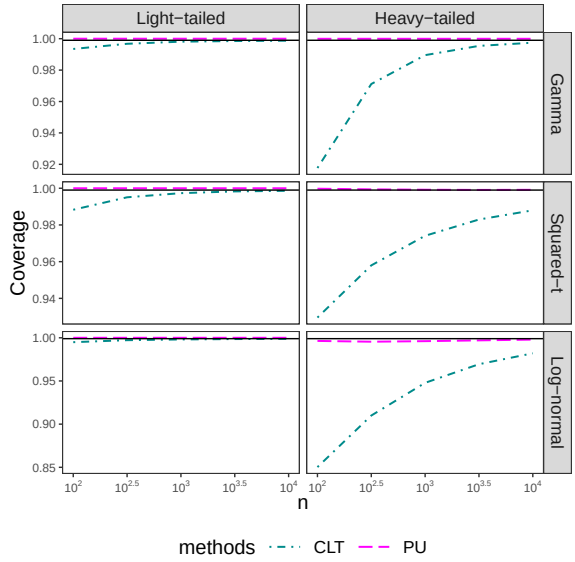

Figure 27. Coverage $P(\widehat{R}(\lambda) \geq R(\lambda))$

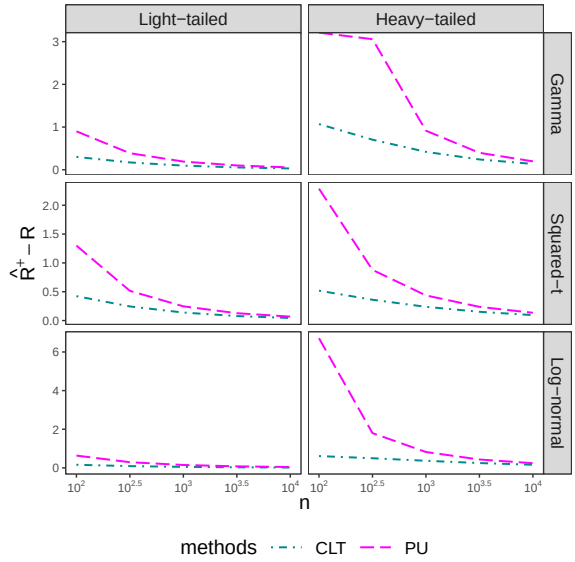

Figure 28. Median of $\widehat{R}(\lambda) - R(\lambda)$

Figure 29. Numerical evaluations of the PU bound (9) with the estimated coefficient of variation and the CLT bound (10), on a million independent samples of size n from each distribution in Table 1 with $\delta = 0.001$. Each row corresponds to a type of distribution and each column corresponds to a value of the mean.

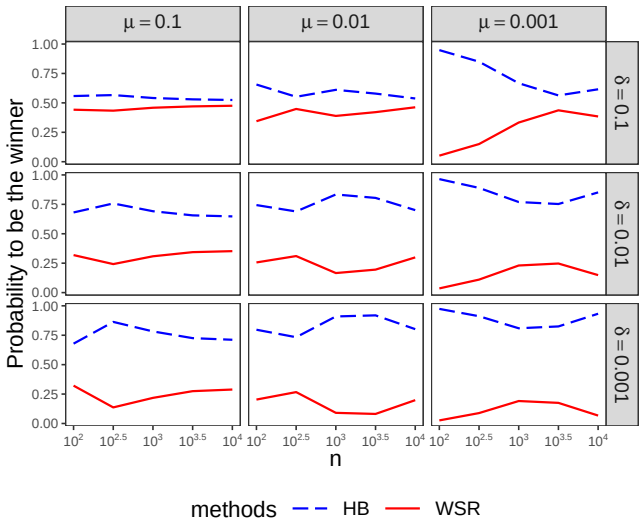

Figure 30. Fraction of samples on which the HB bound or the WSR bound is the winner among the four bounds, excluding the CLT bound, for Bernoulli distributions. Each row corresponds to a level and each column corresponds to a value of mean.

Define the discrete Gaussian blur operator as $g : [0,1]^{m\times n} \times \mathbb{R}_{++} \times \mathbb{O}_+ \to [0,1]^{m\times n}$, where $\mathbb{O}_+$ is the set of odd numbers. The second argument to g is the standard deviation σ of a Gaussian kernel in pixels and k is the side length of the kernel in pixels. The Gaussian kernel is then the matrix

$$K(\sigma,k)_{i,j} = C \exp\left\{-\frac{1}{2\sigma^2}\|[i,j] - [\lceil k/2\rceil, \lceil k/2\rceil]\|^2\right\},$$

where C is chosen such that $\sum_{i,j} K_{i,j} = 1$. The function g then becomes $g(S,\sigma,k) = S * K(\sigma,k)$, where $*$ denotes the 2D convolution operator.

We borrow a technique from mathematical morphology called *reconstruction by dilation* and use it to separate local score peaks from their background. We point the reader to Robinson and Whelan (Robinson & Whelan, 2004) for an involved description of the algorithm we applied in our codebase. For the purposes of this paper, we write the reconstruction by dilation algorithm as $dil : [0,1]^{m\times n} \to [0,1]^{m\times n}$. The output of dil is an array containing only the local peaks from the input, with all other areas set to zero.

Define the binarization function $bin_t : [0,1]^{m\times n} \to \{0,1\}^{m\times n}$ as $bin(x)_{i,j} = \bot$.

In the next step, we binarize the local peaks and then split them into disjoint regions through the 2-connected-components function $conn : \{0,1\}^{m\times n} \to 2^{\{0,1\}^{m\times n}}$. Viewing a binary matrix M as a graph, we can express it as an adjacency matrix $A \in mn \times mn$ where

$$\begin{aligned} A(M)_{i,j} = \{\|[\lfloor i/n\rfloor, \mod(i,n)] - [\lfloor j/n\rfloor, \mod(j,n)]\| < 2 \\ M_{\lfloor i/n\rfloor, \mod(i,n)} = 1 \\ M_{\lfloor j/n\rfloor, \mod(j,n)} = 1\}. \end{aligned}$$

In words, each entry of A corresponds to a pixel, and two pixels are connected by an edge if and only if they are adjacent with entry 1 in the matrix M. We can use A to define a function $isconnected : \{0,1\}^{m\times n} \times m \times n \times m \times n \to \{0,1\}$ that takes a binary matrix M and two coordinates (i,j) and (i',j') and returns 1 if the coordinates are connected by a path. Explicitly, $isconnected = \{\exists k \ : \ A^k_{ni+j,ni'+j'} = 1\}$. Since $isconnected$ is reflexive, symmetric, and transitive, it defines an equivalence relation $\sim$. We can formally define the set of all equivalence classes over indexes,

$$\mathcal{E}(A) = \{\{(i,j) \in m \times n : (i,j) \sim (i',j')\} : (i',j') \in m \times n\}.$$

Using $\mathcal{E}$, we can draw bounding boxes around each object as

$$bboxes(\mathcal{E}) = \{[\inf\{i : (i,j) \in E \text{ for some } j\}, \sup\{i : (i,j) \in E \text{ for some } j\}] \times \\ [\inf\{j : (i,j) \in E \text{ for some } i\}, \sup\{j : (i,j) \in E \text{ for some } i\}] : E \in \mathcal{E}\}$$

We can proceed to define a function $renorm$ that takes in a matrix of scores M and a set of bounding boxes $bboxes$ and returns a renormalized matrix of scores:

$$renorm(M, bboxes)_{i,j} = \frac{M_{i,j}}{\min\limits_{b \in bboxes} \max\limits_{\substack{(i',j') \in b \\ (i,j) \in b}} M_{i',j'}}.$$

We can finally define r as $r(M) = renorm(M, bboxes(\mathcal{E}(A(bin_t(g(M, \sigma, k))))))$ for use in Equation A.3.