Conformal Risk Control

Anastasios N. Angelopoulos[1], Stephen Bates[1], Adam Fisch[2], Lihua Lei[3], and Tal Schuster[4]

[1]University of California, Berkeley
[2]Massachusetts Institute of Technology
[3]Stanford University
[4]Google Research

Abstract

We extend conformal prediction to control the expected value of any monotone loss function. The algorithm generalizes split conformal prediction together with its coverage guarantee. Like conformal prediction, the conformal risk control procedure is tight up to an $\mathcal{O}(1/n)$ factor. Worked examples from computer vision and natural language processing demonstrate the usage of our algorithm to bound the false negative rate, graph distance, and token-level F1-score.

1 Introduction

We seek to endow some pre-trained machine learning model with guarantees on its performance as to ensure its safe deployment. Suppose we have a base model f that is a function mapping inputs $x \in \mathcal{X}$ to values in some other space, such as a probability distribution over classes. Our job is to take the output of f and post-process it into quantities with statistical guarantees.

Split conformal prediction [1, 2], which we will henceforth refer to simply as "conformal prediction," has been useful in areas such as computer vision [3] and natural language processing [4] to provide such a guarantee. By measuring the model's performance on a *calibration dataset* $\{(X_i, Y_i)\}_{i=1}^{n}$ of feature-response pairs, conformal prediction post-processes the model to construct prediction sets that bound the *miscoverage*,

$$\mathbb{P}\Big(Y_{n+1} \notin \mathcal{C}(X_{n+1})\Big) \leq \alpha, \tag{1}$$

where (X_{n+1}, Y_{n+1}) is a new test point, α is a user-specified error rate like 10%, and $\mathcal{C}$ is a function of the model and calibration data that outputs a prediction set. Note that $\mathcal{C}$ is formed using the first n data points, and the probability in (1) is over the randomness in all $n+1$ data points.

We extend conformal prediction to prediction tasks where the natural notion of error is not miscoverage. In particular, our main result is that a generalization of conformal prediction provides guarantees of the form

$$\mathbb{E}\Big[\ell\big(\mathcal{C}(X_{n+1}), Y_{n+1}\big)\Big] \leq \alpha, \tag{2}$$

for any bounded *loss function* ℓ that shrinks as $\mathcal{C}(X_{n+1})$ grows. We call this a *conformal risk control* guarantee. Note that (2) recovers the conformal miscoverage guarantee in (1) when using the miscoverage loss, $\ell\big(\mathcal{C}(X_{n+1}), Y_{n+1}\big) = \mathbb{1}\left\{Y_{n+1} \notin \mathcal{C}(X_{n+1})\right\}$. However, our algorithm also extends conformal prediction to situations where other loss functions, such as the false negative rate (FNR) or F1-score, are more appropriate.

As an example, consider multilabel classification, where the $Y_i \subseteq \{1, ..., K\}$ are sets comprising a subset of K classes. Given a trained multilabel classifier $f : \mathcal{X} \to [0,1]^K$, we want to output sets that include a large fraction of the true classes in Y_i. To that end, we post-process the model's raw outputs into the set of classes with sufficiently high scores, $\mathcal{C}_\lambda(x) = \{k : f(X)_k \geq 1 - \lambda\}$. Note that for a fixed x, as the threshold λ grows, we include more classes in $\mathcal{C}_\lambda(x)$—it becomes more conservative. Conformal risk control finds a threshold value $\hat{\lambda}$ that controls the fraction of missed classes, i.e., the expected value of $\ell\big(\mathcal{C}_{\hat{\lambda}}(X_{n+1}), Y_{n+1}\big) = 1 - |Y_{n+1} \cap \mathcal{C}_\lambda(X_{n+1})|/|Y_{n+1}|$. Setting $\alpha = 0.1$ would ensure that $\mathcal{C}_{\hat{\lambda}}(X_{n+1})$ contains 90% of the true classes in Y_{n+1} on average.

1.1 Algorithm and preview of main results

Formally, we will consider post-processing the predictions of the model f to create a function $\mathcal{C}_\lambda(\cdot)$. The function has a parameter λ that encodes its level of conservativeness: larger λ values yield more conservative outputs (e.g., larger prediction sets). To measure the quality of the output of $\mathcal{C}_\lambda$, we consider a loss function $\ell(\mathcal{C}_\lambda(x), y) \in (-\infty, B]$ for some $B < \infty$. We require the loss function to be non-increasing as a function of λ. Our goal is to choose $\hat{\lambda}$ based on the observed data $\{(X_i, Y_i)\}_{i=1}^n$ so that risk control as in (2) holds.

We now rewrite this same task in a more notationally convenient and abstract form. Consider an exchangeable collection of non-increasing, random functions $L_i : \Lambda \to (-\infty, B]$, $i = 1, \ldots, n+1$. We seek to use the first n functions to choose a value of the parameter, $\hat{\lambda}$, in such a way that the risk on the unseen function is controlled:

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big)\Big] \leq \alpha. \tag{3}$$

We are primarily motivated by the case where $L_i(\lambda) = \ell(\mathcal{C}_\lambda(X_i), Y_i)$, in which case the guarantee in (3) coincides with risk control as in (2).

Now we describe the algorithm. Let $\widehat{R}_n(\lambda) = (L_1(\lambda) + \ldots + L_n(\lambda))/n$. Given any desired risk level upper bound $\alpha \in (-\infty, B)$, define

$$\hat{\lambda} = \inf\left\{\lambda : \frac{n}{n+1}\widehat{R}_n(\lambda) + \frac{B}{n+1} \leq \alpha\right\}. \tag{4}$$

Our proposed *conformal risk control* algorithm is to deploy $\hat{\lambda}$ on the forthcoming test point. Our main result is that this algorithm satisfies (3). When the L_i are i.i.d. from a continuous distribution, the algorithm satisfies a tight lower bound saying it is not too conservative,

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big)\Big] \geq \alpha - \frac{2B}{n+1}.$$

We show the reduction from conformal risk control to conformal prediction in Section 2.3. Finally, if the risk is non-monotone, then this algorithm does not control the risk; we discuss this in Section 2.4.

1.2 Related work

Conformal prediction was developed by Vladimir Vovk and collaborators beginning in the late 1990s [1, 5]. See [6] for a modern introduction to the area or [7] for a more classical alternative. We primarily build on *split conformal prediction* [2]; statistical properties of this algorithm including the coverage upper bound were studied in [8]. Recently there have been many extensions of the conformal algorithm, mainly targeting deviations from exchangeability [9–12] and improved conditional coverage [3, 13–16]. Most relevant to us is recent work on risk control in high probability [17–19] and its applications [20–26]. Though these works closely relate to ours in terms of motivation, the algorithm presented herein differs greatly: it has a guarantee in expectation, and neither the algorithm nor its analysis share much technical similarity with these previous works.

2 Theory

2.1 Risk control

We first show that the proposed algorithm leads to risk control when the loss is monotone.

Theorem 1. *Assume that $L_i(\lambda)$ is non-increasing in λ, right-continuous, and*

$$\inf_\lambda L_i(\lambda) \leq \alpha - \frac{B}{n+1}, \quad \sup_\lambda L_i(\lambda) \leq B < \infty \text{ almost surely.} \tag{5}$$

Then

$$\mathbb{E}[L_{n+1}(\hat{\lambda})] \leq \alpha.$$

Proof. Let $\widehat{R}_{n+1}(\lambda) = (L_1(\lambda) + \ldots + L_{n+1}(\lambda))/(n+1)$ and

$$\hat{\lambda}' = \inf\left\{\lambda : \widehat{R}_{n+1}(\lambda) \leq \alpha\right\}.$$

Since $\inf_\lambda L_i(\lambda) \leq \alpha - \frac{B}{n+1}$, $\hat{\lambda}'$ and $\hat{\lambda}$ are both well-defined almost surely. Since $L_{n+1}(\lambda) \leq B$, we know $\widehat{R}_{n+1}(\lambda) = \frac{n}{n+1}\widehat{R}_n(\lambda) + \frac{L_{n+1}(\lambda)}{n+1} \leq \frac{n}{n+1}\widehat{R}_n(\lambda) + \frac{B}{n+1}$. Thus,

$$\frac{n}{n+1}\widehat{R}_n(\lambda) + \frac{B}{n+1} \leq \alpha \Longrightarrow \widehat{R}_{n+1}(\lambda) \leq \alpha.$$

This implies $\hat{\lambda}' \leq \hat{\lambda}$. Since $L_i(\lambda)$ is non-increasing in λ,

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big)\Big] \leq \mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}'\big)\Big]. \tag{6}$$

Let E be the multiset of loss functions $\{L_1, \ldots, L_{n+1}\}$. Then $\hat{\lambda}'$ is a function of E, or, equivalently, $\hat{\lambda}'$ is a constant conditional on E. Additionally, $L_{n+1}(\lambda)|E \sim \text{Uniform}(\{L_1, ..., L_{n+1}\})$ by exchangeability. These facts combined with the right-continuity of L_i imply

$$\mathbb{E}\left[L_{n+1}(\hat{\lambda}') \mid E\right] = \frac{1}{n+1}\sum_{i=1}^{n+1} L_i(\hat{\lambda}') \leq \alpha.$$

The proof is completed by the law of total expectation and (6). □

2.2 A tight risk lower bound

Next we show that the conformal risk control procedure is tight up to a factor $2B/(n+1)$ that cannot be improved in general. Like the standard conformal coverage upper bound, the proof will rely on a form of continuity that prohibits large jumps in the risk function. Towards that end, we will define the *jump function* below, which quantifies the size of the discontinuity in a right-continuous input function l at point λ:

$$J(l, \lambda) = \lim_{\epsilon \to 0^+} l(\lambda - \epsilon) - l(\lambda)$$

The jump function measures the size of a discontinuity at $l(\lambda)$. When there is a discontinuity and l is non-increasing, $J(l, \lambda) > 0$. When there is no discontinuity, the jump function is zero. The next theorem will assume that the probability that L_i has a discontinuity at any pre-specified λ is $\mathbb{P}(J(L_i, \lambda) > 0) = 0$. Under this assumption the conformal risk control procedure is not too conservative.

Theorem 2. *In the setting of Theorem 1, further assume that the L_i are i.i.d, $L_i \geq 0$, and for any λ, $\mathbb{P}(J(L_i, \lambda) > 0) = 0$. Then*

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big)\Big] \geq \alpha - \frac{2B}{n+1}.$$

This bound is tight for general monotone loss functions, as we show next.

Proposition 1. *In the setting of Theorem 2, for any $\epsilon > 0$, there exists a loss function and $\alpha \in (0, 1)$ such that*

$$\mathbb{E}\left[L_{n+1}\big(\hat{\lambda}\big)\right] \leq \alpha - \frac{2B - \epsilon}{n+1}.$$

Since we can take ϵ arbitrarily close to zero, we conclude that the factor $2B/(n+1)$ in Theorem 2 is required in the general case.

2.3 Conformal prediction is a special case of conformal risk control

Conformal prediction can be thought of as controlling the expectation of an indicator loss function. Recall that the risk upper bound (2) specializes to the conformal coverage guarantee in (1) when the loss function is the indicator of a miscoverage event. The conformal risk control procedure specializes to conformal prediction under this loss function as well. However, the risk lower bound in Theorem 2 has a slightly worse constant than the usual conformal guarantee. We now describe these correspondences.

First, we show the equivalence of the algorithms. In conformal prediction, we have conformal scores $s(X_i, Y_i)$ for some score function $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. Based on this score function, we create prediction sets for the test point X_{n+1} as

$$\mathcal{C}_{\hat{\lambda}}(X_{n+1}) = \left\{y : s(X_{n+1}, y) \leq \hat{\lambda}\right\},$$

where $\hat{\lambda}$ is the conformal quantile, a parameter that is set based on the calibration data. In particular, conformal prediction chooses $\hat{\lambda}$ to be the $\lceil (n+1)(1-\alpha) \rceil / n$ sample quantile of $\{s(X_i, Y_i)\}_{i=1}^n$. To formulate this in the language of risk control, we consider a *miscoverage loss* $L_i^{\text{Cvg}}(\lambda) = \mathbb{1}\left\{Y_i \notin \widehat{\mathcal{C}}_\lambda(X_i)\right\} = \mathbb{1}\left\{s(X_i, Y_i) > \lambda\right\}$. Direct calculation of $\hat{\lambda}$ from (4) then shows the equivalence of the proposed procedure to conformal prediction:

$$\hat{\lambda} = \inf\left\{\lambda : \frac{1}{n+1}\sum_{i=1}^{n} \mathbb{1}\left\{s(X_i, Y_i) > \lambda\right\} + \frac{1}{n+1} \leq \alpha\right\} = \underbrace{\inf\left\{\lambda : \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\left\{s(X_i, Y_i) \leq \lambda\right\} \geq \frac{\lceil (n+1)(1-\alpha) \rceil}{n}\right\}}_{\text{conformal prediction algorithm}}.$$

Next, we discuss how the risk lower bound relates to its conformal prediction equivalent. In the setting of conformal prediction, [8] proves that $\mathbb{P}(Y_{n+1} \notin \mathcal{C}_{\hat{\lambda}}(X_{n+1})) \geq \alpha - 1/(n+1)$ when the conformal score function follows a continuous distribution. Theorem 2 recovers this guarantee with a slightly worse constant: $\mathbb{P}(Y_{n+1} \notin \mathcal{C}_{\hat{\lambda}}(X_{n+1})) \geq \alpha - 2/(n+1)$. First, note that our assumption in Theorem 2 about the distribution of discontinuities specializes to the continuity of the score function when the miscoverage loss is used:

$$\mathbb{P}\left(J\left(L_i^{\text{Cvg}}, \lambda\right) > 0\right) = 0 \iff \mathbb{P}(s(X_i, Y_i) = \lambda) = 0.$$

However, the bound for the conformal case is better than the bound for the general case in Theorem 2 by a factor of two, which cannot be improved according to Proposition 1. The fact that conformal prediction has a slightly tighter lower bound than conformal risk control is an interesting oddity of the binary loss function; however, it is of little practical importance, as the difference between $1/(n+1)$ and $2/(n+1)$ is small even for moderate values of n.

2.4 Controlling general loss functions

We next show that the conformal risk control algorithm does *not* control the risk if the L_i are not assumed to be monotonic. In particular, (3) does not hold. We show this by example.

Proposition 2. *For any ϵ, there exists a non-monotone loss function such that*

$$\mathbb{E}\left[L_{n+1}\left(\hat{\lambda}\right)\right] \geq 1 - \epsilon.$$

Notice that for any desired level α, the expectation in (3) can be arbitrarily close to 1. Since the function values here are in $[0, 1]$, this means that even for bounded random variables, risk control can be violated by an arbitrary amount unless further assumptions are placed on the L_i.

The algorithms developed may still be appropriate for near-monotone loss functions. Simply 'monotonizing' all loss functions L_i and running conformal risk control will guarantee (3), but this strategy will only

be powerful if the loss is near-monotone. For concreteness, we describe this procedure below as a corollary of Theorem 1.

Corollary 1. *Allow $L_i(\lambda)$ to be any (possibly non-monotone) function of λ satisfying 5. Take*

$$\tilde{L}_i(\lambda) = \sup_{\lambda' \geq \lambda} L_i(\lambda'), \quad \tilde{R}_n(\lambda) = \frac{1}{n}\sum_{i=1}^{n} \tilde{L}_i(\lambda), \quad \textit{and } \tilde{\lambda} = \inf\left\{\lambda : \frac{n}{n+1}\tilde{R}_n(\lambda) + \frac{B}{n+1} \leq \alpha\right\}.$$

Then,

$$\mathbb{E}\left[L_{n+1}(\tilde{\lambda})\right] \leq \alpha.$$

If the loss function is already monotone, then $\tilde{\lambda}$ reduces to $\hat{\lambda}$. We propose a further algorithm for picking λ in Appendix A that provides an asymptotic risk-control guarantee for *non-monotone* loss functions. However, this algorithm again is only powerful when the risk $\mathbb{E}[L_{n+1}(\lambda)]$ is near-monotone and reduces to the standard conformal risk control algorithm when the loss is monotone.

3 Examples

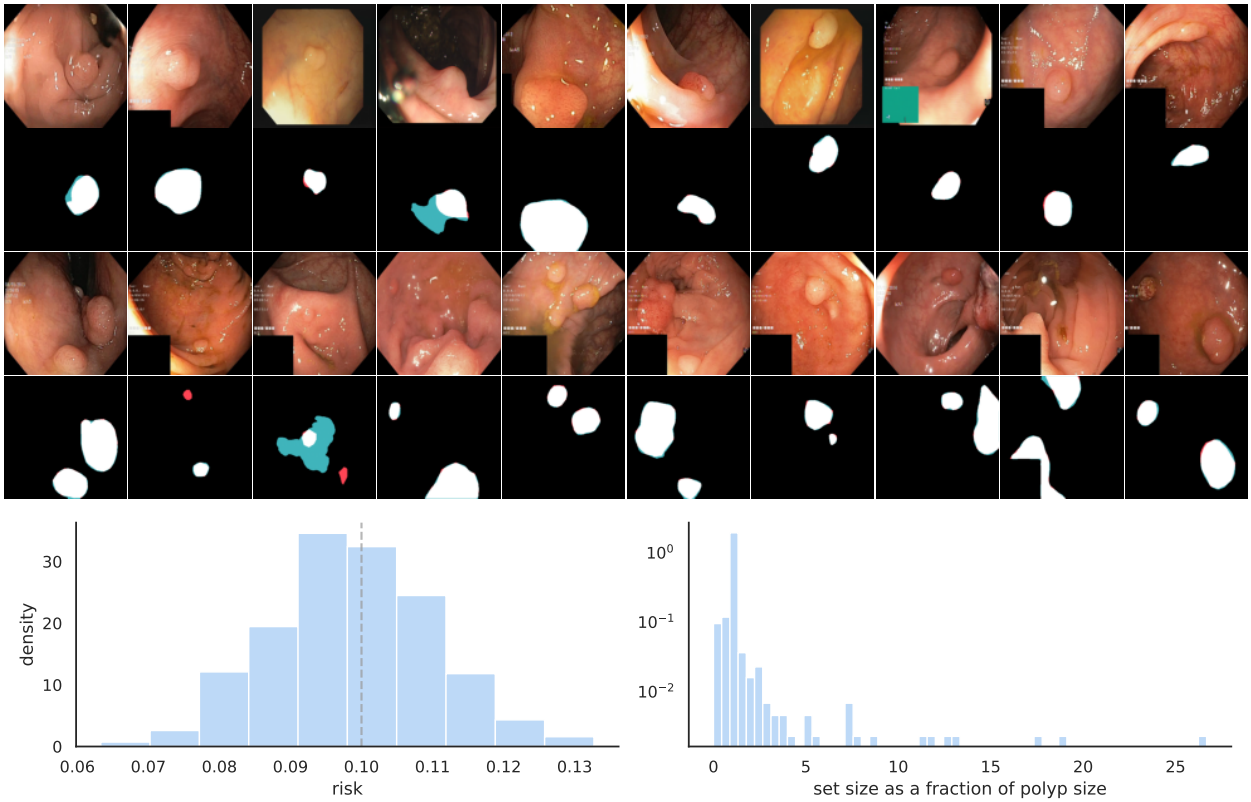

Figure 1: **FNR control in tumor segmentation**. The top figure shows examples of our procedure with correct pixels in white, false positives in blue, and false negatives in red. The bottom plots report FNR and set size over 1000 independent random data splits. The dashed gray line marks α.

To demonstrate the flexibility and empirical effectiveness of the proposed algorithm, we apply it to four tasks across computer vision and natural language processing. All four loss functions are non-binary, monotone losses bounded by 1. They are commonly used within their respective application domains. Our results validate that the procedure bounds the risk as desired and gives useful outputs to the end-user. Code to reproduce our examples is available at `https://github.com/aangelopoulos/conformal-risk`.

3.1 FNR control in tumor segmentation

In the tumor segmentation setting, our input is a $d \times d$ image and our label is a set of pixels $Y_i \in \wp(\{(1,1),(1,2),...,(d,d)\})$, where $\wp$ denotes the power set. We build on an image segmentation model $f : \mathcal{X} \to [0,1]^{d \times d}$ outputting a probability for each pixel and measure loss as the number of false positives,

$$L_i^{\text{FNR}}(\lambda) = 1 - \frac{|Y_i \cap \mathcal{C}_\lambda(X_i)|}{|Y_i|}, \text{ where } \mathcal{C}_\lambda(X_i) = \{y : f(X_i)_y \geq 1 - \lambda\}. \tag{7}$$

The expected value of L_i^{FNR} is the FNR. Since L_i^{FNR} is monotone, so is the FNR. Thus, we use the technique in Section 2.1 to pick $\hat{\lambda}$ by (4) that controls the FNR on a new point, resulting in the following guarantee:

$$\mathbb{E}\left[L_{n+1}^{\text{FNP}}(\hat{\lambda})\right] \leq \alpha. \tag{8}$$

For evaluating the proposed procedure we pool data from several online open-source gut polyp segmentation datasets: Kvasir, Hyper-Kvasir, CVC-ColonDB, CVC-ClinicDB, and ETIS-Larib. We choose a PraNet [27] as our base model f and used $n = 1000$, and evaluated risk control with the 781 remaining validation data points. We report results with $\alpha = 0.1$ in Figure 1. The mean and standard deviation of the risk over 1000 trials are 0.0987 and 0.0114, respectively.

3.2 FNR control in multilabel classification

Figure 2: **FNR control on MS COCO**. The top figure shows examples of our procedure with correct classes in black, false positives in blue, and false negatives in red. The bottom plots report FNR and set size over 1000 independent random data splits. The dashed gray line marks α.

In the multilabel classification setting, our input X_i is an image and our label is a set of classes $Y_i \subset \{1,\ldots,K\}$ for some number of classes K. Using a multiclass classification model $f : \mathcal{X} \to [0,1]^K$, we form

prediction sets and calculate the number of false positives exactly as in (7). By Theorem 1, picking $\hat{\lambda}$ as in (4) again yields the FNR-control guarantee in (8).

We use the Microsoft Common Objects in Context (MS COCO) computer vision dataset [28], a large-scale 80-class multiclass classification baseline dataset commonly used in computer vision, to evaluate the proposed procedure. We choose a TResNet [29] as our base model f and used $n = 4000$, and evaluated risk control with 1000 validation data points. We report results with $\alpha = 0.1$ in Figure 2. The mean and standard deviation of the risk over 1000 trials are 0.0996 and 0.0052, respectively.

3.3 Control of graph distance in hierarchical image classification

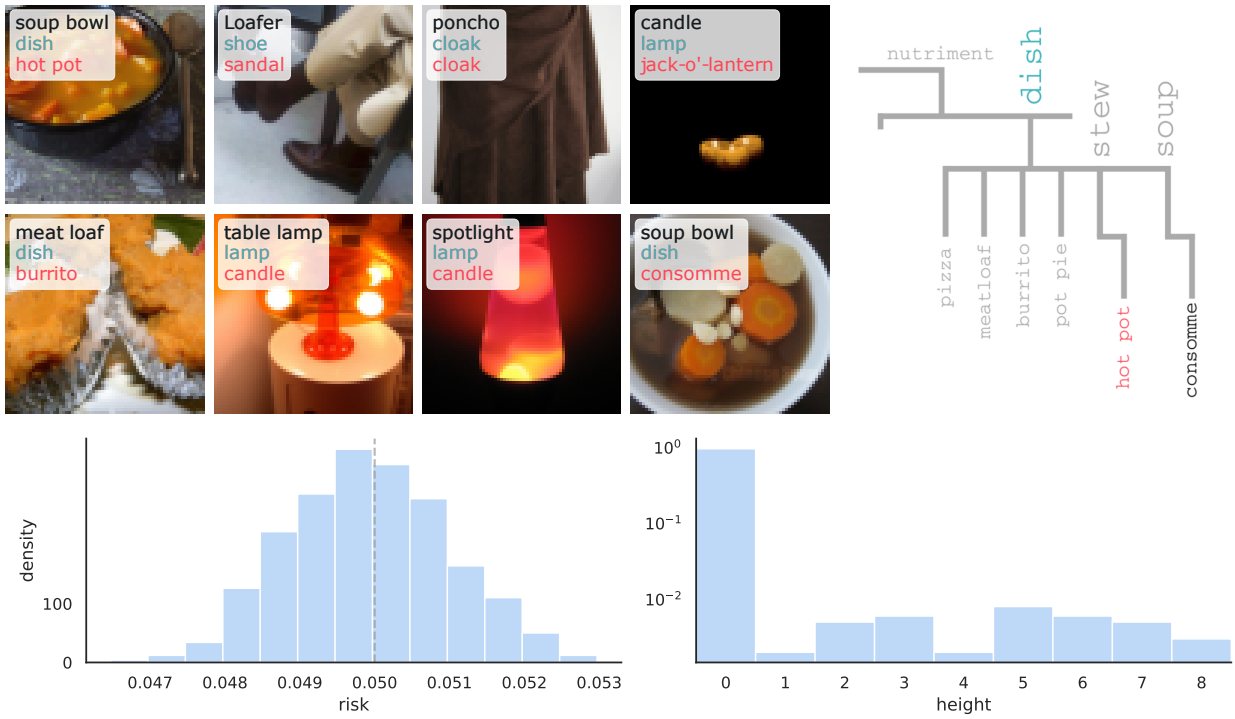

Figure 3: **Control of graph distance on hierarchical ImageNet**. The top figure shows examples of our procedure with correct classes in black, false positives in blue, and false negatives in red. The bottom plots report FDR and set size over 1000 independent random data splits. The dashed gray line marks α.

In the K-class hierarchical classification setting, our input X_i is an image and our label is a leaf node $Y_i \in \{1, ..., K\}$ on a tree with nodes $\mathcal{V}$ and edges $\mathcal{E}$. Using a single-class classification model $f : \mathcal{X} \to \Delta^K$, we calibrate a loss in graph distance between the interior node we select and the closest ancestor of the true class. For any $x \in \mathcal{X}$, let $\hat{y}(x) = \arg\max_k f(x)_k$ be the class with the highest estimated probability. Further, let $d : \mathcal{V} \times \mathcal{V} \to \mathbb{Z}$ be the function that returns the length of the shortest path between two nodes, let $\mathcal{A} : \mathcal{V} \to 2^{\mathcal{V}}$ be the function that returns the ancestors of its argument, and let $\mathcal{P} : \mathcal{V} \to 2^{\mathcal{V}}$ be the function that returns the set of leaf nodes that are descendants of its argument. We also let $g(v, x) = \sum_{k \in \mathcal{P}(v)} f(x)_k$ be the sum of scores of leaves descended from v. Further, define a hierarchical distance

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

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

$$L_i^{\text{Graph}}(\lambda) = \inf_{s \in \mathcal{C}_\lambda(X_i)} \{d_H(y, s)\}/D, \text{ where } \mathcal{C}_\lambda(x) = \bigcap_{\{a \in \mathcal{A}(\hat{y}(x)) \,:\, g(a,x) \geq -\lambda\}} \mathcal{P}(a).$$

This loss returns zero if y is a child of any element in $\mathcal{C}_\lambda$, and otherwise returns the minimum distance between any element of $\mathcal{C}_\lambda$ and any ancestor of y, scaled by the depth D. Thus, it is a monotone loss function and can be controlled by choosing $\hat{\lambda}$ as in (4) to achieve the guarantee

$$\mathbb{E}\left[L_{n+1}^{\text{Graph}}(\hat{\lambda})\right] \leq \alpha.$$

For this experiment, we use the ImageNet dataset [30], which comes with an existing label hierarchy, WordNet, of maximum depth $D = 14$. We choose a ResNet152 [31] as our base model f and used $n = 30000$, and evaluated risk control with the remaining 20000. We report results with $\alpha = 0.05$ in Figure 3. The mean and standard deviation of the risk over 1000 trials are 0.0499 and 0.0011, respectively.

3.4 F1-score control in open-domain question answering

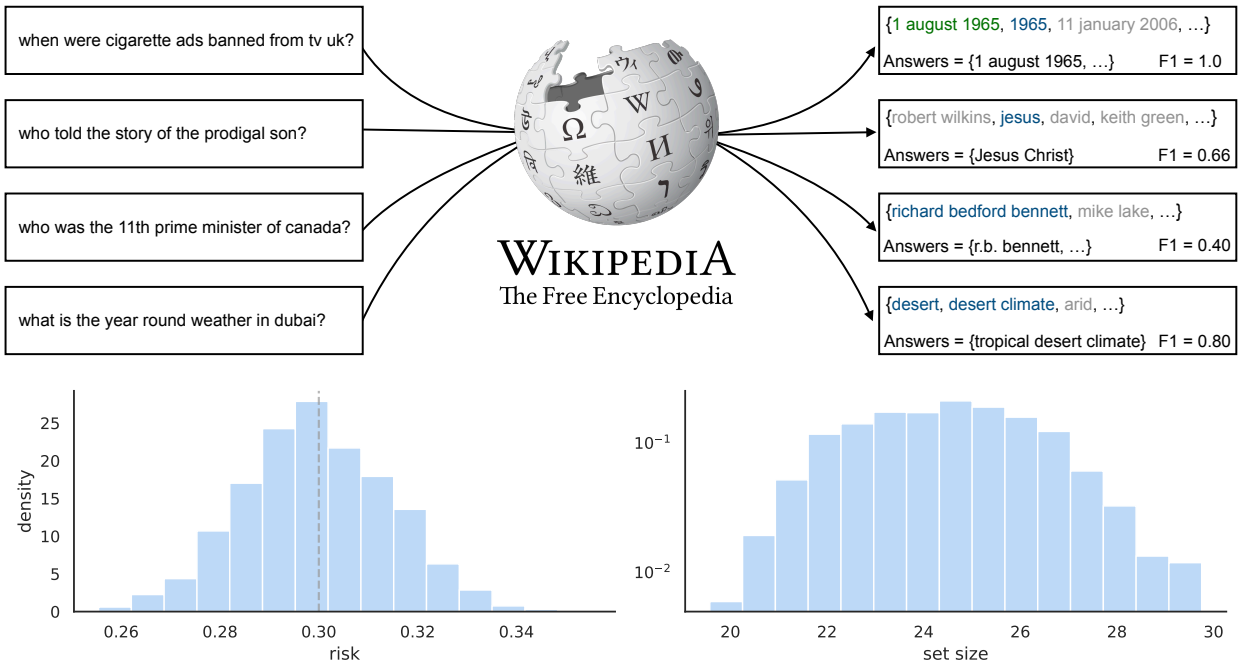

Figure 4: **F1-score control on Natural Questions**. The top figure shows examples of our procedure with fully correct answers in green, partially correct answers in blue, and false positives in gray. Note that due to the nature of the evaluation, answers that are technically correct may still be down-graded if they do not match the reference. We treat this as part of the randomness in the task. The bottom plots report the F1 risk and average set size over 1000 independent random data splits. The dashed gray line marks α.

In the open-domain question answering setting, our input X_i is a question and our label Y_i is a set of (possibly non-unique) correct answers. For example, the input X_{n+1} = "Where was Barack Obama Born?" could have the answer set $Y_{n+1} = \{$"Hawaii", "Honolulu, Hawaii", "Kapo'olani Medical Center"$\}$. Formally, here we treat all questions and answers as being composed of sequences (up to size m) of tokens in a vocabulary $\mathcal{V}$—i.e., assuming k valid answers, we have $X_i \in \mathcal{Z}$ and $Y_i \in \mathcal{Z}^k$, where $\mathcal{Z} := \mathcal{V}^m$. Using an open-domain question answering model that individually scores candidate output answers $f\colon \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$, we calibrate the *best* token-based F1-score of the prediction set, taken over all pairs of predictions and answers:

$$L_i^{\text{F1}}(\lambda) = 1 - \max\left\{\text{F1}(a, c)\colon c \in \mathcal{C}_\lambda(X_i), a \in Y_i\right\}, \text{ where } \mathcal{C}_\lambda(X_i) = \left\{y \in \mathcal{V}^m : f(X_i, y) \geq \lambda\right\}.$$

We define the F1-score following popular QA evaluation metrics [32], where we treat predictions and ground truth answers as bags of tokens and compute the geometric average of their precision and recall (while ignor-

ing punctuation and articles {"a", "an", "the"}). Since L_i^{F1}, as defined in this way, is monotone and upper bounded by 1, it can be controlled by choosing $\hat{\lambda}$ as in Section 2.1 to achieve the following guarantee:

$$\mathbb{E}\left[L_{n+1}^{\text{F1}}(\hat{\lambda})\right] \leq \alpha.$$

We use the Natural Questions (NQ) dataset [33], a popular open-domain question answering baseline, to evaluate our method. We use the splits distributed as part of the Dense Passage Retrieval (DPR) package [34]. Our base model is the DPR Retriever-Reader model [34], which retrieves passages from Wikipedia that might contain the answer to the given query, and then uses a reader model to extract text sub-spans from the retrieved passages that serve as candidate answers. Instead of enumerating all possible answers to a given question (which is intractable), we retrieve the top several hundred candidate answers, extracted from the top 100 passages (which is sufficient to control all risks of interest). We use $n = 2500$ calibration points, and evaluate risk control with the remaining 1110. We report results with $\alpha = 0.3$ in Figure 4. The mean and standard deviation of the risk over 1000 trials are 0.2996 and 0.0150, respectively.

4 Discussion

This generalization of conformal prediction broadens its scope to new applications, as shown in Section 3. The mathematical tools developed in Section 2 and in the appendix may be of independent technical interest, since they provide a new and more general language for studying conformal prediction along with new results about its validity. Important questions remain, such as (1) whether a conformal-type algorithm exists for providing a finite-sample bound when the losses are not monotone but the risk is, (2) whether other functions, such as the value-at-risk, can be controlled with a similar algorithm, and (3) whether there exists a full conformal or cross conformal version of conformal risk control.

Acknowledgements

The authors would like to thank Amit Kohli, Sherrie Wang, and Tijana Zrnić for comments on early drafts. A. A. would like to thank Ziheng (Tony) Wang for helpful conversations. A. A. is funded by the NSF GRFP and a Berkeley Fellowship. S. B. is supported by the NSF FODSI fellowship and the Simons institute. A. F. is partially funded by the NSF GRFP and MIT MLPDS.

References

[1] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic Learning in a Random World*. New York, NY, USA: Springer, 2005.

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

[3] A. N. Angelopoulos, S. Bates, J. Malik, and M. I. Jordan, "Uncertainty sets for image classifiers using conformal prediction," in *International Conference on Learning Representations (ICLR)*, 2021.

[4] A. Fisch, T. Schuster, T. S. Jaakkola, and R. Barzilay, "Efficient conformal prediction via cascaded inference with expanded admission," in *International Conference on Learning Representations*, 2021.

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

[6] A. N. Angelopoulos and S. Bates, *A gentle introduction to conformal prediction and distribution-free uncertainty quantification*, 2021.

[7] G. Shafer and V. Vovk, "A tutorial on conformal prediction," *Journal of Machine Learning Research*, vol. 9, no. Mar, pp. 371–421, 2008.

[8] J. Lei, M. G'Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, "Distribution-free predictive inference for regression," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1094–1111, 2018.

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

[10] I. Gibbs and E. Candes, "Adaptive conformal inference under distribution shift," *Advances in Neural Information Processing Systems*, vol. 34, pp. 1660–1672, 2021.

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

[12] C. Fannjiang, S. Bates, A. N. Angelopoulos, J. Listgarten, and M. I. Jordan, "Conformal prediction for the design problem," *arXiv preprint arXiv:2202.03613*, 2022.

[13] R. Barber, E. Candès, A. Ramdas, and R. Tibshirani, "The limits of distribution-free conditional predictive inference," *Information and Inference*, vol. 10, Aug. 2020.

[14] Y. Romano, E. Patterson, and E. Candès, "Conformalized quantile regression," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 3543–3553.

[15] L. Guan, "Conformal prediction with localization," 2020.

[16] Y. Romano, M. Sesia, and E. Candès, "Classification with valid and adaptive coverage," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 3581–3591.

[17] V. Vovk, "Conditional validity of inductive conformal predictors," in *Proceedings of the Asian Conference on Machine Learning*, vol. 25, 2012, pp. 475–490.

[18] S. Bates, A. Angelopoulos, L. Lei, J. Malik, and M. I. Jordan, "Distribution-free, risk-controlling prediction sets," *Journal of the ACM*, vol. 68, no. 6, Sep. 2021.

[19] A. N. Angelopoulos, S. Bates, E. J. Candès, M. I. Jordan, and L. Lei, "Learn then Test: Calibrating predictive algorithms to achieve risk control," *arXiv preprint arXiv:2110.01052*, 2021.

[20] S. Park, O. Bastani, N. Matni, and I. Lee, "PAC confidence sets for deep neural networks via calibrated prediction," in *International Conference on Learning Representations (ICLR)*, 2020.

[21] A. Fisch, T. Schuster, T. Jaakkola, and R. Barzilay, "Conformal prediction sets with limited false positives," *arXiv preprint arXiv:2202.07650*, 2022.

[22] T. Schuster, A. Fisch, T. Jaakkola, and R. Barzilay, "Consistent accelerated inference via confident adaptive transformers," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2021.

[23] T. Schuster, A. Fisch, J. Gupta, M. Dehghani, D. Bahri, V. Q. Tran, Y. Tay, and D. Metzler, "Confident adaptive language modeling," *arXiv preprint arXiv:2207.07061*, 2022.

[24] S. Sankaranarayanan, A. N. Angelopoulos, S. Bates, Y. Romano, and P. Isola, "Semantic uncertainty intervals for disentangled latent spaces," *arXiv preprint arXiv:2207.10074*, 2022.

[25] A. N. Angelopoulos, A. P. Kohli, S. Bates, M. Jordan, J. Malik, T. Alshaabi, S. Upadhyayula, and Y. Romano, "Image-to-image regression with distribution-free uncertainty quantification and applications in imaging," in *International Conference on Machine Learning*, PMLR, 2022, pp. 717–730.

[26] A. N. Angelopoulos, K. Krauth, S. Bates, Y. Wang, and M. I. Jordan, "Recommendation systems with distribution-free reliability guarantees," *arXiv preprint arXiv:2207.01609*, 2022.

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

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

[29] T. Ridnik, H. Lawen, A. Noy, and I. Friedman, "TResNet: High performance GPU-dedicated architecture," *arXiv:2003.13630*, 2020.

[30] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[32] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ questions for machine comprehension of text," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Nov. 2016, pp. 2383–2392.

[33] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M.-W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov, "Natural questions: A benchmark for question answering research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 452–466, 2019.

[34] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, "Dense passage retrieval for open-domain question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online: Association for Computational Linguistics, Nov. 2020.

[35] J. DeHardt, "Generalizations of the Glivenko-Cantelli theorem," *The Annals of Mathematical Statistics*, vol. 42, no. 6, pp. 2050–2055, 1971.

A Monotonizing non-monotone risks

We next show that the proposed algorithm leads to asymptotic risk control for non-monotone risk functions when applied to a monotonized version of the empirical risk. We set the *monotonized empirical risk* to be

$$\widehat{R}_n^{\uparrow}(\lambda) = \sup_{t \geq \lambda} \widehat{R}_n(t),$$

then define

$$\hat{\lambda}_n^{\uparrow} = \inf\left\{\lambda : \widehat{R}_n^{\uparrow}(\lambda) \leq \alpha\right\}.$$

Theorem A.1. *Let the $L_i(\lambda)$ be right-continuous, i.i.d., bounded (both above and below) functions satisfying* (5)*. Then,*

$$\lim_{n\to\infty} \mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}_n^{\uparrow}\big)\Big] \leq \alpha.$$

Theorem A.1 implies that an analogous procedure to 4 also controls the risk asymptotically. In particular, taking

$$\tilde{\lambda}^{\uparrow} = \inf\left\{\lambda : \widehat{R}_n^{\uparrow}(\lambda) + \frac{B}{n+1} \leq \alpha\right\}$$

also results in asymptotic risk control (to see this, plug $\tilde{\lambda}^{\uparrow}$ into Theorem A.1 and see that the risk level is bounded above by $\alpha - \frac{B}{n+1}$). Note that in the case of a monotone loss function, $\tilde{\lambda}^{\uparrow} = \hat{\lambda}$. However, the counterexample in Proposition 2 does not apply to $\tilde{\lambda}^{\uparrow}$, and it is currently unknown whether this procedure does or does not provide finite-sample risk control.

B Proofs

The proof of Theorem 2 uses the following lemma on the approximate continuity of the empirical risk.

Lemma 1 (Jump Lemma)**.** *In the setting of Theorem 2, any jumps in the empirical risk are bounded, i.e.,*

$$\sup_{\lambda} J\big(\widehat{R}_n, \lambda\big) \overset{a.s.}{\leq} \frac{B}{n}.$$

Proof of Jump Lemma, Lemma 1. By boundedness, the maximum contribution of any single point to the jump is $\frac{B}{n}$, so

$$\exists \lambda :\ J\big(\widehat{R}_n, \lambda\big) > \frac{B}{n} \Longrightarrow \exists \lambda :\ J(L_i, \lambda) > 0 \text{ and } J(L_j, \lambda) > 0 \text{ for some } i \neq j.$$

Call $\mathcal{D}_i = \{\lambda : J(L_i, \lambda) > 0\}$ the sets of discontinuities in L_i. Since L_i is bounded monotone, $\mathcal{D}_i$ has countably many points. The union bound then implies that

$$\mathbb{P}\left(\exists \lambda :\ J(\widehat{R}_n, \lambda) > \frac{B}{n}\right) \leq \sum_{i \neq j} \mathbb{P}(\mathcal{D}_i \cap \mathcal{D}_j \neq \emptyset)$$

Rewriting each term of the right-hand side using tower property and law of total probability gives

$$\mathbb{P}\left(\mathcal{D}_i \cap \mathcal{D}_j \neq \emptyset\right) = \mathbb{E}\left[\mathbb{P}\big(\mathcal{D}_i \cap \mathcal{D}_j \neq \emptyset \;\big|\; \mathcal{D}_j\big)\right] \leq \mathbb{E}\left[\sum_{\lambda \in \mathcal{D}_j} \mathbb{P}\left(\lambda \in \mathcal{D}_i \;\middle|\; \mathcal{D}_j\right)\right] = \mathbb{E}\left[\sum_{\lambda \in \mathcal{D}_j} \mathbb{P}\left(\lambda \in \mathcal{D}_i\right)\right],$$

Where the second inequality is because the union of the events $\lambda \in \mathcal{D}_j$ is the entire sample space, but they are not disjoint, and the third equality is due to the independence between $\mathcal{D}_i$ and $\mathcal{D}_j$. Rewriting in terms of the jump function and applying the assumption $\mathbb{P}\left(J(L_i, \lambda) > 0\right) = 0$,

$$\mathbb{E}\left[\sum_{\lambda \in \mathcal{D}_j} \mathbb{P}\left(\lambda \in \mathcal{D}_i\right)\right] = \mathbb{E}\left[\sum_{\lambda \in \mathcal{D}_j} \mathbb{P}\left(J(L_i, \lambda) > 0\right)\right] = 0.$$

Chaining the above inequalities yields $\mathbb{P}\left(\exists \lambda : J(\widehat{R}_n, \lambda) > \frac{B}{n}\right) \leq 0$, so $\mathbb{P}\left(\exists \lambda : J(\widehat{R}_n, \lambda) > \frac{B}{n}\right) = 0$. □

Proof of Theorem 2. Define the quantity

$$\hat{\lambda}'' = \inf\left\{\lambda : \widehat{R}_{n+1}(\lambda) + \frac{B}{n+1} \leq \alpha\right\}.$$

Deterministically, $\frac{n}{n+1}\widehat{R}_n(\lambda) \leq \widehat{R}_{n+1}(\lambda)$, which yields $\hat{\lambda} \leq \hat{\lambda}''$. Again since $L_i(\lambda)$ is non-increasing in λ,

$$\mathbb{E}\left[L_{n+1}\big(\hat{\lambda}''\big)\right] \leq \mathbb{E}\left[L_{n+1}\big(\hat{\lambda}\big)\right]$$

By exchangeability and the fact that $\hat{\lambda}''$ is a symmetric function of $L_1, \ldots, L_{n+1}$,

$$\mathbb{E}\left[L_{n+1}\big(\hat{\lambda}''\big)\right] = \mathbb{E}\left[\widehat{R}_{n+1}\big(\hat{\lambda}''\big)\right]$$

For the remainder of the proof we focus on lower-bounding $\widehat{R}_{n+1}\big(\hat{\lambda}''\big)$. We begin with the following identity:

$$\alpha = \widehat{R}_{n+1}\big(\hat{\lambda}''\big) + \frac{B}{n+1} - \Big(\widehat{R}_{n+1}\big(\hat{\lambda}''\big) + \frac{B}{n+1} - \alpha\Big).$$

Rearranging the identity,

$$\widehat{R}_{n+1}\big(\hat{\lambda}''\big) = \alpha - \frac{B}{n+1} + \Big(\widehat{R}_{n+1}\big(\hat{\lambda}''\big) + \frac{B}{n+1} - \alpha\Big).$$

Using the Jump Lemma to bound $\left(\widehat{R}_{n+1}(\hat{\lambda}'') + \frac{B}{n+1} - \alpha\right)$ below by $-\frac{B}{n+1}$ gives

$$\widehat{R}_{n+1}(\hat{\lambda}'') \geq \alpha - \frac{2B}{n+1}.$$

Finally, chaining together the above inequalities,

$$\mathbb{E}\Big[L_{n+1}(\hat{\lambda})\Big] \geq \mathbb{E}\Big[\widehat{R}_{n+1}(\hat{\lambda}'')\Big] \geq \alpha - \frac{2B}{n+1}.$$

□

Proof of Proposition 1. Without loss of generality, assume $B = 1$. Fix any $\epsilon' > 0$. Consider the following loss functions, which satisfy the conditions in Theorem 2:

$$L_i(\lambda) \overset{i.i.d.}{\sim} \begin{cases} 1 & \lambda \in [0, Z_i) \\ \frac{k}{k+1} & \lambda \in [Z_i, W_i) \\ 0 & \text{else} \end{cases},$$

where $k \in \mathbb{N}$, the $Z_i \overset{i.i.d.}{\sim} \text{Uniform}(0, 0.5)$, the $W_i \overset{i.i.d.}{\sim} \text{Uniform}(0.5, 1)$ for $i \in \{1, ..., n+1\}$ and $\alpha = \frac{k+1-\epsilon'}{n+1}$. Then, by the definition of $\hat{\lambda}$, we know

$$\widehat{R}_n(\hat{\lambda}) \leq \frac{k - \epsilon'}{n}. \tag{9}$$

If $n > k+1$, $\widehat{R}(\lambda) \geq \frac{k}{k+1} > \frac{k}{n}$ whenever $\lambda \leq \frac{1}{2}$. Thus, we must have $\hat{\lambda} > \frac{1}{2}$. Since k is an integer and by (9), we know that $\left|\{i \in \{1, ..., n\} : L_i(\hat{\lambda}) > 0\}\right| \leq \lfloor (k+1)(k-\epsilon')/k \rfloor \leq k$. This immediately implies that

$$\hat{\lambda} \geq W_{(n-k+1)},$$

where $W_{(j)}$ denotes the j-th order statistic. Notice that for all $\lambda > \frac{1}{2}$,

$$R(\lambda) = \mathbb{E}\left[L_i(\lambda)\right] = \frac{k}{k+1}\mathbb{P}\big(W_i > \lambda\big) = \frac{k}{k+1} \cdot 2(1-\lambda),$$

so $R(\hat{\lambda}) \leq \frac{k}{k+1} \cdot 2(1 - W_{(n-k+1)})$. Let $U_{(k)}$ be the k-th smallest order statistic of n i.i.d. uniform random variables on $(0, 1)$. Then, by symmetry and rescaling, $2(1 - W_{(n-k+1)}) \overset{d}{=} U_{(k)}$,

$$R(\hat{\lambda}) \preceq \frac{k}{k+1} U_{(k)},$$

where $\preceq$ denotes the stochastic dominance. It is well-known that $U_{(k)} \sim \text{Beta}(k, n+1-k)$ and hence

$$\mathbb{E}[R(\hat{\lambda})] \leq \frac{k}{k+1} \cdot \frac{k}{n+1}.$$

Thus,

$$\alpha - \mathbb{E}\Big[R(\hat{\lambda})\Big] \geq \frac{k+1-\epsilon}{n+1} - \frac{k^2}{(n+1)(k+1)} = \frac{1}{n+1} \cdot \frac{(2-\epsilon')k + 1 - \epsilon'}{k+1}.$$

For any given $\epsilon > 0$, let $\epsilon' = \epsilon/2$ and $k = \lceil \frac{2}{\epsilon} - 1 \rceil$. Then

$$\frac{(2-\epsilon')k + 1 - \epsilon'}{k+1} \geq 2 - \epsilon,$$

implying that

$$\alpha - \mathbb{E}\Big[R(\hat{\lambda})\Big] \geq \frac{2-\epsilon}{n+1}.$$

□

Proof of Proposition 2. Assume $\hat{\lambda}$ takes values in $[0,1]$ and $\alpha \in (1/(n+1), 1)$. Let $p \in (0,1)$, N be any positive integer, and $L_i(\lambda)$ be i.i.d. right-continuous piecewise constant (random) functions with

$$L_i(N/N) = 0, \quad (L_i(0/N), L_i(1/N), \ldots, L_i((N-1)/N)) \overset{i.i.d.}{\sim} \mathrm{Ber}(p).$$

By definition, $\hat{\lambda}$ is independent of L_{n+1}. Thus, for any $j = 0, 1, \ldots, N-1$,

$$\Big\{ L_{n+1}(\hat{\lambda}) \mid \hat{\lambda} = j/N \Big\} \sim \mathrm{Ber}(p), \quad \Big\{ L_{n+1}(\hat{\lambda}) \mid \hat{\lambda} = 1 \Big\} \sim \delta_0.$$

Then,

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big) \Big] = p \cdot \mathbb{P}(\hat{\lambda} \neq 1)$$

Note that

$$\hat{\lambda} \neq 1 \iff \min_{j \in \{0,\ldots,N-1\}} \frac{1}{n+1} \sum_{i=1}^{n} L_i(j/N) \leq \alpha - \frac{1}{n+1}.$$

Since $\alpha > 1/(n+1)$,

$$\begin{aligned}
\mathbb{P}(\hat{\lambda} \neq 1) = 1 - \mathbb{P}(\hat{\lambda} = 1) &= 1 - \mathbb{P}\left(\text{for all } j, \text{ we have } \frac{1}{n+1} \sum_{i=1}^{n} L_i(j/N) > \alpha - \frac{1}{n+1} \right) \\
&= 1 - \left(\sum_{k=\lceil (n+1)\alpha \rceil}^{n} \binom{n}{k} p^k (1-p)^{(n-k)} \right)^N \\
&= 1 - \big(1 - \mathrm{BinoCDF}\big(n, p, \lceil (n+1)\alpha \rceil - 1\big)\big)^N
\end{aligned}$$

As a result,

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big) \Big] = p \left(1 - \big(1 - \mathrm{BinoCDF}\big(n, p, \lceil (n+1)\alpha \rceil - 1\big)\big)^N \right).$$

Now let N be sufficiently large such that

$$\left(1 - \big(1 - \mathrm{BinoCDF}\big(n, p, \lceil (n+1)\alpha \rceil - 1\big)\big)^N \right) > p.$$

Then

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}\big) \Big] > p^2$$

For any $\alpha > 0$, we can take p close enough to 1 to render the claim false. □

Proof of Theorem A.1. Define the *monotonized population risk* as

$$R^{\uparrow}(\lambda) = \sup_{t \geq \lambda} \mathbb{E}\Big[L_{n+1}(t) \Big]$$

Note that the independence of L_{n+1} and $\hat{\lambda}_n^{\uparrow}$ implies that for all n,

$$\mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}_n^{\uparrow}\big) \Big] \leq \mathbb{E}\Big[R^{\uparrow}\big(\hat{\lambda}_n^{\uparrow}\big) \Big].$$

Since $R^{\uparrow}$ is bounded, monotone, and one-dimensional, a generalization of the Glivenko-Cantelli Theorem given in Theorem 1 of [35] gives that uniformly over λ,

$$\lim_{n \to \infty} \sup_{\lambda} |\widehat{R}_n(\lambda) - R(\lambda)| \overset{a.s.}{\to} 0.$$

As a result,

$$\lim_{n\to\infty} \sup_{\lambda} |\widehat{R}_n^{\uparrow}(\lambda) - R^{\uparrow}(\lambda)| \overset{a.s.}{\to} 0,$$

which implies that

$$\lim_{n\to\infty} |\widehat{R}_n^{\uparrow}(\hat{\lambda}^{\uparrow}) - R^{\uparrow}(\hat{\lambda}^{\uparrow})| \overset{a.s.}{\to} 0.$$

By definition, $\widehat{R}^{\uparrow}(\hat{\lambda}^{\uparrow}) \leq \alpha$ almost surely and thus this directly implies

$$\limsup_{n\to\infty} R^{\uparrow}\big(\hat{\lambda}_n^{\uparrow}\big) \leq \alpha \quad \text{a.s..}$$

Finally, since for all n, $R^{\uparrow}\big(\hat{\lambda}_n^{\uparrow}\big) \leq B$, by Fatou's lemma,

$$\lim_{n\to\infty} \mathbb{E}\Big[L_{n+1}\big(\hat{\lambda}_n^{\uparrow}\big)\Big] \leq \limsup_{n\to\infty} \mathbb{E}\Big[R^{\uparrow}\big(\hat{\lambda}_n^{\uparrow}\big)\Big] \leq \mathbb{E}\Big[\limsup_{n\to\infty} R^{\uparrow}\big(\hat{\lambda}_n^{\uparrow}\big)\Big] \leq \alpha.$$

□