STAT9700 Final Project: Conformal Prediction

Joseph Rudoler

December 13, 2023

Abstract

Conformal prediction is a model-agnostic framework for constructing valid prediction sets without making any assumptions about the underlying data distribution. As modern machine learning systems continue to scale in both size and complexity, the ability to make valid predictions without making assumptions about the training data or model is becoming increasingly important. In particular, the success and ubiquity of deep and over-parameterized neural networks has led to a growing interest in techniques that can provide statistically rigorous uncertainty quantification for models that are effectively a "black box". In this report, we will provide an overview of the conformal prediction framework and some important results, discuss some extensions and demonstrate an application of conformal prediction for a natural language processing (NLP) task.

Contents

1 Introduction

Conformal prediction (also known as conformal inference) was first introduced by Vovk et al. (1999) and has seen a surge of interest within the last decade. Contributing to this increased interest is the growing popularity of machine learning models which are effectively "black boxes" - that is, models which have so many parameters and such flexible architectures that it is nearly impossible do inference on their learned parameters. Deep neural networks with millions or billions of parameters are now ubiquitous in important tasks in computer vision, natural language processing, and reinforcement learning. These models are for safety-critical applications like medical diagnosis or autonomous driving, making uncertainty quantification paramount for their deployment.

Conformal prediction provides a framework for constructing prediction sets (in continuous settings like regression, prediction intervals) with rigorous coverage guarantees for *any* model at test time. The only assumption required is that the data are exchangeable, and even this assumption can be relaxed when we have information about the distribution shift that violates exchangeability.

This report is organized as follows: We will begin with an overview of the conformal prediction framework and some important results, then briefly summarize some key extensions, and finally provide an example application of conformal prediction to text classification with a large language model.

2 Foundations

This section will begin with a discussion of some conformal prediction basics, loosely following A. N. Angelopoulos and Bates, 2022. This will include an abbreviated proof of the validity of conformal prediction sets for exchangeable data, and an overview of some important applications and extensions of conformal prediction. We will then discuss how conformal prediction can be used in a cross-validation setting to obtain confidence intervals with rigorous coverage guarantees (Barber et al., 2020). Next, we will discuss in more detail how conformal prediction can be married to kernel density estimation to obtain prediction sets with asymptotic efficiency guarantees (Lei et al., 2013). Finally, we will see what happens when we try to apply conformal prediction to non-exchangeable data (Barber et al., 2023).

2.1 Conformal prediction basics

The central goal of conformal prediction is to construct prediction sets which have coverage guarantees at a specified confidence level $1-\alpha$. That is, given the task of mapping some inputs $X \in \mathcal{X} \mapsto \mathcal{Y}$, we ideally want to construct a set $\hat{C}(X) \subset \mathcal{Y}$ such that $\mathbb{P}(Y \in \hat{C}(X)) \geq 1-\alpha$ for all $X \in \mathcal{X}$. This set can be discrete or continous, depending on the prediction space $\mathcal{Y}$ for a given problem. As it turns out, when the data are i.i.d. or even just exchangeable, this is pretty easy. The definition of conformal prediction sets and the formal statement of the coverage guarantee, for exchangeable data, is as follows:

Theorem 1 (Conformal prediction sets for exchangeable data)**.** *Suppose that* (X_i, Y_i) *for* $i = 1, \ldots, n$ *and* (X_{test}, Y_{test}) *are exchangeable random variables.*
Define

$$\hat{q} = \inf\left\{q : \frac{|i : s(X_i, Y_i) \leq q|}{n} \geq \frac{\lceil (n+1)(1-\alpha) \rceil}{n}\right\}$$

where $\lceil \cdot \rceil$ is the ceiling function.
Define a prediction set

$$\hat{C}(X) = \{y : s(X, y) \leq \hat{q}\}$$

Then

$$\mathbb{P}(Y_{test} \in \hat{C}(X_{test})) \geq 1 - \alpha$$

Informally, we define a *conformal score* function $s : (\mathcal{X}, \mathcal{Y} \mapsto \mathbb{R})$ which serves as a heuristic measure of how unlikely a given observation is (usually based on a trained model). This name is perhaps a bit misleading, since it actually measures the degree of an observation's nonconformity with the data used to train the model. Next, for a test observation X_{test} we predict the set of all Y which are below approximately the $(1-\alpha)$ quantile of these scores on a calibration dataset. This means that we're choosing a prediction set which contains all Y which are "not too unlikely" given the calibration data. Note that the coverage guarantee holds for any s, but the choice of s will affect the size (and consequently, the usefulness) of the prediction set $\hat{C}(X)$ - intuitively, the less informative s is, the larger $C(X)$ will need to be to achieve the desired coverage.
The proof of this coverage guarantee (with the simplifying assumption that $s(X_i, Y_i)$ are unique, to avoid handling ties) is as follows:

Proof. Begin with a set of exchangeable pairs $(X_1, Y_1), \ldots, (X_n, Y_n), (X_{\text{test}}, Y_{\text{test}})$. We want to obtain a prediction set for X_{test} which contains the true value Y_{test} with probability at least $1 - \alpha$. Assume without loss of generality that we have a model (e.g. a trained neural network) which can be used to compute a score $s(X, Y)$ for any (X, Y) pair.
We can sort the scores $s(X_i, Y_i)$ for $i = 1, \ldots, n$ in ascending order, and denote the ith smallest score by $s_{(i)}$. We set the threshold

$$\hat{q} = \inf\left\{q : \frac{|i : s(X_i, Y_i) \leq q|}{n} \geq \frac{\lceil (n+1)(1-\alpha) \rceil}{n}\right\}$$

Since the scores are sorted (with no ties), we can see that

$$\hat{q} = \begin{cases} s_{(\lceil (n+1)(1-\alpha) \rceil)} & \text{if } \alpha \geq \frac{1}{n+1} \\ +\infty & \text{otherwise} \end{cases}$$

And accordingly construct a prediction set:

$$\hat{C}(X) = \{y : s(X, y) \leq \hat{q}\}$$

What is the probability that $Y_{\text{test}} \in C(X_{\text{test}})$? This is where the exchangeability assumption is crucial. Since the data are exchangeable, s_{test} is distributed identically to s_i for any $i = 1, \ldots, n$. It follows that s_{test} is equally likely to be fall into any of the $n+1$ intervals between the n calibration scores. Formally, for any integer k we have:

$$\mathbb{P}(s_{\text{test}} \leq s_{(k)}) = \frac{k}{n+1}$$

In particular, we have:

$$\mathbb{P}(s_{\text{test}} \leq s_{(\lceil (n+1)(1-\alpha) \rceil)}) = \frac{\lceil (n+1)(1-\alpha) \rceil}{n+1} \geq 1 - \alpha$$

Which proves the coverage guarantee. □

Part of the "magic" of conformal prediction is that this procedure can be applied post-training to any model, for basically any prediction task. That means we can treat any model as a "black box" and still obtain valid prediction sets without making any assumptions about how the model was parameterized. While we assume that the data are exchangeable (i.e. identically distributed), they need not be independent or to follow any particular distribution. We will see later how this assumption can be relaxed to allow for non-exchangeable data.
To help solidify our intuition for conformal prediction, we include a simple outline of the conformal procedure as described in (A. N. Angelopoulos & Bates, 2022) and concurrently expand the example case of multi-label image classification:

1. Given a pre-trained model, we **identify a heuristic notion of uncertainty** for the model's predictions. In image classification, we might use the softmax probabilities $\hat{f}(X)$ output by a convolutional neural network.
2. **Define a conformal score function $s(\cdot)$** based on this uncertainty measure. For our example, we might define $s(X, Y) = 1 - \hat{f}(X)_Y$. This score function is lower for more likely image labels, and higher for less likely image labels.
3. **Set a threshold $\hat{q}$** at the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ quantile of the scores $s(X_i, Y_i)$ for $i = 1, \ldots, n$.
4. **Construct a prediction set** $\hat{C}(X_{\text{test}}) = \{y : s(X_{\text{test}}, y) \leq \hat{q}\}$. In image classification, this is the set of all image labels with a score below the $\hat{q}$ quantile of scores on the calibration dataset. In other words, you add the mostly likely labels to the prediction set until their total probability mass under the distribution of scores on the calibration dataset is at least $1 - \alpha$. Then the prediction set contains the true image label with probability at least $1 - \alpha$.

Again, we emphasize that the key to making conformal prediction sets *useful* lies in the choice of the score function s. While any s provides sets which are valid (in the sense that they have the desired coverage), a good choice of s will result in a small prediction sets which are also *adaptive*. That is, the size of the prediction set will depend on how much uncertainty the model has about the prediction.

Split vs Full conformal prediction In the above discussion, we assumed that we had access to a calibration dataset of exchangeable pairs (X_i, Y_i) for $i = 1, \ldots, n$ which is non-overlapping with the training data used to fit the model. This is known as *split conformal prediction* (or *inductive conformal prediction*), and it has the desirable property of allowing a user to evaluate a pre-trained model without doing any additional training. However, in some cases we may not have enough data to split into training and calibration sets. In this case, we can use the full dataset to construct a valid prediction set. This is known as *full conformal prediction* (or *transductive conformal prediction*). Full conformal prediction is computationally more expensive than split conformal prediction, as it involves fitting a new model for each possible value $y \in \mathcal{Y}$ and recomputing the scores. However, it is still possible to obtain valid prediction sets without splitting the data. In full conformal prediction, we construct an augmented dataset of exchangeable pairs (X_i, Y_i) for $i = 1, \ldots, n$ by adding a new observation (X_{n+1}, y) to the training data. We then fit a model to this augmented dataset and compute the scores $s^y(X_i, Y_i)$ for $i = 1, \ldots, n+1$. The threshold $\hat{q}$ is now dependent on the value y we added to the training data:

$$\hat{q}^y = \inf \left\{ q : \frac{|i : s^y(X_i, Y_i) \leq q|}{n} \geq \frac{\lceil (n+1)(1-\alpha) \rceil}{n} \right\}$$

And the prediction set becomes:

$$\hat{C}(X) = \{y : s^y(X_{n+1}, y) \leq \hat{q}^y\}$$

Intuitively, this is testing if the the new observation (X_{n+1}, y) is exchangeable with all of the other observations, under the null hypothesis that y is the true label for X_{n+1}. As we will see later on in 2.3, this procedure has provable asymptotic statistical efficiency guarantees. We would expect that this procedure would be more accurate than split conformal prediction, since it uses the full dataset to calibrate the model instead of "wasting" a subset by splitting. The drawback is that full conformal prediction is prohibitively slow, especially when $|\mathcal{Y}|$ is large - in continuous setting like regression, we would need to fit a new model for each possible value of $y \in \mathbb{R}$. In practice there are computational tricks to speed this up (e.g. evaluating on a grid of values), but it is still much slower than split conformal prediction. Full conformal prediction is also not possible when we have access to a pretrained model but not the training data itself, since it requires fitting a new model to the full dataset – we will see an example of this in 4.

Next, we consider iterative holdout methods which use more data for calibration than split conformal prediction, but manage the tradeoff with computational efficiency.

2.2 Holdout methods: Jackknife+ and CV+

Barber et al. (2020) discuss the application of conformal prediction to iterative holdout methods, which are a popular approach to uncertainty quantification in machine learning. To avoid overfitting, it is common to train a model on a partial dataset and reserve the remaining observation as a holdout for validation (or test). Split conformal prediction, as described above, is an example of a method with a single holdout - the conformal procedure is calibrated on part of the data and then applied to the remaining holdout data. (Actually, in practice there is a third set of data - the training data - which is used to fit the model, but not used in the conformal procedure.) Instead of using a single holdout, iterative holdout methods like jackknife and cross-validation iteratively hold out different subsets of the data in order get unbiased estimates of the model's performance while still making use of the entire dataset.

Before proceeding, we define some added notation. Take $\hat{q}^+\{\cdot\}$ is the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ quantile, and $\hat{q}^-\{\cdot\}$ is the $\frac{\lfloor (n+1)\alpha \rfloor}{n}$ quantile. We will also use the notation $\hat{f}_{-i}(X_i)$ to denote the model fit to the full dataset without the ith observation.

The original jackknife method involves fitting a model to the full dataset, and then generating an empirical distribution of test residuals obtained by computing the for each observation from a model fit to the full dataset without that observation. We can use the quantiles of this distribution to construct a confidence region. Specifically, we can define a prediction set for a new observation X_{n+1} as:

$$\hat{C}^{JK}(X_{n+1}) = \hat{f}(X_{n+1}) \pm \hat{q}\{|Y_i - \hat{f}_{-i}(X_i)|\}$$

Or, equivalenty:

$$\hat{C}^{JK}(X_{n+1}) = \left[\hat{q}^- \left\{\hat{f}(X_{n+1}) - |Y_i - \hat{f}_{-i}(X_i)|\right\}, \hat{q}^+ \left\{\hat{f}(X_{n+1}) + |Y_i - \hat{f}_{-i}(X_i)|\right\}\right]$$

While this procedure has been shown to provide good empirical coverage, it is not valid in general - only in the case where the model meets certain stability criteria (Steinberger & Leeb, 2022).

Barber et al. (2020) propose a conformalized version of the jackknife method, which they call Jackknife+. In particular, they define the prediction set for a new observation X_{n+1} as:

$$\hat{C}^{JK+}(X_{n+1}) = \Big[\hat{q}^{-}\left\{\hat{f}_{-i}(X_{n+1}) - |Y_i - \hat{f}_{-i}(X_i)|\right\}, \\ \hat{q}^{+}\left\{\hat{f}_{-i}(X_{n+1}) + |Y_i - \hat{f}_{-i}(X_i)|\right\}\Big]$$

The key difference here is that the test point for X_{n+1} is variable, since it is produced by the iterative model fits excluding observation i. So, instead of getting a symmetric interval around $\hat{f}(X_{n+1})$, we get an interval about the median of $\{\hat{f}_{-i}(X_{n+1})\}$ which is not symmetric in general.

The main result of Barber et al. (2020) is to prove that this slight modification leads to guaranteed coverage of $1 - 2\alpha$. That is,

$$\mathbb{P}(Y_{n+1} \in \hat{C}^{JK+}(X_{n+1})) \geq 1 - 2\alpha$$

The authors note that in practice, the empirical coverage of both jackknife and jackknife+ is close to $1 - \alpha$, and the main advantage of jackknife+ is that it still provides some coverage guarantee even in worst-case scenarios when the model is not stable.

The authors also propose a generalization of this procedure to k-fold cross-validation, which they call CV+. The procedure is the same as above, but we leave out the set of observations S_k in the kth fold rather than a single observation. The prediction set is then:

$$\hat{C}^{CV+}(X_{n+1}) = \Big[\hat{q}^{-}\left\{\hat{f}_{-S_k}(X_{n+1}) - |Y_i - \hat{f}_{-S_k}(X_i)|\right\}, \\ \hat{q}^{+}\left\{\hat{f}_{-S_k}(X_{n+1}) + |Y_i - \hat{f}_{-S_k}(X_i)|\right\}\Big]$$

Of course, these guarantees are weaker than the $1-\alpha$ coverage guarantee of both the split and full conformal prediction procedures. The advantage of jackknife+ over full conformal prediction is that it is computationally less expensive, since one does not need to fit a new model for each possible value of y. Of course, split conformal prediction is still the most computationally efficient, since it does not require any additional model fitting - but in data-poor settings we may prefer to fit the model to as much data as possible. Jackknife+ and CV+ provide a middle ground - one can choose the number of folds to trade off between computational efficiency and statistical efficiency.

2.3 Density estimation

Generating conformalized prediction sets can be cast as a problem of estimating density level sets for an unknown probability distribution (Lei et al., 2013). Consider a sequence of i.i.d. data drawn from some unknown distribution with density $p(y)$. If we want to find a set C for a new observation such that $\mathbb{P}(Y \in C) \geq 1 - \alpha$, this is equivalent to finding an estimated density level set

$$L(t_\alpha) = C_\alpha = \{y : \hat{p}(y) \geq t_\alpha\} \quad \text{where} \quad t_\alpha = \inf\{t : \mathbb{P}(Y \in L(t)) \geq 1 - \alpha\}$$

We will deal with this more general notation in this section (following the original paper), but it is worth noting that in typical machine learning applications we deal with a model that predicts Y based on some features X. The goal is

then using that model's predictions to find a set $C(X)$ such that the probability $\mathbb{P}(Y \in C(X)) \geq 1 - \alpha$.
Note that if $\hat{p}(y)$ were estimated perfectly, we would have no need to conformalize the density level set – we would simply greedily add the most likely observations to the set until we reached the desired coverage. However, in practice we will need to estimate $\hat{p}(y)$ from a finite sample of data and inevitably introduce some error which might cause us to undercover or overcover.
We now consider the standard kernel density estimator (KDE) with kernel fuction K and bandwidth h,

$$\hat{p}(u) = \frac{1}{n} \sum_{i=1}^{n} K_h(u - Y_i)$$

For our score function, we will use an augmented version of the KDE which is estimated on the augmented dataset $Y_1, \ldots, Y_{n+1}$ where $Y_{n+1} = y$:

$$\hat{p}^y(u) = \frac{1}{n+1} \sum_{i=1}^{n+1} K_h(u - Y_i)$$

The paper defines a conformal score a bit differently than we have so far, such that higher conformal scores correspond to more likely labels. This is helpful in order to more naturally frame the prediction set as a density level set. So we have $s^y(y) = \hat{p}^y(y)$, and we set the threshold $\hat{q}^y$ at the $\frac{\lfloor (n+1)\alpha \rfloor}{n}$ quantile of the scores $s^y(X_i, Y_i)$ for $i = 1, \ldots, n$ and take the valid prediction set to be:

$$\hat{C}_\alpha = L(\hat{t}_\alpha) = \{y : s^y(y) \geq \hat{q}^y\}$$

So, our prediction set is all y such that the estimated density level set $L(\hat{t}_\alpha)$ contains y with probability at least $1 - \alpha$. (Note that this is equivalent to making the score function $s^y(y) = -\hat{p}^y(y)$, defining $\hat{q}^y$ at the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ quantile of the scores, taking a prediction set $\hat{C}_\alpha = \{y : s^y(y) \leq \hat{q}^y\}$.)
Since this kernel density estimator is expensive to compute for every possible value of y, Lei et al. propose a faster approximation that sandwiches the kernel density estimator between two simpler estimators (L^- and L^+) and preserves finite-sample validity. That is, they they define the following lemma:

Lemma 1. *Let $\hat{C}_\alpha$ be the conformal prediction set based on the KDE with kernel function K, and assume $\sup_u |K(u)| = K(0)$. Let $Y_{(1)} \ldots Y_{(n)}$ be the data in increasing order by their estimated kernel density.*

$$L^- = L\left(\hat{p}(Y_{(\lfloor (n+1)\alpha \rfloor)})\right) \quad \textit{and} \quad L^+ = L\left(\hat{p}(Y_{(\lfloor (n+1)\alpha \rfloor)})\right) - \frac{1}{nh} \sup_{u,u'} |K(u) - K(u')|$$

Then

$$L^- \subseteq \hat{C}_\alpha \subseteq L^+$$

Importantly, this sandwiching technique also provides lower and upper bounds on the efficiency of the kernel density estimator, which itself is harder to analyze. Lei et al. (2013) proceed to show that this procedure is asymptotically efficient at rate $r_n = (\frac{\log n}{n})^{c_{p_\alpha}}$ where c_{p_α} is a constant depending on the smoothness of the true underlying density. By efficient, we mean that probability of the loss $R(\hat{C}_\alpha, C_\alpha) = \mu(\hat{C}_\alpha \setminus C_\alpha) \leq \mu(\hat{C}_\alpha) - \mu(C_\alpha)$ exceeding r_n converges to zero as $n \to \infty$.

It is worth noting as well that while the asymptotic efficiency of the kernel density estimator is only proven for the full conformal prediction procedure, kernel density estimation can be used to obtain valid prediction sets in the split conformal prediction setting as well.

2.4 Beyond exchangeability

So far we have emphasized that conformal prediction provides wonderful coverage guarantees without assuming anything except exchangeability of the calibration and test data. While this is nice, exchangeability is actually rather strong assumption. In this section, we will discuss extensions of conformal prediction to nonexchangeable data, following Barber et al. (2023). That work provides accomodations for nonexchangeability in the context of split conformal, full conformal, and jackknife+. Here we will focus only on the split conformal case here – it is the most popular and is nearly always the best choice in practice, as datasets in modern machine learning applications are usually quite large.

There are two main ways in which the exchangeability assumption can be violated:

1. The observations $Z_i = (X_i, Y_i)$ are themselves not identically distributed. This could be due to distribution shift between the calibration and test data (perhaps they come from different sources or were collected at different times), or there might be some dependence between the observations in the data-generating process that results in nonexchangeability (e.g. correlation in space or time).
2. The algorithm $\mathcal{A}$ which maps the data to a fitted model $\hat{f}$ does not handle the data symmetrically. For example, $\mathcal{A}$ might assume that the data are chronologically ordered and assign more weight to recent observations.

Since we are dealing with split conformal prediction, where we assume that the model is pre-fitted, this second form of nonexchangeability is not a concern. We will quickly note that the authors prove that their method for nonexchangeable full conformal and jackknife+ works for both symmetric and asymmetric algorithms, but we will not go into detail here. However, this robustness is important because it means we can accomodate many important machine learning models which are not symmetric, like weighted regression or autoregressive models.

We now introduce a bit more notation that helps formalize nonexchangeability. Let $Z = Z_1, \ldots, Z_n, Z_{\text{test}} = (X_1, Y_1), \ldots, (X_n, Y_n), (X_{\text{test}}, Y_{\text{test}})$, and now let Z^i denote the dataset Z where the test observation is swapped with the ith observation. That is,

$$Z^i = Z_1, \ldots, Z_{i-1}, Z_{\text{test}}, Z_{i+1}, \ldots, Z_n, Z_i$$

We define the coverage gap Δ induced by nonexchangeability as:

$$\Delta = (1 - \alpha) - \mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}}))$$

The method for nonexchangeable conformal prediction proposed in Barber et al. assigns weights $w_i \in [0, 1]$ to each observation Z_i. The intuition is is that we want to assign more weight to observations which we think are more "trustworthy" in the sense that they are more likely to be exchangeable with the test observation (e.g. more recent observations a timeseries analysis, or nearby observations in an analysis of spatially correlated data).

Nonexchangeable split conformal prediction Assign weights w_i to each observation, and further define the normalized weights:

$$\tilde{w}_i = \frac{w_i}{1 + \sum_{j=1}^{n} w_j}, \quad \text{and} \quad \tilde{w}_{\text{test}} = \frac{1}{1 + \sum_{j=1}^{n} w_j}$$

Define the prediction set as:

$$\hat{C}(X_{\text{test}}) = \{y : s(X_{\text{test}}, y) \leq \hat{q}\}$$

where the threshold is $\hat{q}$ is defined as

$$\hat{q} = 1 - \alpha \text{ quantile of } \sum_{i=1}^{n} \tilde{w}_i \cdot \delta_{s_i} + \tilde{w}_{\text{test}} \cdot \delta_{+\infty}$$

where δ_{s_i} is the Dirac delta function at $s_i = s(X_i, Y_i)$ and $\delta_{+\infty}$ is the Dirac delta function at $+\infty$.

In other words, we compute the conformal scores for each observation Z_i, and then compute the weighted sum of these scores. The test observation is assigned a weight of $\tilde{w}_{\text{test}}$, and all other observations are assigned weights $\tilde{w}_i$. The prediction set is then the set of all y such that the score is less than the $1 - \alpha$ quantile of the weighted sum of scores. This departure in notation is aided a bit by noting that the original split conformal threshold $\hat{q}$ can equivalently be written as:

$$\hat{q} = 1 - \alpha \text{ quantile of } \sum_{i=1}^{n} \frac{1}{n+1}\delta_{s_i} + \frac{1}{n+1}\delta_{+\infty}$$

This is exactly the reduced form of the nonexchangeable threshold above when all the weights are equal to 1 (i.e. we assume exchangeability).

Barber et al. prove that weighting the observations in this manner leads to an upper bound on the coverage gap Δ (phrased differently here than in the original paper):

Theorem 2 (Coverage gap induced by nonexchangeability)**.** *Consider a sequence of (possibly) nonexchangeable data data Z. For prediction sets defined above, the coverage gap Δ induced by nonexchangeability is bounded as follows:*

$$\Delta \leq \sum_{i=1}^{n} \tilde{w}_i \cdot d_{TV}(Z, Z^i)$$

where d_{TV} is the total variation distance between the distributions of Z and Z^i.

Importantly in the case of independent (but not necessarily identically distributed) data, there is a helpful result:

$$d_{TV}(Z, Z^i) \leq 2 \cdot d_{TV}(Z_i, Z_{\text{test}}) - d_{TV}(Z_i, Z_{\text{test}})^2 \leq 2 \cdot d_{TV}(Z_i, Z_{\text{test}})$$

Which allows the bound in our theorem to be simplified to:

$$\Delta \leq 2 \sum_{i=1}^{n} \tilde{w}_i \cdot d_{TV}(Z_i, Z_{\text{test}})$$

This bound implies that the effect of each observation on the coverage gap can be weighted independently – therefore, if we assign small weights to observations which are nonexchangeable with the test observation, we can mitigate the coverage gap. This confirms our intuition that we should assign more weight to observations which we think are more "trustworthy", and means that if we know something about the distribution shifts affecting our data, we can leverage that knowledge to get approximately valid coverage. There is, however, a tradeoff – assigning small weights reduces the effective sample size, which will lead to larger prediction sets.

Another happy consequence of this theorem is that in the case of exchangeable data, the joint distribution is invariant under permutations so $d_{TV} = 0$ and the coverage gap is zero. This means that applying the nonexchangeable split conformal prediction procedure to exchangeable data will not result in any loss of coverage. This means that we get additional robustness to nonexchangeability without sacrificing coverage guarantees.

A final benefit is that the above theorem implies a previously unknown bound on the coverage gap for the original split conformal prediction procedure, which is the special case of the above procedure where all $w_i = 1$:

$$\Delta \leq \frac{\sum_{i=1}^{n} d_{TV}(Z_i, Z_{\text{test}})}{n+1}$$

This bound helps to explain why the original procedures still work reasonably well under mild violations of exchangeability (i.e. when the total variation distances between permutations are nonzero but are either sparse or reasonably small).

3 Extensions and related work

Much of the recent work on conformal prediction has focused on the design of score functions that accomodate various notions of uncertainty or achieve specific desiderata like small prediction sets. In this section, we will briefly discuss some of these extensions. The discussion here is inspired by A. N. Angelopoulos and Bates (2022), which provides a very comprehensive and up-to-date survey.

3.1 Conformalized Bayes.

Framing conformal prediction as a density level set estimation problem, as in 2.3, also connects nicely to Bayesian inference. In particular, we can choose the conformal score function to be the posterior predictive density of the label given the data:

$$s(X, y) = -\hat{f}(y|X)$$

Then the prediction set will be the poster predictive density level set with coverage guarantee $1 - \alpha$. "Conformalized Bayes" allows us to obtain valid prediction sets for Bayesian models without making any assumptions about the correctness of the model or the data distribution – in this sense it mitigates concerns about misspecified priors. Hoff (2021) shows that under certain conditions, the split conformal procedure with the above score is Bayes optimal (it minimizes the integrated risk – wherein we consider the expectation of the loss over both the data and the prior – among all valid prediction sets). The important takeaway is that we can incorporate prior knowledge into our predictions without sacrificing coverage guarantees.

While typical Bayesian modeling has strong inductive biases and is often not appropriate in settings where we know little about the data generating process, this is still relevant as Bayesian deep learning (Gal & Ghahramani, 2016; MacKay, 1992; Wilson & Izmailov, 2020) has gained popularity. Bayesian neural networks are neural networks with some (usually noninformative) prior distribution over the weights, which use a variety of clever techniques to estimate uncertainty by approximating the posterior predictive distribution. Like regular Bayesian models trained with expensive sampling procedures, Bayesian neural networks are computationally expensive compared to their non-Bayesian counterparts (though not prohibitively so).

3.2 Scalar uncertainty estimates

There are many existing methods in the machine learning literature which provide estimates of uncertainty at the level of individual predictions, and often they are quite precise and even provide excellent empirical coverage. The problem is that these approaches do not in general have theoretical coverage guarantees. Luckily, these approaches are compatible with conformal prediction, and in fact conformal prediction *relies* on heuristic uncertainty estimates built into the conformal score in order to construct prediction sets. As discussed in 2.1, without an informative score function the prediction sets will be large and imprecise. Conformalizing these heuristic uncertainty estimates can give us prediction sets which are both accurate and precise.

One example of a popular approach to uncertainty quantification, especially in deep learning, is to use a modified loss function which downweights residuals with a high estimated uncertainty. A large neural network can easily be modified to output a scalar uncertainty estimate along with its prediction (A. N. Angelopoulos, Kohli, et al., 2022; Lakshminarayanan et al., 2017; Nix & Weigend, 1994). A simple update to the typical squared error loss is a Gaussian loss function (i.e. maximizing the log likelihood):

$$\mathcal{L}(x, y) = \log(\hat{\sigma}(x)) + \frac{(y - \hat{\mu}(x))^2}{\hat{\sigma}(x)}$$

where $\hat{\mu}(x)$ and $\hat{\sigma}(x)$ are the predicted mean (point estimate) and variance.

With this uncertainty estimate in hand, we can construct a score function:

$$s(X, y) = \frac{|y - \hat{\mu}(X)|}{\hat{\sigma}(X)}$$

This score function grows when the residual is large relative to the uncertainty estimate. In other words, large scores reflect unexpectedly large residuals – the score adaptively shrinks when the model expects high variance. Taking $\hat{q}$ to be the $\frac{\lceil (n+1)(1-\alpha) \rceil}{n}$ quantile the calibration scores, we can form the prediction intervals:

$$\hat{C}(X_{\text{test}}) = \left[\hat{f}(x) - \hat{q} \cdot \hat{\sigma}(x), \hat{f}(x) + \hat{q} \cdot \hat{\sigma}(x) \right]$$

Effectively, $\hat{q}$ serves as a multiplicative correction factor for the uncertainty estimate that ensures that the prediction set has the desired coverage. If the uncertainty estimates are too conservative (i.e. the residuals are consistently smaller than the uncertainty estimates), $\hat{q}$ will be smaller than 1 and will shrink the prediction set. If the uncertainty estimates are too optimistic (i.e. the residuals are consistently larger than the uncertainty estimates), $\hat{q}$ will be larger than 1 and will expand the prediction set.

We note that this uncertainty estimate does not need to be the Gaussian variance estimated this way, and other modified loss functions have been proposed (Tohme et al., 2023). There are many ways to obtain scalar uncertainty estimates of a neural network's predictions, including estimating variance across ensembles or when randomly dropping out neurons or layers in a network. These approaches are totally compatible with the above conformal score and prediction set construction.

3.3 Quantile Regression

Briefly, we mention that a related and popular technique is quantile regression (Koenker & Bassett, 1978) which also modifies the loss function so that the model

learns to predict quantiles instead of point estimates. Like the Gaussian loss function, this can be used on top of practically any model architecture and can similarly be conformalized to grow or shrink the prediction set to achieve coverage during calibration (Romano et al., 2019). This approach inherits nice statistical properties from quantile regression, including asymptotic conditional coverage (which we discuss next).

3.4 Adaptivity and conditional coverage

The overall coverage guarantees of conformal prediction are reassuring, but we might have additional desiderata for our prediction sets. In particular, the coverage guarantee is averaged over all of the data in our calibration set, but any practitioner of data science knows that most datasets are not heterogeneous – a model may perform better on some subsets of the data than others. In conformal prediction, this can lead to some subsets being overcovered and others undercovered. We therefore want our conformal procedure to adaptively provide wider coverage for the "harder" observations. This notion of *adaptivity* led to the development of conformal scores and procedures which produce adaptively sized prediction sets (A. Angelopoulos et al., 2022; Romano et al., 2020). We defer further discussion to 4, where we provide an in-depth example in the setting of text classification.

One way to think about adaptivity is that if it worked perfectly, it would actually amount to a much stronger coverage guarantee: *conditional coverage*. The original coverage guarantee can be thought of as *marginal coverage*, where we achieved coverage on average over all the data. Conditional coverage would be a guarantee that we achieve coverage conditional on the input. Formally, conditional coverage requires

$$\mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}})|X_{\text{test}}) \geq 1 - \alpha$$

Unfortunately, conditional coverage is actually impossible to achieve in the most general case – there are certain requirements for the sample size and desired error tolerance, as well as properties of the underlying distribution Barber et al., 2021; Vovk, 2012. Further work has developed some theory and various metrics to evaluate conditional coverage (A. Angelopoulos et al., 2022; Cauchois et al., 2021; Feldman et al., 2021).

3.5 Group-balancing and fairness

A related topic that has garnered substantial recent attention is subgroup fairness in machine learning (Deng et al., 2022; "Fairness in Criminal Justice Risk Assessments: The State of the Art - Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, Aaron Roth, 2021", n.d.; Kearns et al., 2018). There are actually many possible notions of fairness, but for the task at hand we are interested in conditional coverage guarantees for each subgroup (this is similar to controlling type II error within groups in some of the fairness literature). While general conditional coverage cannot be achieved, this special case is feasible as proved by Vovk (2012). Consider a dataset with groups $g = \{1, \ldots, G\}$, and let the first feature of every feature vector be the group label. We express the coverage guarantee as:

$$\mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}})|X_{\text{test},1} = g) \geq 1 - \alpha$$

To accomodate this we compute the conformal scores as usual for each observation, but then we compute the quantiles of the scores for each group separately. That is,

$$\hat{q}_g = \frac{\lceil (n_g + 1)(1 - \alpha) \rceil}{n_g} \text{ quantile of } \{s(X_i, Y_i) : X_{i,1} = g\}$$

Then the prediction set for a new observation X_{test} is:

$$\hat{C}(X_{\text{test}}) = \left\{ y : s(X_{\text{test}}, y) \leq \hat{q}_{X_{\text{test},1}} \right\}$$

This procedure achieves the desired conditional coverage guarantee.

A closely related problem for classification problems is ensuring coverage conditional on the label. That is,

$$\mathbb{P}(Y_{\text{test}} \in C(X_{\text{test}}) | Y_{\text{test}} = y) \geq 1 - \alpha$$

for all $y \in \mathcal{Y}$. This is harder because we do not have access to the labels at test, but the solution is quite similar. We again compute the quantiles of the scores for each label k separately, and then the prediction set for a new observation X_{test} is:

$$\hat{C}(X_{\text{test}}) = \{ y : s(X_{\text{test}}, y) \leq \hat{q}_y \}$$

The key difference is that because we don't know the true class at test time, the quantile $\hat{q}_y$ is different for each provisional label y. So, the scores for each provisional label are compared against different thresholds.

3.6 More general notions of risk

We briefly note that some work has been done to extend conformal prediction to more general notions of risk control, including controlling for multiple risk functions (A. N. Angelopoulos, Bates, et al., 2022; Bates et al., 2021). We will not go into detail here as it requires developing substantial notation and theory for a topic that is a bit outside of the scope of this report, but this is a recent advent in the conformal prediction literature that is a promising direction for future work.

4 Application: Conformal Prediction for Text Classification

The following case study demonstrates how conformal prediction can be used to obtain prediction sets with rigorous coverage guarantees for text classification with a large language model (LLM). We compare the procedure with a naive conformal score to a score which is adaptive to the difficulty of each observation, and show that the adaptive score can lead to smaller prediction sets while preserving valid coverage.

Model. We classify text with the transformer-based language model BART (Lewis et al., 2019), finetuned on the Multi-Genre Natural Language Inference (MNLI) categorized text corpus [1], as proposed by Yin et al. (2019) for zero-shot text classification. The approach is to treat the task of text classification as a natural language inference (NLI) task, where the premise is the text to be classified and

[1] https://huggingface.co/datasets/multi_nli

the hypothesis is a statement about the label like "This text is about politics". The model computes the probability that the hypothesis is true given the premise, and we repeat this for each label independently.

We note that this model is no longer state of the art, but it is more lightweight (400 million parameters) than the larger and more performant models which have gained prominence (e.g. Meta's Llama2 models range from 7-70 billion parameters), and therefore more suitable for fast inference on a laptop. Since we are interested in demonstrating the utility of conformal prediction, we do not need to use the most powerful model available. We will see that this model results in prediction sets which are fairly large, and it is likely that a more powerful model would result give us better heuristic uncertainty estimates and therefore smaller prediction sets.

Data. Our prediction task is to classify text from a new data source, a Kaggle dataset of text documents belonging to 5 new categories[2]. These categories – Politics, Sport, Technology, Entertainment, and Business – are mostly nonoverlapping with the categories in the MNLI corpus. The one exception being Politics, which is similar to the Government genre in MNLI – we will see that this category is, unsurprisingly, the easiest to classify. Of course, BART's original training data surely contained text related to all of these genres (which is why we expect it have enough knowledge to be capable of zero-shot classification), but it was not specifically trained to classify them. We split this Kaggle dataset, using half the data for calibration and the other half for testing.

Code All the code for this project is available at https://github.com/jrudoler/rudoler-stat9700-conformal. This includes the code for language model inference, conformal prediction, and generating the figures in this report. We use the HuggingFace implementation of `bart-large-mnli`[3], and set up a zero-shot text classification pipeline with the `transformers` library (Wolf et al., 2019).

Conformal scores We will compare two different choices of score function s for the conformal prediction procedure:

1. **Nonadaptive**: We just use the predicted probability of each label independently to compute the score. That is, for a given text document X, we compute the score
$$s(X, y) = 1 - \hat{f}(X)_y$$
for each label y.

2. **Adaptive**: We consider the predicted probability of all labels simultaneously in computing the score. Specifically we reorder the labels by descending probability, such that $\hat{f}(X)_{(1)}$ is the most likely label, $\hat{f}(X)_{(2)}$ is the second most likely label, and so on. We define k such that $\hat{f}(X)_{(k)} = y$, and compute the score:
$$s(X, y) = \sum_{j=1}^{k} \hat{f}(X)_{(j)}$$
Basically, we add up the predicted probabilities of all the labels in descending order up to and including the true label, y.

[2]https://www.kaggle.com/datasets/sunilthite/text-document-classification-dataset/data
[3]https://huggingface.co/facebook/bart-large-mnli

Romano et al. (2020) introduce the adaptive score function we use here, and A. Angelopoulos et al. (2022) propose a regularized version and discuss adaptive prediction sets in detail in the context of image classification.

Lastly, we also implement the class-conditional conformal method described in 3.5 to ensure approximate conditional coverage for each label. We still use the adaptive score function, but we compute the quantiles of the scores for each label separately.

Results We evaluate the prediction sets produced via conformal prediction with each of the two score functions described above, and with the conformal procedure which stratifies scores by class label (text category). Figure 1 shows the overall coverage of the prediction sets for each approach, as well as the conditional coverage for each category. Figure 2 shows the average size of the prediction sets for each score function, again both overall and for each category.

The adaptive score function does seem to adapt the size of its prediction sets to ensure that the hardest categories are covered more often. Notice that the "Entertainment" category was initially undercovered, but the adaptive procedure assigns larger prediction sets to this category and ensures coverage. It actually seems to overcompensate a bit, as the "Entertainment" category is now overcovered while previously covered categories are now undercovered. The advantage of adapting to the complex distribution of data is that we maintain marginal coverage while shrinking the average size of the prediction sets, as shown in Figure 2. This is a critical gain, as smaller predictions sets are fundamentally more useful in practice.

One qualification is that these prediction set sizes, overall, are still quite large. This is probably because, as mentioned above, we employ a relatively lightweight model. Consequently, the heuristic uncertainty estimates (the model's predicted label probabilities) are not as accurate as they could be. We expect that a more powerful model would achieve smaller prediction sets.

The last important result is that as expected, neither score function alone achieves conditional coverage for each category. However, the class-conditional conformal procedure improves the conditional coverage dramatically and achieves approximate coverage.

References

Angelopoulos, A., Bates, S., Malik, J., & Jordan, M. I. (2022, September 3). *Uncertainty Sets for Image Classifiers using Conformal Prediction*. arXiv: 2009.14193 [cs, math, stat]. Retrieved November 21, 2023, from http://arxiv.org/abs/2009.14193

Angelopoulos, A. N., & Bates, S. (2022, December 7). *A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification*. arXiv: 2107.07511 [cs, math, stat]. Retrieved November 2, 2023, from http://arxiv.org/abs/2107.07511

Angelopoulos, A. N., Bates, S., Candès, E. J., Jordan, M. I., & Lei, L. (2022, September 29). *Learn then Test: Calibrating Predictive Algorithms to Achieve Risk Control*. arXiv: 2110.01052 [cs, stat]. https://doi.org/10.48550/arXiv.2110.01052

Angelopoulos, A. N., Kohli, A. P., Bates, S., Jordan, M., Malik, J., Alshaabi, T., Upadhyayula, S., & Romano, Y. (2022). Image-to-Image Regression with Distribution-Free Uncertainty Quantification and Applications in Imaging. *Proceedings of the 39th International Conference on Machine Learning*, 717–730. Retrieved November 21, 2023, from https://proceedings.mlr.press/v162/angelopoulos22a.html

Barber, R. F., Candes, E. J., Ramdas, A., & Tibshirani, R. J. (2020, May 29). *Predictive inference with the jackknife+*. arXiv: 1905.02928 [stat]. https://doi.org/10.48550/arXiv.1905.02928

Barber, R. F., Candes, E. J., Ramdas, A., & Tibshirani, R. J. (2023, March 16). *Conformal prediction beyond exchangeability*. arXiv: 2202.13415 [stat]. https://doi.org/10.48550/arXiv.2202.13415

Barber, R. F., Candès, E. J., Ramdas, A., & Tibshirani, R. J. (2021). The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, *10*(2), 455–482. https://doi.org/10.1093/imaiai/iaaa017

Bates, S., Angelopoulos, A., Lei, L., Malik, J., & Jordan, M. (2021). Distribution-free, Risk-controlling Prediction Sets. *Journal of the ACM*, *68*(6), 43:1–43:34. https://doi.org/10.1145/3478535

Cauchois, M., Gupta, S., & Duchi, J. C. (2021). Knowing what You Know: Valid and validated confidence sets in multiclass and multilabel prediction. *Journal of Machine Learning Research*.

Deng, Z., Zhang, J., Zhang, L., Ye, T., Coley, Y., Su, W. J., & Zou, J. (2022, June 6). *FIFA: Making Fairness More Generalizable in Classifiers Trained on Imbalanced Data*. arXiv: 2206.02792 [cs, stat]. https://doi.org/10.48550/arXiv.2206.02792

Fairness in Criminal Justice Risk Assessments: The State of the Art - Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, Aaron Roth, 2021. (n.d.). Retrieved December 14, 2023, from https://journals-sagepub-com.proxy.library.upenn.edu/doi/full/10.1177/0049124118782533

Feldman, S., Bates, S., & Romano, Y. (2021, October 2). *Improving Conditional Coverage via Orthogonal Quantile Regression*. arXiv: 2106.00394 [cs]. https://doi.org/10.48550/arXiv.2106.00394

Gal, Y., & Ghahramani, Z. (2016, October 4). *Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning*. arXiv: 1506.02142 [cs, stat]. https://doi.org/10.48550/arXiv.1506.02142

Hoff, P. (2021, May 28). *Bayes-optimal prediction with frequentist coverage control*. arXiv: 2105.14045 [math, stat]. https://doi.org/10.48550/arXiv.2105.14045

Kearns, M., Neel, S., Roth, A., & Wu, Z. S. (2018, December 3). *Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness*. arXiv: 1711.05144 [cs]. https://doi.org/10.48550/arXiv.1711.05144

Koenker, R., & Bassett, G. (1978). Regression Quantiles. *Econometrica*, *46*(1), 33–50. https://doi.org/10.2307/1913643

Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. *Advances in Neural Information Processing Systems*, *30*. Retrieved November 22, 2023, from https://proceedings.neurips.cc/paper_files/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html

Lei, J., Robins, J., & Wasserman, L. (2013). Distribution Free Prediction Sets. *Journal of the American Statistical Association*, *108*(501), 278–287. https://doi.org/10.1080/01621459.2012.751873

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019, October 29). *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension*. arXiv: 1910.13461 [cs, stat]. https://doi.org/10.48550/arXiv.1910.13461

MacKay, D. J. C. (1992). *Bayesian methods for adaptive models* [Doctoral dissertation, California Institute of Technology]. https://doi.org/10.7907/H3A1-WM07

Nix, D., & Weigend, A. (1994). Estimating the mean and variance of the target probability distribution. *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, *1*, 55–60 vol.1. https://doi.org/10.1109/ICNN.1994.374138

Romano, Y., Patterson, E., & Candès, E. J. (2019, May 8). *Conformalized Quantile Regression*. arXiv: 1905.03222 [stat]. https://doi.org/10.48550/arXiv.1905.03222

Romano, Y., Sesia, M., & Candes, E. (2020). Classification with Valid and Adaptive Coverage. *Advances in Neural Information Processing Systems*, *33*, 3581–3591. Retrieved November 23, 2023, from https://proceedings.neurips.cc/paper/2020/hash/244edd7e85dc81602b7615cd705545f5-Abstract.html

Steinberger, L., & Leeb, H. (2022, May 12). *Conditional predictive inference for stable algorithms*. arXiv: 1809.01412 [math, stat]. https://doi.org/10.48550/arXiv.1809.01412

Tohme, T., Vanslette, K., & Youcef-Toumi, K. (2023). Reliable neural networks for regression uncertainty estimation. *Reliability Engineering & System Safety*, *229*, 108811. https://doi.org/10.1016/j.ress.2022.108811

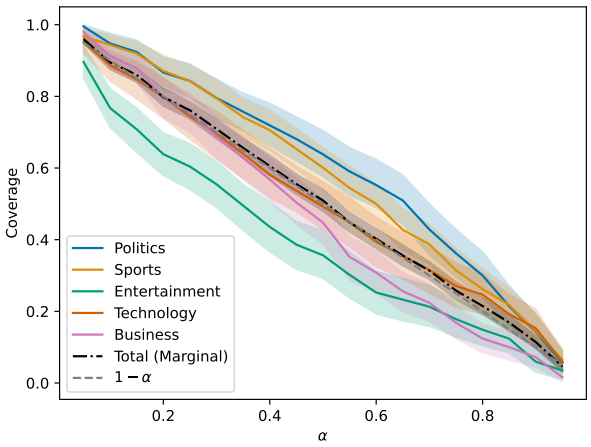
Vovk, V. (2012). Conditional Validity of Inductive Conformal Predictors. *Proceedings of the Asian Conference on Machine Learning*, 475–490. Retrieved December 12, 2023, from https://proceedings.mlr.press/v25/vovk12.html

Vovk, V., Gammerman, A., & Saunders, C. (1999). Machine-Learning Applications of Algorithmic Randomness. *Proceedings of the Sixteenth International Conference on Machine Learning*, 444–453.

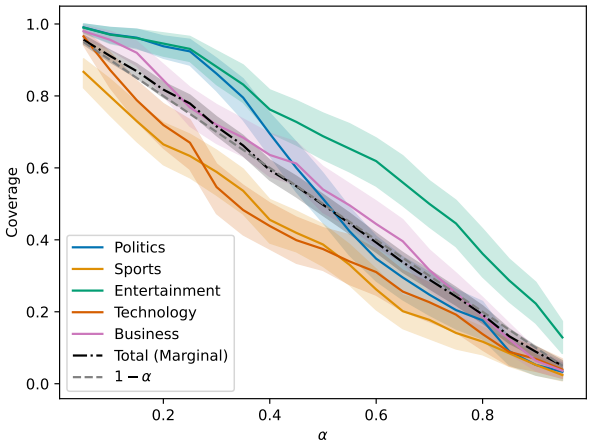
Wilson, A. G., & Izmailov, P. (2020). Bayesian deep learning and a probabilistic perspective of generalization. *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 4697–4708.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., ... Rush, A. M. (2019, October 9). *HuggingFace's Transformers: State-of-the-art Natural Language Processing*. arXiv.org. Retrieved December 12, 2023, from https://arxiv.org/abs/1910.03771v5

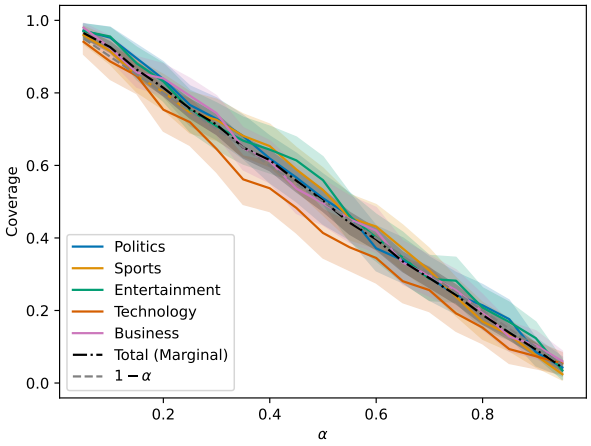
Yin, W., Hay, J., & Roth, D. (2019, August 31). *Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach*. arXiv: 1909.00161 [cs]. https://doi.org/10.48550/arXiv.1909.00161



(a) Nonadaptive conformal prediction

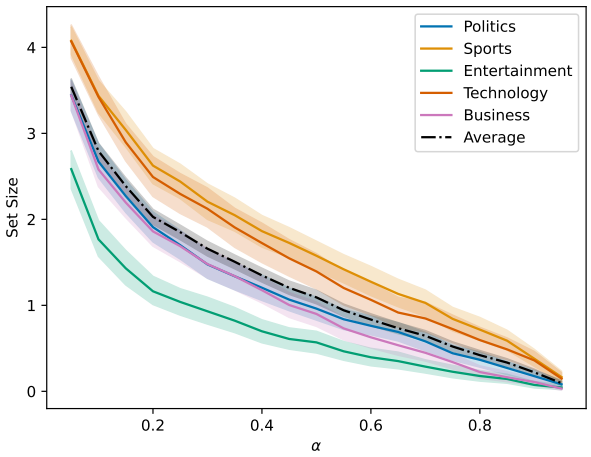


(b) Adaptive conformal prediction

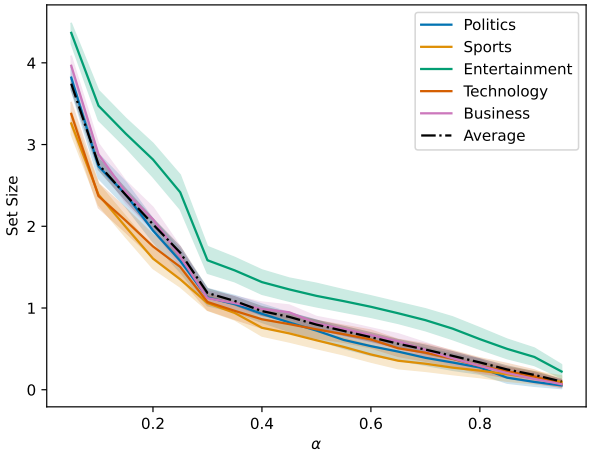


(c) Adaptive conformal prediction with class-conditional coverage

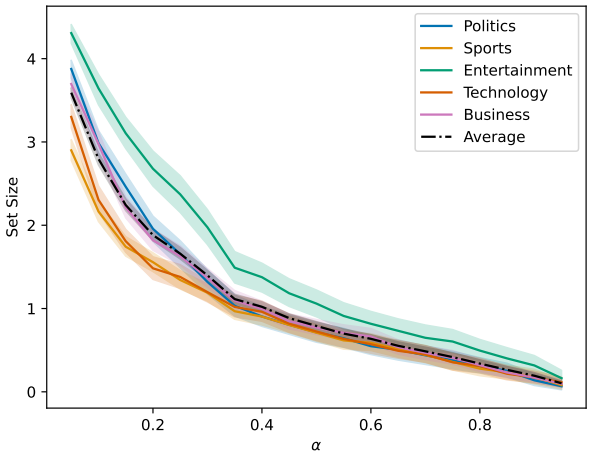
Figure 1: Comparison of coverage for nonadaptive and adaptive conformal prediction for text classification. Both procedures have marginal coverage guarantees of $1 - \alpha$. Only the class-conditional procedure has approximate conditional coverage for each label.



(a) Nonadaptive conformal prediction



(b) Adaptive conformal prediction



(c) Adaptive conformal prediction with class-conditional coverage

Figure 2: Comparison of set size for nonadaptive and adaptive conformal prediction for text classification. The adaptive procedure improves coverage for the hardest categories by making their prediction sets larger.