Aligning Model Properties via Conformal Risk Control

William Overman[1] **Jacqueline Jil Vallon**[2] **Mohsen Bayati**[1]

[1] Stanford Graduate School of Business [2] Management Science and Engineering
{wpo,bayati}@stanford.edu, jjvallon@alumni.stanford.edu

Abstract

AI model alignment is crucial due to inadvertent biases in training data and the underspecified machine learning pipeline, where models with excellent test metrics may not meet end-user requirements. While post-training alignment via human feedback shows promise, these methods are often limited to generative AI settings where humans can interpret and provide feedback on model outputs. In traditional non-generative settings with numerical or categorical outputs, detecting misalignment through single-sample outputs remains challenging, and enforcing alignment during training requires repeating costly training processes. In this paper we consider an alternative strategy. We propose interpreting model alignment through property testing, defining an aligned model f as one belonging to a subset $\mathcal{P}$ of functions that exhibit specific desired behaviors. We focus on post-processing a pre-trained model f to better align with $\mathcal{P}$ using conformal risk control. Specifically, we develop a general procedure for converting queries for testing a given property $\mathcal{P}$ to a collection of loss functions suitable for use in a conformal risk control algorithm. We prove a probabilistic guarantee that the resulting conformal interval around f contains a function approximately satisfying $\mathcal{P}$. We exhibit applications of our methodology on a collection of supervised learning datasets for (shape-constrained) properties such as monotonicity and concavity. The general procedure is flexible and can be applied to a wide range of desired properties. Finally, we prove that pre-trained models will always require alignment techniques even as model sizes or training data increase, as long as the training data contains even small biases.

1 Introduction

The emergence of large foundation models has increased the attention to the problem of alignment. Aligned models are artificial intelligences designed to pursue goals that align with human values, principles, and intentions (Leike et al., 2018; Ouyang et al., 2022; Hendrycks et al., 2023; Ngo et al., 2024). Although the alignment problem is predominantly examined in the context of potential artificial general intelligence (AGI), large language models (LLMs), and reinforcement learning (RL) agents, it also has roots in the modern machine learning pipeline (D'Amour et al., 2022). Motivated by this, we introduce a broader notion of alignment in this paper, extending beyond the aforementioned generative models to include even tabular regression models.

As an example, consider a regression task where property $\mathcal{P}$ represents models that are monotonically decreasing in a given feature (covariate). For example, predicting cancer patient survival should be monotonically decreasing in cancer stage (Vallon et al., 2022, 2024). Constraining a prediction model during training to maintain monotonicity in this feature can be viewed as a form of alignment. For a pre-trained model f that was trained without such constraints, ensuring monotonically decreasing predictions in this feature can be more complex. This complexity arises particularly in non-generative

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

settings where the user cannot update f or obtain any outputs other than point predictions $f(X)$ for a given input X.

In this work, we propose an approach to aligning a pre-trained model f that is motivated by property testing (Ron, 2008; Goldreich, 2017) and conformal risk control (Angelopoulos et al., 2024). Property testing aims to design efficient algorithms for determining membership to the set $\mathcal{P}$ of functions with a given property, that require fewer resources than learning algorithms for $\mathcal{P}$ (Ron, 2008). This is particularly relevant for modern deep learning, where a user may need to determine if a pre-trained model f belongs to $\mathcal{P}$ without the resources to train a model of comparable size.

Property testing algorithms use local queries to determine, with high probability, whether a function has a given global property or is far from having it. We map such queries for a property $\mathcal{P}$ to a set of loss functions, which we then use in a conformal risk control procedure (Angelopoulos et al., 2024) to establish a notion of alignment for $\mathcal{P}$. We prove that this procedure yields a conformal interval around f containing a function close to $\mathcal{P}$.

We demonstrate our methodology on real-world datasets for the properties of monotonicity and concavity. Motivated by the potential for systematic under- or over-estimation bias in f, we provide a straightforward extension of Angelopoulos et al. (2024) to obtain asymmetric conformal intervals with multi-dimensional parameters. While we examine both monotonicity and concavity constraints, the majority of our focus is on monotonicity, as these constraints have been shown to promote crucial aspects of alignment to human values, such as fairness and adherence to social norms (Wang and Gupta, 2020).

While our methodology provides a way to align pre-trained models, one may question whether such techniques will remain necessary as AI capabilities advance. Given the outstanding capabilities of modern AI models with substantially large numbers of parameters and training data, one may argue that the alignment problem may naturally disappear as such advances continue (Kaplan et al., 2020). However, another contribution of this paper is to refute this argument in a stylized setting, building on recent advances in the theory of linearized neural networks (Mei and Montanari; Misiakiewicz and Montanari, 2023). Specifically, we show that increasing the size of the training data or the number of parameters in a random feature model (a theoretically tractable neural network proxy where hidden layer weights are randomly initialized and fixed (Rahimi and Recht, 2007)) cannot help it satisfy a property $\mathcal{P}$, if the pre-training data has biased labels. Our simulations show that the result holds even if only a small fraction of the training labels are impacted by the bias.

Summarizing our main contributions, we: (1) introduce an alignment perspective based on property testing, (2) use conformal risk control to post-process predictions of pre-trained models for better alignment, and (3) demonstrate that increasing training data and parameters in a random feature model does not eliminate the need for alignment. We discuss related work in Section 6, particularly our connections to Yadkori et al. (2024), who use conformal risk control to address large language model hallucinations (Ji et al., 2023).

2 Preliminaries

In this section we provide key definitions drawn from property testing as well as a condensed overview to conformal prediction and conformal risk control. We provide a short introduction to propery testing in the extended version of the paper Overman et al. (2024) and an extensive introduction to the field can be found in Goldreich (2017).

2.1 Properties and Property Testing for Set-Valued Functions

Our perspective on alignment in this work is motivated by the field of property testing (Goldreich, 2017; Ron, 2008). Property testing studies algorithms that, by making a small number of local queries to a large object (such as a function or a graph), can determine whether the object has a certain property or is significantly far from having it.

Classic examples include linearity testing of Boolean functions (Blum et al., 1993), testing whether a function is a low-degree polynomial (Kaufman and Ron, 2006; Bhattacharyya et al., 2009), and testing k-juntas (Blais, 2009). These algorithms generally operate by randomly sampling and querying the object, leveraging local information to infer global properties.

In this work, we focus on *set-valued functions*, which are functions that map elements of a domain $\mathcal{X}$ to subsets of a codomain $\mathcal{Y}$, i.e., $F : \mathcal{X} \to 2^{\mathcal{Y}}$. While the standard definitions of property testing are technically sufficient for our purposes—since we can consider set-valued functions as functions with range $2^{\mathcal{Y}}$—we introduce specialized definitions to maintain clarity and to facilitate the transition between discussing $\mathcal{Y}$ and $2^{\mathcal{Y}}$.

Definition 1 (Satisfying and Accommodating a Property)**.** *Let **property** $\mathcal{P}$ denote a specific subset of all functions that map $\mathcal{X}$ to $\mathcal{Y}$. A function $f : \mathcal{X} \to \mathcal{Y}$ **satisfies** the property $\mathcal{P}$ if $f \in \mathcal{P}$.*

*A set-valued function $F : \mathcal{X} \to 2^{\mathcal{Y}}$ **accommodates** a property $\mathcal{P}$ if there exists a function $g \in \mathcal{P}$ such that $g(x) \in F(x)$ for all $x \in \mathcal{X}$.*

Intuitively, F accommodates $\mathcal{P}$ if it contains at least one function g satisfying $\mathcal{P}$ within its possible outputs.

We extend the notion of ε-farness from a property (as defined in the extended version of the paper Overman et al. (2024)) to set-valued functions. For set-valued functions, we measure the distance based on how often the outputs of any function $g \in \mathcal{P}$ fall within the sets provided by F.

Definition 2 (ε-Faraway)**.** *For a set-valued function $F : \mathcal{X} \to 2^{\mathcal{Y}}$, a distribution $\mathcal{D}$ over $\mathcal{X}$, $\varepsilon > 0$, and a property $\mathcal{P}$, we say F is ε-**Faraway** from $\mathcal{P}$ with respect to $\mathcal{D}$ if $\delta_{\mathcal{P},\mathcal{D}}(F) > \varepsilon$, where*

$$\delta_{\mathcal{P},\mathcal{D}}(F) \stackrel{def}{=} \inf_{g \in \mathcal{P}} \delta_{\mathcal{D}}(F, g) \quad \text{and} \quad \delta_{\mathcal{D}}(F, g) \stackrel{def}{=} \Pr_{X \sim \mathcal{D}}[g(X) \notin F(X)].$$

Note. Throughout this work, we assume that $\mathcal{D}$ is the empirical distribution of a fixed and finite calibration dataset, and thus has finite support. While this assumption is not strictly necessary, most property testing results are over finite domains. Property testing over functions with Euclidean domains is a in general a difficult problem, though there have been notable recent successes (Fleming and Yoshida, 2020; Arora et al., 2023)..

With these definitions in place, we can define *testers* for set-valued functions. We focus on *one-sided error testers*, which are algorithms that take in a set-valued function F, a distribution $\mathcal{D}$, and a distance parameter ε and output either Accept or Reject. These algorithms never reject a function that accommodates the property. The standard definition of one-sided error testers (provided in the extended version of the paper Overman et al. (2024)) extends naturally to set-valued functions by replacing the notion of satisfying a property with accommodating it.

Definition 3 (One-Sided Error Tester for Set-Valued Functions)**.** *A* one-sided error tester *for a property $\mathcal{P}$ in the context of set-valued functions is a probabilistic oracle machine $\mathcal{M}$ that, given a distance parameter $\varepsilon > 0$, oracle access to a set-valued function $F : \mathcal{X} \to 2^{\mathcal{Y}}$, and oracle access to samples from a fixed but unknown distribution $\mathcal{D}$ over $\mathcal{X}$, satisfies:*

1. *If F accommodates $\mathcal{P}$, then $\Pr[\mathcal{M}^{F,\mathcal{D}}(\varepsilon) = \textit{Accept}] = 1$.*
2. *If F is ε-Faraway from $\mathcal{P}$ with respect to $\mathcal{D}$, then $\Pr[\mathcal{M}^{F,\mathcal{D}}(\varepsilon) = \textit{Accept}] \leq \frac{1}{3}$.*

Here, $\mathcal{M}^{F,\mathcal{D}}(\varepsilon)$ denotes the execution of the tester $\mathcal{M}$ when given oracle access to the function F, the distribution $\mathcal{D}$, and the parameter ε.

Note that $\mathcal{M}$ itself is an abstract algorithm; $\mathcal{M}^{F,\mathcal{D}}$ is the instantiation of this algorithm with specific oracle access to F and $\mathcal{D}$.

In many property testing algorithms, the parameter ε is used only to determine the number of iterations or samples required, not the core logic of the tester. This leads to the concept of *proximity-oblivious testers* (POTs), where the basic testing procedure is independent of ε. The general definition of POTs (given in the extended version of the paper Overman et al. (2024)) also extends naturally to set-valued functions.

Definition 4 (Proximity-Oblivious Tester for Set-Valued Functions)**.** *A* proximity-oblivious tester *for a property $\mathcal{P}$ in the context of set-valued functions is a probabilistic oracle machine $\mathcal{T}$ that satisfies:*

1. *If F accommodates $\mathcal{P}$, then $\Pr[\mathcal{T}^{F,\mathcal{D}} = \textit{Accept}] = 1$*
2. *There exists a non-decreasing function $\rho : (0, 1] \to (0, 1]$ (called the* detection probability*) such that if F is ε-Faraway from $\mathcal{P}$,*

$$\Pr[\mathcal{T}^{F,\mathcal{D}} = \textit{Reject}] \geq \rho(\varepsilon).$$

Here, $\mathcal{T}^{F,\mathcal{D}}$ denotes the execution of the tester $\mathcal{T}$ when given oracle access to the function F and the distribution $\mathcal{D}$.

To obtain a one-sided error tester with parameter ε, we can make $\Theta\left(\frac{1}{\rho(\varepsilon)}\right)$ independent calls to the POT $\mathcal{T}$ and accept if and only if all the calls accept (Goldreich and Ron, 2008). We denote by $\mathcal{T}^{F,\mathcal{D}}(X)$ the output when applied to a specific sample $X \sim \mathcal{D}$, and note that with abuse of notation we will late consider $\mathcal{D}$ to be the empirical distribution of calibration dataset $\{(X_i, Y_i)\}_{i=1}^n$ in which case we write $\mathcal{T}^{F,\mathcal{D}}(X_i, Y_i)$ for the output on this specific sample from $\mathcal{D}$.

Example. Consider functions $f : \mathbb{R}^d \to \mathbb{R}$, and let $\mathcal{P}$ denote the property that f is constant in the k-th dimension. This property has has connections to fairness among other applications Caton and Haas (2024). Assume $\mathcal{D}$ is the empirical distribution of the inputs $X \in \mathbb{R}$ for some fixed dataset.

Restrict to set-valued functions F that output compact and connected intervals of the form $[a, b] \subseteq \mathbb{R}$ for $a, b \in \mathbb{R}$. The candidate POT $\mathcal{T}^{F,\mathcal{D}}$ for whether such a set-valued function F accommodates $\mathcal{P}$ is then as follows: sample $X, X' \sim \mathcal{D}$, If $F(X) \cap F(X') \neq \varnothing$, then Accept; otherwise, Reject. We prove that this satisfies Definition 4 in the extended version of the paper Overman et al. (2024).

2.2 Conformal prediction and conformal risk control

Our main tool for achieving alignment from this property perspective is built on conformal prediction and conformal risk control (Vovk et al., 2005; Bates et al., 2021; Angelopoulos et al., 2024). Conformal prediction post-processes the outputs of any model f to create prediction intervals $C(\cdot)$ that ensure certain statistical coverage guarantees. Using a calibration dataset $\{(X_i, Y_i)\}_{i=1}^n$ consisting of ground truth input-output pairs, conformal prediction constructs intervals around the predictions of f such that $\Pr[Y_{n+1} \notin C(X_{n+1})] \leq \alpha$ for a user-specified error rate α on a test point (X_{n+1}, Y_{n+1}).

This guarantee is notably distribution-free and holds for any function f. The probability is over the randomness in all $n + 1$ points; both the calibration set and the test point. The construction of $C(\cdot)$ depends on both the model f and the draw of the calibration data.

The conformal risk control framework extends conformal prediction to notions of error beyond miscoverage (Angelopoulos et al., 2024). Consider a paramater set $\Lambda \subset \mathbb{R}_{\geq 0}$ that is a bounded subset of the nonnegative reals. Given an exchangeable collection of non-increasing, random loss functions $L_i : \Lambda \to (-\infty, B]$, $i = 1, \ldots, n + 1$, conformal risk control uses the first n loss functions and calibration data $\{(X_i, Y_i)\}_{i=1}^n$ to determine $\hat{\lambda}$ such that

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

Consider loss functions of the form $L_i(\lambda) = \ell(C_\lambda(X_i), Y_i)$, where $C_\lambda(X_i)$ is a set of outputs constructed by f and the calibration data. Larger values of λ generate more conservative prediction sets $C_\lambda(\cdot)$. Let the risk on the calibration data for a given λ be $\hat{R}_n(\lambda) = \frac{1}{n}\sum_{i=1}^n L_i(\lambda)$. For a user-specified risk rate α, we let

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

This choice of $\hat{\lambda}$ guarantees the desired risk control $\mathbb{E}[L_{n+1}(\hat{\lambda})] \leq \alpha$ (Angelopoulos et al., 2024).

3 Conformal property alignment

Our main methodology is to use conformal risk control to create prediction intervals that align with specific properties $\mathcal{P}$. Our approach allows us to post-process the outputs of a pre-trained model f to ensure that within the resulting conformal band, with a given probability, there exists predictions that adhere to desired properties such as monotonicity.

3.1 Multi-lambda conformal risk control

We make particular use of the conformal risk control algorithm to allow for a k-dimensional vector of tuning parameters $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_k)$, where larger values of $\boldsymbol{\lambda} \in \boldsymbol{\Lambda} \subset \mathbb{R}^k$ yield more

conservative outputs, where $\mathbf{\Lambda} \subset \mathbb{R}^k_{\geq 0}$ is a bounded subset of $\mathbb{R}^k_{\geq 0}$. This works by mapping $\boldsymbol{\lambda}$ to a scalar and then applying standard conformal risk control. We emphasize that this result is not new and follows essentially directly from Angelopoulos et al. (2024). The construction of the output set $F_{\boldsymbol{\lambda}}(X) \subseteq \mathcal{Y}$ depends on the specific application and provides flexibility in how the function $f(X)$ and the parameters $\boldsymbol{\lambda}$ are utilized.

Definition 5 (Construction of $F_{\boldsymbol{\lambda}}(X)$)**.** *Let $f : \mathcal{X} \to \mathcal{Y}$ be a given function. For each $\boldsymbol{\lambda} \in \mathbb{R}^k$, define the set-valued function $F_{\boldsymbol{\lambda}} : \mathcal{X} \to 2^{\mathcal{Y}}$ such that, for each $X \in \mathcal{X}$, $F_{\boldsymbol{\lambda}}(X)$ is a set of predictions for X constructed from f and $\boldsymbol{\lambda}$. The specific construction of $F_{\boldsymbol{\lambda}}(X)$ should satisfy the following properties:*

1. *When $\boldsymbol{\lambda} = \mathbf{0}$, we have $F_{\mathbf{0}}(X) = \{f(X)\}$.*
2. *For any $\boldsymbol{\lambda}, \boldsymbol{\lambda}' \in \mathbb{R}^k$, if $\boldsymbol{\lambda} \leq \boldsymbol{\lambda}'$ (i.e., $\lambda_i \leq \lambda'_i \; \forall i = 1, 2, \ldots, k$), then $F_{\boldsymbol{\lambda}}(X) \subseteq F_{\boldsymbol{\lambda}'}(X)$.*

This definition ensures that increasing the parameters $\boldsymbol{\lambda}$ leads to larger (more conservative) prediction sets, and that when all parameters are zero, the prediction set reduces to the point prediction given by $f(X)$.

Following the original scalar λ setting, we assess $F_{\boldsymbol{\lambda}}$ using non-increasing random loss functions $L_i = \ell(F_{\boldsymbol{\lambda}}(X_i), Y_i) \in (-\infty, B]$ for $B < \infty$. In particular, we consider an exchangeable collection of non-increasing random functions $L_i : \mathbf{\Lambda} \to (-\infty, B], i = 1, ..., n+1$, where $\mathbf{\Lambda} \subset \mathbb{R}^k_{\geq 0}$ is a bounded subset of $\mathbb{R}^k_{\geq 0}$, with bound $\lambda_j^{\max}$ in each dimension $j \in [k]$.

As in Angelopoulos et al. (2024), we use the first n functions to determine $\hat{\boldsymbol{\lambda}}$ so that the risk on the $(n+1)$-th function is controlled, specifically so that $\mathbb{E}[L_{n+1}(\hat{\boldsymbol{\lambda}})] \leq \alpha$.

We apply a similar algorithm. Given $\alpha \in (\infty, B)$ and letting $\hat{R}_n(\boldsymbol{\lambda}) = \frac{L_1(\boldsymbol{\lambda}) + \cdots + L_n(\boldsymbol{\lambda})}{n}$, define

$$\mathbf{\Lambda}_{\min} = \min \left\{ \boldsymbol{\lambda} : \frac{n}{n+1} \hat{R}_n(\boldsymbol{\lambda}) + \frac{B}{n+1} \leq \alpha \right\}$$

to be the set of minimal elements (Boyd and Vandenberghe, 2004) of $\mathbf{\Lambda}$ that satisfy the condition $\frac{n}{n+1} \hat{R}_n(\boldsymbol{\lambda}) + \frac{B}{n+1} \leq \alpha$. Let $g : \mathbf{\Lambda} \to \mathbb{R}$ be a strictly increasing function such that $L_i(\boldsymbol{\lambda})$ is non-increasing with respect to the level sets defined by $g(\boldsymbol{\lambda})$. Then select $\hat{\boldsymbol{\lambda}} \in \mathbf{\Lambda}_{\min}$ to be a minimizer of g over $\mathbf{\Lambda}_{\min}$.

We then deploy the resulting set-valued function $F_{\hat{\boldsymbol{\lambda}}}$ on the test point X_{n+1}. For this choice of $\hat{\boldsymbol{\lambda}}$, we have a risk control guarantee that mimics the result of Angelopoulos et al. (2024), specifically:

Proposition 1. *Assume that $L_i(\boldsymbol{\lambda})$ is non-increasing with respect to the partial ordering of $\mathbf{\Lambda}$ inherited from $\mathbb{R}^k$. Additionally, assume that $L_i(\boldsymbol{\lambda})$ is non-increasing with respect to $g(\boldsymbol{\lambda})$ for some strictly increasing function $g : \mathbf{\Lambda} \to \mathbb{R}$. Also assume L_i is right-continuous in each dimension, $L_i(\boldsymbol{\lambda}^{\max}) \leq \alpha$, and $\sup_{\boldsymbol{\lambda}} L_i(\boldsymbol{\lambda}) \leq B < \infty$ almost surely. Then*

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

The proof is similar to the proof of the guarantee for the conformal risk control algorithm in Angelopoulos et al. (2024) and is deferred to the extended version of the paper Overman et al. (2024)

To provide intuition on $g(\boldsymbol{\lambda})$, we note that for our primary use case we will take $g(\boldsymbol{\lambda}) = \sum_{i=1}^k \lambda_i$. Clearly this function is strictly increasing in $\boldsymbol{\lambda}$ and intuitively it is reasonable to consider loss functions L_i that are non-increasing as the sum of the components of $\boldsymbol{\lambda}$ increases.

3.2 Conformal property alignment from proximity oblivious testers

We now demonstrate how to construct a conformal risk control problem using proximity-oblivious testers (POTs) for a given property $\mathcal{P}$. Suppose we are given a pre-trained model $f : \mathcal{X} \to \mathcal{Y}$. We aim to extend the point predictions of f to prediction sets, where the size or conservativeness of the set is parameterized by a parameter $\boldsymbol{\lambda}$. Let $F_{\boldsymbol{\lambda}} : \mathcal{X} \to 2^{\mathcal{Y}}$ denote the set-valued function that outputs, for each $X \in \mathcal{X}$, the set $F_{\boldsymbol{\lambda}}(X) \subseteq \mathcal{Y}$ determined by f, X, and $\boldsymbol{\lambda}$.

Let $\mathcal{T}^{F,\mathcal{D}}$ be a proximity-oblivious tester for whether a set-valued function F accommodates the property $\mathcal{P}$ as given y Definition 4. We denote the random output of $\mathcal{T}^{F,\mathcal{D}}$ evaluated at $(X,Y) \sim \mathcal{D}$ by $\mathcal{T}^{F,\mathcal{D}}(X,Y)$.

We now define a loss function, generated from $\mathcal{T}^{F,\mathcal{D}}$, which will be crucial in formulating our conformal risk control problem.

Definition 6 (Loss Function Generated from a POT)**.** *Let $\mathcal{T}^{F,\mathcal{D}}$ be a proximity-oblivious tester for a property $\mathcal{P}$. We define the loss function L_i as:*

$$L_i = \begin{cases} 0, & \text{if } \mathcal{T}^{F,\mathcal{D}}(X_i,Y_i) = \textit{Accept}, \\ 1, & \textit{otherwise}, \end{cases}$$

where (X_i,Y_i) are samples from the distribution $\mathcal{D}$.

Example. Consider the POT for the property $\mathcal{P}$ of a function $f : \mathbb{R} \to \mathbb{R}$ being constant, as mentioned in Section 2.1. Assume we have access to a calibration set $\{(X_i,Y_i)\}_{i=1}^n$ of size n. We use a two-dimensional parameter $\boldsymbol{\lambda} = (\lambda^-, \lambda^+)$, and define the set-valued function:

$$F_{\boldsymbol{\lambda}}(X) = [f(X) - \lambda^-,\ f(X) + \lambda^+],$$

for each $X \in \mathbb{R}$. This creates prediction intervals around the point prediction $f(X)$, with widths controlled by λ^- and λ^+.

We then apply the loss function generated by $\mathcal{T}^{F_\lambda,\mathcal{D}}$ as given in Definition 6, and use conformal risk control to tune $\boldsymbol{\lambda}$ such that the expected loss on the $(n+1)$th point falls below a given target level α.

Note that in this case the tester and loss function does not depend on the Y_i. This is because the property of f being constant does not depend on the Y_i from the calibration set and here $\mathcal{D}$ is only used to obtain samples of the X_i. This is not the case in general, however, and properties can be defined with respect to the whole sample $(X_i,Y_i) \sim \mathcal{D}$. For example, we could consider the property $\mathcal{P}$ that f does not over-predict, that is, for $(X,Y) \sim \mathcal{D}$ we have $f(X) \le Y$. Now we state our main theorem.

Theorem 1. *Let $\mathcal{T}$ be a proximity-oblivious tester for a property $\mathcal{P}$ with detection probability function $\rho(\cdot)$. Assume access to a calibration dataset $\{(X_i,Y_i)\}_{i=1}^n$ sampled independently from a distribution $\mathcal{D}$. Suppose we run conformal risk control on this calibration dataset using risk parameter α and loss functions L_i for property $\mathcal{P}$ generated from $\mathcal{T}$ (as in Definition 6). Then, for any ε such that $\rho(\varepsilon) > \alpha$, the probability that $F_{\hat{\boldsymbol{\lambda}}}$ is ε-Faraway from $\mathcal{P}$ satisfies:*

$$\Pr_{(X_1,Y_1),\ldots,(X_n,Y_n)} \left(F_{\hat{\boldsymbol{\lambda}}} \textit{ is } \varepsilon\textit{-Faraway from } \mathcal{P}\right) \le \frac{\alpha}{\rho(\varepsilon)}.$$

Proof. Let $\mathcal{E}$ denote the event that $F_{\hat{\boldsymbol{\lambda}}}$ is ε-Faraway from the property $\mathcal{P}$. Our goal is to bound the probability $\Pr_{(X_1,Y_1),\ldots,(X_n,Y_n)}[\mathcal{E}]$.

The conformal risk control procedure ensures that the expected loss on a new sample (X_{n+1},Y_{n+1}) satisfies:

$$\mathbb{E}_{(X_1,Y_1),\ldots,(X_n,Y_n),(X_{n+1},Y_{n+1})}[L_{n+1}] \le \alpha.$$

Now, we can write

$$\mathbb{E}[L_{n+1}] = \Pr(\mathcal{E}) \cdot \mathbb{E}[L_{n+1} \mid \mathcal{E}] + \Pr(\mathcal{E}^c) \cdot \mathbb{E}[L_{n+1} \mid \mathcal{E}^c].$$

When $\mathcal{E}$ occurs, $F_{\hat{\boldsymbol{\lambda}}}$ is ε-Faraway from $\mathcal{P}$. By the properties of the proximity-oblivious tester $\mathcal{T}$, we have:

$$\Pr_{(X,Y)\sim\mathcal{D}} \left[\mathcal{T}^{F_{\hat{\boldsymbol{\lambda}}},\mathcal{D}}(X,Y) = \text{Reject} \mid \mathcal{E}\right] \ge \rho(\varepsilon).$$

Thus, the conditional expected loss satisfies:

$$\mathbb{E}[L_{n+1} \mid \mathcal{E}] = \Pr\left[\mathcal{T}^{F_{\hat{\boldsymbol{\lambda}}},\mathcal{D}}(X_{n+1},Y_{n+1}) = \text{Reject} \mid \mathcal{E}\right] \ge \rho(\varepsilon).$$

And when since $\Pr(\mathcal{E}^c) \cdot \mathbb{E}[L_{n+1} \mid \mathcal{E}^c]$ is non-negative because L_{n+1} is non-negative, we obtain

$$\mathbb{E}[L_{n+1}] \ge \Pr(\mathcal{E}) \cdot \rho(\varepsilon).$$

Combining this result with our guarantee from the conformal risk control procedure,

$$\alpha \geq \mathbb{E}[L_{n+1}] \geq \Pr(\mathcal{E}) \cdot \rho(\varepsilon).$$

This implies:

$$\Pr(\mathcal{E}) \leq \frac{\alpha}{\rho(\varepsilon)}.$$

Therefore, the probability that $F_{\hat{\boldsymbol{\lambda}}}$ is ε-Faraway from $\mathcal{P}$ satisfies:

$$\Pr_{(X_1,Y_1),\ldots,(X_n,Y_n)} \left(F_{\hat{\boldsymbol{\lambda}}} \text{ is } \varepsilon\text{-Faraway from } \mathcal{P}\right) \leq \frac{\alpha}{\rho(\varepsilon)}.$$

□

Amplifying Detection Probability via Independent Calls. When the detection probability $\rho(\varepsilon)$ of the proximity-oblivious tester $\mathcal{T}$ is less than or close to the risk parameter α, the bound provided by Theorem 1 may not be tight or meaningful (since $\alpha/\rho(\varepsilon)$ could be greater than or equal to 1). To address this issue, we can amplify the detection probability by performing multiple independent executions of $\mathcal{T}$ and combining their results appropriately.

To increase the detection probability beyond α, we execute the proximity-oblivious tester $\mathcal{T}$ independently k times on independent samples and define a new tester $\mathcal{T}'$ that rejects if *any* of the k executions reject (i.e., by applying a logical OR to the outcomes). This amplification technique yields an adjusted detection probability

$$\rho'(\varepsilon) = 1 - (1 - \rho(\varepsilon))^k,$$

representing the probability that at least one of the k independent executions rejects when the function is ε-Faraway from $\mathcal{P}$.

In this approach, the calibration dataset needs to be partitioned into $n' = \left\lfloor \frac{n}{k} \right\rfloor$ disjoint batches, each containing k samples. Each batch provides the independent samples required for the k executions of $\mathcal{T}$ per calibration point. As a result, the effective sample size available for calibrartion becomes n' due to this batching of samples.

4 Examples

4.1 Monotonicity

Monotonic behavior is important in various applications. We focus on monotonicity in a single feature, where we expect that $f(X)$ should have monotonically increasing or decreasing behavior with respect to a certain feature x^k when other features x^{-k} are held fixed. While there is a long-standing literature on using monotonic constraints for regularization (Brunk et al., 1973; Sill and Abu-Mostafa, 1996; You et al., 2017; Bonakdarpour et al., 2018) and on integrating such monotonic shape constraints into prediction models (Groeneboom and Jongbloed, 2014; Cano et al., 2018; Runje and Shankaranarayana, 2023), our aim is not to view monotonicity as a possible means to improve test accuracy, but rather as a user-desired property for safe or fair deployment of a given model. For example, Wang and Gupta (2020) highlight the importance of monotonicity in models for criminal sentencing, wages, and medical triage.

Consider a user given a pre-trained model f that was not trained with monotonic constraints. The user, however, wishes for the sake of safe or fair deployment to make predictions in a way that is as monotonic as possible. In particular, let $\mathcal{P}$ be the property that f is monotonically decreasing in dimension k. To apply our methodology we consider the proximity oblivious tester $\mathcal{T}$ for $\mathcal{P}$ as given in Algorithm 1.

We prove in the extended version of the paper Overman et al. (2024) that Algorithm 1 is indeed a POT for the property $\mathcal{P}$ of being monotonically decreasing in a given dimension. Then let $\mathcal{M}$ be the one-sided error tester for $\mathcal{P}$ resulting from $\Theta(1/\rho(\varepsilon))$ calls to $\mathcal{T}$. Now assume we have access to a calibration dataset $\{(X_i, Y_i)\}_{i=1}^n$ sampled from $\mathcal{D}$ of size $n \in \Omega(1/\rho(\varepsilon))$. We will use this calibration dataset to determine the setting of $\boldsymbol{\lambda} = (\lambda^+, \lambda^-)$ via conformal risk control where the loss function is generated as in Definition 6. Here the set-valued function will be constructed as $F_{\boldsymbol{\lambda}}(X) = [f(X) - \lambda^-, f(X) + \lambda^+]$. Then by Theorem 1 if the tester has sufficient detection probability $\rho(\varepsilon) > \alpha$ we expect to obtain a set-valued function $F_{\hat{\boldsymbol{\lambda}}}$ at most ε from $\mathcal{P}$. We now investigate this empirically.

Algorithm 1 POT $\mathcal{T}$ for property $\mathcal{P}$ of monotonically decreasing in dimension k

Sample $X_1 \sim \mathcal{D}$. Let $X_1 = (x_1, x^{-k})$
Sample x_2 from the marginal distribution of $\mathcal{D}$ in dimension k. Set $X_2 = (x_2, x^{-k})$
if $x_1 < x_2$ and $\max F(X_1) < \min F(X_2)$ **then**
    return Reject
else if $x_2 < x_1$ and $\max F(X_2) < \min F(X_1)$ **then**
    return Reject
end if
return Accept

Setup. We align for monotonicity on various UCI ML repository datasets (Dua and Graff, 2023) with a 70-15-15 train-calibrate-test split, averaged over 30 random splits. We use XGBoost regression models (Chen and Guestrin, 2016). For each dataset, we select a feature for which we desire the model to be monotonic, not with the intention of improving test-set accuracy, but from the perspective of a user who desires this property.

We train two models per dataset: one unconstrained, trained on the training set, and another constrained to be monotonic, trained on both the training and calibration sets. The conformal risk control procedure is applied to the unconstrained model using the calibration data. The constrained model can be considered best possible from the user's perspective, using all available pre-test data and satisfying the monotonicity property $\mathcal{P}$ during training.

To compare performance with respect to the training metric of accuracy, we convert conformal intervals into point predictions by taking k-quantiles of the constrained feature, linearly interpolating between adding λ^+ at the lowest quantile to subtracting λ^- at the highest quantile for monotonically decreasing, or vice versa for monotonically increasing.

Results. Table 4.1 presents results on the test set for the Combined Cycle Power Plant dataset (Tfekci and Kaya, 2014). In practice, Exhaust-vacuum is known to negatively influence turbine efficiency (Tfekci and Kaya, 2014). The conformal procedure outperforms the constrained model in terms of MSE for all α, which is a fortuitous but unexpected outcome. The constrained model should be seen as an oracle benchmark in the sense that the model was given to the user already trained to satisfy the desired property. The risk metric closely matches the theoretical guarantee from conformal risk control and achieves optimal performance of 0 for the constrained model. Additional datasets and results are detailed in the extended version of the paper Overman et al. (2024).

Table 1: Power Plant, $n = 9568$. Monotonically decreasing on Exhaust Vacuum. $\boldsymbol{\lambda}^{\max} = (10, 10)$.

α	$\boldsymbol{\lambda}$	Metric	Unconstrained	Adjusted	Constrained
0.1	$\lambda^+ = 0.51_{(\pm 0.24)}$	MSE	$10.19_{(\pm 0.46)}$	$10.47_{(\pm 0.46)}$	$16.21_{(\pm 0.45)}$
	$\lambda^- = 0.76_{(\pm 0.24)}$	Risk	$0.75_{(\pm 0.09)}$	$0.10_{(\pm 0.001)}$	$0.00_{(\pm 0.00)}$
0.05	$\lambda^+ = 1.09_{(\pm 0.51)}$	MSE	$10.19_{(\pm 0.46)}$	$11.42_{(\pm 0.44)}$	$16.21_{(\pm 0.45)}$
	$\lambda^- = 1.61_{(\pm 0.50)}$	Risk	$0.75_{(\pm 0.09)}$	$0.05_{(\pm 0.001)}$	$0.00_{(\pm 0.00)}$
0.01	$\lambda^+ = 2.39_{(\pm 0.82)}$	MSE	$10.19_{(\pm 0.46)}$	$14.46_{(\pm 0.48)}$	$16.21_{(\pm 0.45)}$
	$\lambda^- = 3.33_{(\pm 0.79)}$	Risk	$0.75_{(\pm 0.09)}$	$0.01_{(\pm 0.001)}$	$0.00_{(\pm 0.00)}$

4.2 Concavity

Concavity and convexity are crucial behaviors in many applications. In this context, we focus on concavity in a single feature. A common example where users might expect concave behavior is in recommendation or preference prediction models. According to economic theory, the utility function with respect to the quantity of an item is often quasi-concave, reflecting the principle of diminishing marginal utility (Mas-Colell et al., 1995). Jenkins et al. (2021) propose a novel loss function to account for this expected concavity, which aligns the model with the concavity property $\mathcal{P}$ during training. Here we again consider aligning a pre-trained model, not trained to satisfy $\mathcal{P}$, using a proximity oblivious tester $\mathcal{T}$ for $\mathcal{P}$ as described in Algorithm 2.

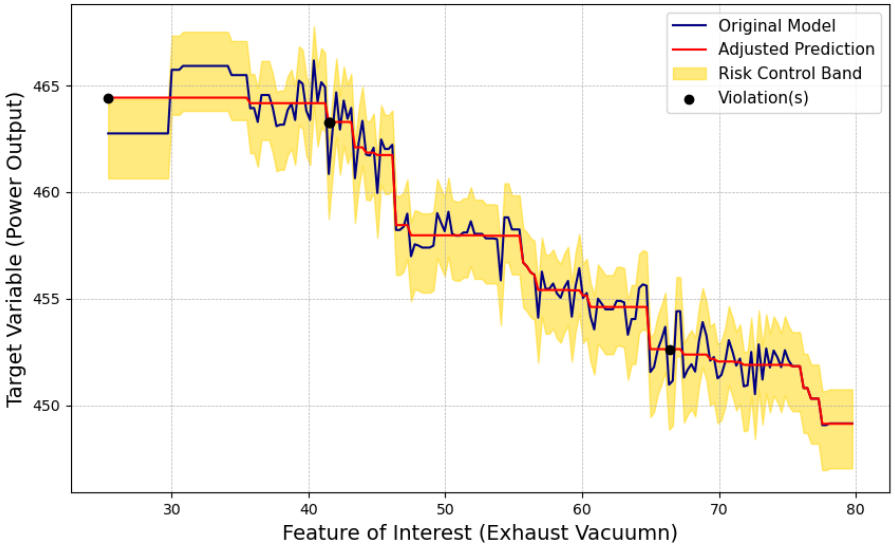

Figure 1: Univariate partial dependence plot of unconstrained model. Risk control band for $\alpha = 0.05$. Dashed line exemplifying Theorem 1 demonstrating existence of monotonically decreasing function falling within the conformal band on $0.975 > 1 - \alpha$ fraction of the domain.

Algorithm 2 POT $\mathcal{T}$ for property $\mathcal{P}$ of concavity in dimension k

Sample $X_{\text{mid}} \sim \mathcal{D}$
Sample $\delta_{\text{left}}, \delta_{\text{right}}$ from empirical differences in feature k
Set X_{left} by decreasing feature k of X_{mid} by δ_{left}
Set X_{right} by increasing feature k of X_{mid} by δ_{right}
Query $F(X_{\text{mid}})$, $F(X_{\text{left}})$, and $F(X_{\text{right}})$
Compute $\alpha = \dfrac{X_{\text{right}}[k] - X_{\text{mid}}[k]}{X_{\text{right}}[k] - X_{\text{left}}[k]}$
if $\min F(X_{\text{mid}}) > \alpha \max F(X_{\text{left}}) + (1-\alpha) \max F(X_{\text{right}})$ **then**
    return Reject
end if
return Accept

We can use a calibration dataset to determine the setting of $\boldsymbol{\lambda} = (\lambda^+, \lambda^-)$ via conformal risk control where the loss function is generated as in Definition 6. The set-valued function will be constructed as $F_{\boldsymbol{\lambda}}(X) = [f(X) - \lambda^-, f(X) + \lambda^+]$. We demonstrate running conformal risk control with this loss function on a real-world dataset in the extended version of the paper Overman et al. (2024)

5 A stylized examination of alignment persistence in AI models

Consider data generated as:

$$y = g(X) + h(X) + \varepsilon\,,$$

where ε is mean-zero noise with variance τ^2 independent of X. Here, $h(X)$ is biased noise we want to ignore, aiming to learn only $g(X)$. Consider the case in which experts expect data to follow $g(X) +$ *unbiased noise*, but biased noise $h(X)$ can obscure this.

One potential reason for the presence of biased noise in data could be due to a measurement error of the outcome that is correlated with select features, leading to an incorrectly calculated outcome. A biased measurement error could occur if there is incomplete data and the presence of the incomplete

data is correlated with select features in an unexpected, systematic way. Our goal is to understand how this bias affects model behavior when trying to learn $g(X)$ alone.

Given n i.i.d. samples $\{(X_i, Y_i)\}_{i=1}^n$ from the above model, we denote this dataset by $\mathcal{D}_n$. We use a random feature model:

$$f_{\mathsf{RF}}(X; \mathbf{a}, \{\mathbf{w}_j\}_{j\in[N]}) = \frac{1}{\sqrt{N}} \sum_{j\in[N]} a_j \sigma(\langle X, \mathbf{w}_j \rangle) \,,$$

where $\mathbf{a} \in \mathbb{R}^N$ are learned weights, and $\{\mathbf{w}_j\}_{j\in[N]}$ are fixed random weights. The squared loss is minimized by ridge regression:

$$\hat{\mathbf{a}}_\lambda = \arg\min_{\mathbf{a}\in\mathbb{R}^N} \sum_{i\in[n]} \left(Y_i - f_{\mathsf{RF}}(X_i)\right)^2 + \lambda \|\mathbf{a}\|_2^2 \,.$$

Users expect a model to exhibit a property $\mathcal{P}$, satisfied by $g(X)$ but not necessarily by $g(X) + h(X)$. We can constrain training to ensure $\mathcal{P}$. Let $C_{\mathcal{P}} = \{\mathbf{a} \mid \mathbf{a} \in \mathbb{R}^N \text{ and } f_{\mathsf{RF}}(X; \mathbf{a}) \text{ satisfies } \mathcal{P}\}$, yielding a constrained model: $\hat{\mathbf{a}}_{\lambda,\mathcal{P}} = \arg\min_{\mathbf{a}\in C_{\mathcal{P}}} \sum_{i\in[n]} \left(Y_i - f_{\mathsf{RF}}(X_i)\right)^2 + \lambda \|\mathbf{a}\|_2^2$.

Assuming g and h are polynomials with $\deg_g < \deg_h$, and given specific conditions on data size and model parameters, we consider two settings: (i) *Classic:* $d^{\deg_g + \delta} < N < d^{\deg_h - \delta}$, and (ii) *Underspecified:* $N > d^{\deg_h + \delta}$ for a small $\delta > 0$.

In the extended version of the paper Overman et al. (2024), we utilize results from Misiakiewicz and Montanari (2023) to derive insights into the impact of model complexity and data size on adherence to $\mathcal{P}$. In particular, we show that under certain assumptions, including small noise bias and robustness of property $\mathcal{P}$, the constrained and unconstrained models have zero distance in the classic setting: $\hat{\mathbf{a}}_{\lambda,\mathcal{P}} = \hat{\mathbf{a}}_\lambda$. However, in the underspecified setting, the constrained and unconstrained models will differ, resulting in a non-zero distance: $\hat{\mathbf{a}}_{\lambda,\mathcal{P}} \neq \hat{\mathbf{a}}_\lambda$. This result implies that in the presence of noise bias, the overparameterized models (i.e., underspecified setting) fail to satisfy the property $\mathcal{P}$, and this cannot be remedied as the data size increases.

6 Related work

Our paper draws from a broad range of areas, hence we refer the reader to textbooks and surveys in alignment (Everitt et al., 2018; Hendrycks et al., 2022; Ji et al., 2024; Hendrycks, 2024), conformal prediction (Angelopoulos and Bates, 2022), property testing (Ron, 2008; Goldreich, 2017), and linearized neural networks (Misiakiewicz and Montanari, 2023).

RLHF (Christiano et al., 2017) has been notably effective in aligning LLMs with human values and intentions, as demonstrated by (Ouyang et al., 2022). Our work considers attempts at alignment that generalzies to models without human-interpretable outputs, which has connections to the scalable oversight problem (Irving et al., 2018; Christiano et al., 2018; Wu et al., 2021). Goal misgeneralization (Langosco et al., 2022; Shah et al., 2022) has potential connections to the underspecified pipeline (D'Amour et al., 2022) considered in this paper in the sense that models with equivalent performance according to the training metric may differ in some other user-desired property during deployment. One of the main methods of assurance (Batarseh et al., 2021), which is concerned with assessing the alignment of pre-trained AI systems, is safety evaluations (Perez et al., 2022; Shevlane et al., 2023) meant to assess risk during deployment, which also has connections to our approach.

The work of Yadkori et al. (2024) closely aligns with ours in both methodology and theme, utilizing conformal risk control to reduce LLM hallucinations (Ji et al., 2023). We discuss connections to this work in the extended version of the paper Overman et al. (2024).

7 Discussion

We introduce a method to align pre-trained models with desired user properties using conformal risk control. By post-processing outputs using property dependent loss functions, we provide probabilistic guarantees that conformal intervals contain functions close to the desired set. This allows for alignment without retraining, effective in both generative and non-generative contexts. Future work should extend these techniques to more properties, explore sample complexity and adaptive querying, and potentially apply them to policy functions in MDP settings for RL agent safety guarantees.

References

Anastasios N. Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2022.

Anastasios Nikolas Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024.

Vipul Arora, Arnab Bhattacharyya, Noah Fleming, Esty Kelman, and Yuichi Yoshida. *Low Degree Testing over the Reals*, pages 738–792. 2023. doi: 10.1137/1.9781611977554.ch31. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611977554.ch31.

Feras A. Batarseh, Laura Freeman, and Chih-Hao Huang. A survey on artificial intelligence assurance. *Journal of Big Data*, 8(1), April 2021. ISSN 2196-1115. doi: 10.1186/s40537-021-00445-7. URL http://dx.doi.org/10.1186/s40537-021-00445-7.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-free, risk-controlling prediction sets, 2021.

Arnab Bhattacharyya, Swastik Kopparty, Grant Robert Schoenebeck, Madhu Sudan, and David Zuckerman. Optimal testing of reed-muller codes. *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 488–497, 2009. URL https://api.semanticscholar.org/CorpusID:25924.

Eric Blais. Testing juntas nearly optimally. pages 151–158, 05 2009. doi: 10.1145/1536414.1536437.

Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1993. ISSN 0022-0000. doi: https://doi.org/10.1016/0022-0000(93)90044-W. URL https://www.sciencedirect.com/science/article/pii/002200009390044W.

Matt Bonakdarpour, Sabyasachi Chatterjee, Rina Foygel Barber, and John Lafferty. Prediction rule reshaping, 2018.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Hugh D. Brunk, Richard E. Barlow, David J. Bartholomew, and Joan M. Bremner. Statistical inference under order restrictions : the theory and application of isotonic regression. *International Statistical Review*, 41:395, 1973. URL https://api.semanticscholar.org/CorpusID:120349543.

José-Ramón Cano, Pedro Antonio Gutiérrez, Bartosz Krawczyk, Michał Woźniak, and Salvador García. Monotonic classification: an overview on algorithms, performance measures and data sets, 2018.

Simon Caton and Christian Haas. Fairness in machine learning: A survey. *ACM Comput. Surv.*, 56 (7), April 2024. ISSN 0360-0300. doi: 10.1145/3616865. URL https://doi.org/10.1145/3616865.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16. ACM, August 2016. doi: 10.1145/2939672.2939785. URL http://dx.doi.org/10.1145/2939672.2939785.

Paul Christiano, Buck Shlegeris, and Dario Amodei. Supervising strong learners by amplifying weak experts, 2018.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(1), jan 2022. ISSN 1532-4435.

Dheeru Dua and Casey Graff. Uci machine learning repository, 2023. URL `http://archive.ics.uci.edu/ml`.

Tom Everitt, Gary Lea, and Marcus Hutter. Agi safety literature review, 2018.

Noah Fleming and Yuichi Yoshida. Distribution-free testing of linear functions on $\mathbb{R}^n$. In Thomas Vidick, editor, *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*, volume 151 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:19, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik. ISBN 978-3-95977-134-4. doi: 10.4230/LIPIcs.ITCS.2020.22. URL `https://drops.dagstuhl.de/opus/volltexte/2020/11707`.

Oded Goldreich. *Introduction to Property Testing*. Cambridge University Press, 2017.

Oded Goldreich and Dana Ron. On proximity-oblivious testing. *Electronic Colloquium on Computational Complexity (ECCC)*, 15, 01 2008. doi: 10.1137/100789646.

Piet Groeneboom and Geurt Jongbloed. *Nonparametric Estimation under Shape Constraints: Estimators, Algorithms and Asymptotics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge, 2014. ISBN 978-0-521-86401-5. doi: 10.1017/CBO9781139020893.

Dan Hendrycks. *AI Safety, Ethics, and Society*. Center for AI Safety, 2024. URL `https://www.aisafetybook.com/`.

Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ml safety, 2022.

Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt. Aligning ai with shared human values, 2023.

Geoffrey Irving, Paul Christiano, and Dario Amodei. Ai safety via debate, 2018.

Porter Jenkins, Ahmad Farag, J. Stockton Jenkins, Huaxiu Yao, Suhang Wang, and Zhenhui Li. Neural utility functions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(9): 7917–7925, May 2021. doi: 10.1609/aaai.v35i9.16966. URL `https://ojs.aaai.org/index.php/AAAI/article/view/16966`.

Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, Fanzhi Zeng, Kwan Yee Ng, Juntao Dai, Xuehai Pan, Aidan O'Gara, Yingshan Lei, Hua Xu, Brian Tse, Jie Fu, Stephen McAleer, Yaodong Yang, Yizhou Wang, Song-Chun Zhu, Yike Guo, and Wen Gao. Ai alignment: A comprehensive survey, 2024.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, March 2023. ISSN 1557-7341. doi: 10.1145/3571730. URL `http://dx.doi.org/10.1145/3571730`.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020.

Tali Kaufman and Dana Ron. Testing polynomials over general fields. *SIAM J. Comput.*, 36:779–802, 10 2006. doi: 10.1137/S0097539704445615.

Lauro Langosco Di Langosco, Jack Koch, Lee D Sharkey, Jacob Pfau, and David Krueger. Goal misgeneralization in deep reinforcement learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 12004–12019. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/langosco22a.html.

Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. Scalable agent alignment via reward modeling: a research direction, 2018.

Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory*. Oxford University Press, New York, 1995.

Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and the double descent curve. 75(4):667–766. ISSN 1097-0312. doi: 10.1002/cpa.22008.

Theodor Misiakiewicz and Andrea Montanari. Six Lectures on Linearized Neural Networks. *arXiv e-prints*, art. arXiv:2308.13431, August 2023. doi: 10.48550/arXiv.2308.13431.

Richard Ngo, Lawrence Chan, and Sören Mindermann. The alignment problem from a deep learning perspective, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

William Overman, Jacqueline Jil Vallon, and Mohsen Bayati. Aligning model properties via conformal risk control, 2024. URL https://arxiv.org/abs/2406.18777.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Ben Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemí Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. Discovering language model behaviors with model-written evaluations, 2022.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://proceedings.neurips.cc/paper_files/paper/2007/file/013a006f03dbc5392effeb8f18fda755-Paper.pdf.

Dana Ron. Property testing: A learning theory perspective. *Found. Trends Mach. Learn.*, 1(3): 307–402, 2008. doi: 10.1561/2200000004. URL https://doi.org/10.1561/2200000004.

Davor Runje and Sharath M. Shankaranarayana. Constrained monotonic neural networks, 2023.

Rohin Shah, Vikrant Varma, Ramana Kumar, Mary Phuong, Victoria Krakovna, Jonathan Uesato, and Zac Kenton. Goal misgeneralization: Why correct specifications aren't enough for correct goals, 2022.

Toby Shevlane, Sebastian Farquhar, Ben Garfinkel, Mary Phuong, Jess Whittlestone, Jade Leung, Daniel Kokotajlo, Nahema Marchal, Markus Anderljung, Noam Kolt, Lewis Ho, Divya Siddarth, Shahar Avin, Will Hawkins, Been Kim, Iason Gabriel, Vijay Bolina, Jack Clark, Yoshua Bengio, Paul Christiano, and Allan Dafoe. Model evaluation for extreme risks, 2023.

Joseph Sill and Yaser Abu-Mostafa. Monotonicity hints. In M.C. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1996. URL https://proceedings.neurips.cc/paper_files/paper/1996/file/c850371fda6892fbfd1c5a5b457e5777-Paper.pdf.

Pnar Tfekci and Heysem Kaya. Combined Cycle Power Plant. UCI Machine Learning Repository, 2014. DOI: https://doi.org/10.24432/C5002N.

Jacqueline J. Vallon, William Overman, Wanqiao Xu, Neil Panjwani, Xi Ling, Sushmita Vij, Hilary P. Bagshaw, John T. Leppert, Sumit Shah, Geoffrey Sonn, Sandy Srinivas, Erqi Pollom, Mark K. Buyyounouski, and Mohsen Bayati. On aligning prediction models with clinical experiential learning: A prostate cancer case study. Unpublished manuscript, 2024.

J.J. Vallon, N. Panjwani, X. Ling, S. Vij, E. Pollom, H.P. Bagshaw, S. Srinivas, J. Leppert, M. Bayati, and M.K. Buyyounouski. Clinically consistent prostate cancer outcome prediction models with machine learning. *International Journal of Radiation Oncology*Biology*Physics*, 114(3, Supplement):e126–e127, 2022. ISSN 0360-3016. doi: https://doi.org/10.1016/j.ijrobp.2022.07.951. URL https://www.sciencedirect.com/science/article/pii/S0360301622016728. ASTRO Annual 2022 Meeting.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World, Second Edition*. January 2005. doi: 10.1007/978-3-031-06649-8. Springer-Verlag New York, Inc. 2005.

Serena Wang and Maya Gupta. Deontological ethics by monotonicity shape constraints, 2020.

Jeff Wu, Long Ouyang, Daniel M. Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback, 2021.

Yasin Abbasi Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, Ali Taylan Cemgil, and Nenad Tomasev. Mitigating llm hallucinations via conformal abstention, 2024.

Seungil You, David Ding, Kevin Canini, Jan Pfeifer, and Maya Gupta. Deep lattice networks and partial monotonic functions, 2017.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the contributions of proposing a model alignment approach via conformal risk control and property testing, with applications demonstrated on real-world datasets. The claims are supported by theoretical results and experimental validation. See Section 3 and 4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper mentions the assumptions made for the theoretical results, such as the nature of the data distribution and the properties of the models. The discussion section also reflects on the limitations regarding the types of properties and models to which the methodology can be applied. See Section 3.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The theoretical results are supported by complete proofs provided in both the main text and the extended version of the paper. Assumptions are clearly stated along with theorems. See Section 3 and the extended version of the paper.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The experimental setup, including data splits, model parameters, and the procedure for applying conformal risk control, is described in detail. This information is sufficient to reproduce the main experimental results. See Section 4 and the extended version of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper provides links to the UCI ML repository for the datasets used, and the code used for experiments will be made available upon publication. Sufficient instructions are included for reproducing the experiments. See Section 4 and the extended version of the paper.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper includes detailed information about data splits, model training, hyperparameters, and the selection process, allowing readers to understand and reproduce the results. See Section 4 and the extended version of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: The paper reports mean squared errors and risks, and the experiments are averaged over multiple random splits, providing sufficient information on variability and significance suitable for our context. See Section 4 and the extended version of the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper specifies the use of XGBoost models and mentions typical execution times and resources used for the experiments. See the extended version of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research adheres to ethical guidelines by ensuring transparency, reproducibility, and careful consideration of societal impacts, especially in the context of model alignment and fairness.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The discussion includes potential positive impacts of improved model alignment on fairness and transparency, as well as the challenges and limitations that need to be addressed. See Section 1.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not involve high-risk data or models.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper uses publicly available datasets from the UCI ML repository, which are properly credited and cited. See Section 4 and the extended version of the paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not introduce new datasets or models.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.