QUANTILE REGULARIZATION: TOWARDS IMPLICIT CALIBRATION OF REGRESSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep learning models are often poorly calibrated, i.e., they may produce overconfident predictions that are wrong, implying that their uncertainty estimates are unreliable. While a number of approaches have been proposed recently to calibrate classification models, relatively little work exists on calibrating regression models. Isotonic Regression has recently been advocated for regression calibration. We provide a detailed formal analysis of the *side-effects* of Isotonic Regression when used for regression calibration. To address these, we investigate the idea of quantile calibration (Kuleshov et al., 2018), recast it as entropy estimation, and leverage the new formulation to construct a novel quantile regularizer, which can be used as a blackbox to calibrate any probabilistic regression model. Unlike most of the existing approaches for calibrating regression models, which are based on *post-hoc* processing of the model's output, and require an additional dataset, our method is trainable in an end-to-end fashion, without requiring an additional dataset. We provide empirical results demonstrating that our approach improves calibration for regression models trained on diverse architectures that provide uncertainty estimates, such as Dropout VI, Deep Ensembles.

1 INTRODUCTION

For supervised machine learning, the notion of calibration of a learned predictive model is a measure of evaluating how well a model's confidence in its prediction matches with the correctness of these predictions. For example, a binary classifier will be considered perfectly calibrated if, among all predictions with probability score 0.9, 90% of the predictions are correct Guo et al. (2017). Likewise, consider a probabilistic regression model that produces credible interval for the predicted outputs. In this setting, the model will be considered perfectly calibrated if the 90% confidence interval contains 90% of the true test outputs (Kuleshov et al., 2018). Unfortunately, modern deep neural networks are known to be poorly calibrated (Guo et al., 2017), raising questions on their reliability.

The notion of calibration for classification problems was originally first considered in meteorology literature (Brier, 1950; Murphy, 1972; Gneiting & Raftery, 2007) and saw one of its first prominent usage used in the machine learning literature by (Platt et al., 1999) in context of Support Vector Machines (SVM), in order to obtain probabilistic predictions from SVMs which are inherently non-probabilistic models. Recently, there has been renewed interested in calibration, especially for classification models, after it has shown (Guo et al., 2017) that modern deep neural networks for classification are often poorly calibrated.

Currently there are three main notions of calibration in case of classification (Kumar et al., 2019; Vaicenavicius et al., 2019; Kull et al., 2019). Most calibration methods (Platt et al., 1999; Zadrozny & Elkan, 2001; 2002; Guo et al., 2017; Kull et al., 2017; 2019) for classification models are *post hoc*, where they learn a calibration mapping $\mathsf{R} : [0, 1] \rightarrow [0, 1]$ using an additional dataset to *recalibrate* an already trained model. There has been recent work showing some of these popular post hoc methods are either themselves miscalibrated or sample-inefficient (Kumar et al., 2019) and they do not actually help the model output well-calibrated probabilities.

An alternative to post hoc processing is to ensure that model outputs well-calibrated probabilities after model training finishes. We refer to these as *implicit* calibration methods. Notably, such an approach does not require an additional dataset to learn the calibration mapping. While almost all post hoc calibration methods for classification models can be seen in a unified manner as density

estimation methods(see section 2.1), existing implicit calibration methods for classification models have been designed with various, often distinct, considerations/approaches. Several heuristics like Mixup (Zhang et al., 2017; Thulasidasan et al., 2019) and Label Smoothing (Szegedy et al., 2016; Müller et al., 2019) that were part of high performance deep networks for classification were later shown empirically to achieve calibration. (Maddox et al., 2019) show that their optimization method instrinsically improves calibration. (Pereyra et al., 2017) found that penalizing high-confidence predictions acts as a regularizer. A more principled way of achieving implicit calibration is by minimizing a loss function that is tailored for calibration (Kumar et al., 2018). This is somewhat similar in spirit to our proposed approach that aims to do it for regression models.

Among the early approaches for calibrating regression models, (Gneiting et al., 2007) were the first to propose a framework for calibrating regression models. However, they do not provide any procedure to correct a miscalibrated model. Recently, (Kuleshov et al., 2018) introduced the notion of *Quantile Calibration* which intuitively says that the p confidence interval predicted by model should have target variable with probability p. They use a *post hoc* calibration method based on Isotonic Regression (Fawcett & Niculescu-Mizil, 2007), which is a well-known calibration technique for classification models. The difference between Isotonic Calibration in classification and Isotonic Calibration in regression is that (i) dataset on which calibration mapping is learnt; and (ii) the function with which learnt calibration mapping is pre-composed . In the former case, it is pre-composed with a probability mass function (PMF) and whereas in the latter, it is pre-composed with a conditional density function (CDF). Both these differences produces side effects i) nature of recalibration dataset already satisfies monotonicity constraint , so there is overfitting in case of smaller calibration datsets ii) since we are composing CDF with piecewise linear function, we will make the resultant CDF discontinuous and corresponding PDF non differentiable (see Sec. 3 for detailed discussion for side effects of the Isotonic Calibration approach). In another recent work, (Song et al., 2019) proposed a much stronger notion of calibration called *Distributional Calibration* which guarantees that among all instances whose predicted probability density function (PDF) of the response variable has mean μ and standard deviation σ, the *marginal* distribution of the target variable should have mean μ and standard deviation σ. They too propose a *post hoc* recalibration method based on expensive Gaussian processes. Among other work, (Keren et al., 2018), consider a different setting where neural networks for classification are used for regression problems and showed that temperature scaling (Hinton et al., 2015; Guo et al., 2017) and their proposed method based on empirical prediction intervals improves calibration for regression problems as well. Again, these are *post hoc* methods.

Our contributions are summarized below:

1. We analyze in detail the side effects of Isotonic Calibration for regression models. We show how using Isotonic Calibration results in truncation of the support, which will result in assigning zero likelihood at test time. We also discuss about Isotonic Calibration resulting in nonsmooth PDFs, and its tendency to produce miscalibration when using small calibration datasets.

2. At test time, after composing predicted CDF with the learned isotonic mapping, the mean prediction (point estimate) also changes. Kuleshov et al. (2018) do not acknowledge changes in the mean estimate. While Song et al. (2019) acknowledge this issue, they use a trapezoidal approximation to remedy this. In contrast, we derive an analytical expression for the updated point estimate after isotonic calibration. We also provide a different expression for updated point estimate, which reduces the time-complexity from $\mathcal{O}(m)$ to $\mathcal{O}(1)$, where m is calibration dataset size.

3. In order to mitigate these shortcomings of Isotonic Calibration, we propose a simple, yet novel and general purpose, trainable loss function for quantile calibration where the smoothness of PDF/CDF is not sacrificed for well-calibrated probabilties. Our approach also eliminates the need for additional calibration dataset.

4. We conduct extensive experiments on a wide range of architectures using the proposed loss function (Quantile Regularization) and show empirically that it improves calibration on wide range of architectures that produce uncertainty estimates.

2 BACKGROUND AND DEFINITIONS

Probabilistic regression model can be seen as conditional PDF/conditional CDF. In the rest of the paper, we express it as conditional CDF $\mathsf{M} : \mathcal{X} \rightarrow (\mathcal{Y} \rightarrow [0,1])$. In practice, model outputs μ and σ or parameters of likelihood distribution under consideration. In the rest of paper we consider Gaussian likelihood unless otherwise stated because it is the most prevalent case. Kuleshov et al. (2018) proposed following notion of calibration of regression models called *quantile calibration*. Importance of following definition is we get reliable confidence intervals.

Definition 1 (Quantile Calibration) *Given a regression model* $\mathsf{M} : \mathcal{X} \rightarrow (\mathcal{Y} \rightarrow [0,1])$ *and* X, Y *jointly distributed as* $\mathbf{P}$*, the model* M *is said to be Quantile Calibrated* iff

$$\mathbb{P}\Big[\, [\, \mathsf{M}(X)\,](Y) \leq p \,\Big] = p \;\; \forall p \in [0,1] \tag{1}$$

Note that $[\, \mathsf{M}(X)\,](Y)$ is cumulative density that the model predicts for random input-response pair drawn from the joint distribution of X and Y. Pick a value between $[0,1]$, say 0.68, if the model is Quantile Calibrated, 68% confidence interval contains target points 68% on average.

2.1 FUNDAMENTAL THEOREM OF POST HOC CALIBRATION

The objective of post hoc calibration is to *recalibrate* a *miscalibrated* model M by learning a mapping $\mathsf{R} : [0,1] \rightarrow [0,1]$ s.t. $\mathsf{R} \circ \mathsf{M}$ is calibrated model. One such mapping can be obtained from the definition of calibration itself. Setting $\mathsf{R}\,(p) = \mathbb{P}\big[[\mathsf{M}(X)](Y) \leq p\big]$ makes $\mathsf{R} \circ \mathsf{M}$ a quantile calibrated model. (Vaicenavicius et al., 2019) call an analogous mapping in the context of classification as *canonical calibration* mapping.

Theorem 1 *For any Model* $\mathsf{M} : \mathcal{X} \rightarrow (\mathcal{Y} \rightarrow [0,1])$ *and given canonical calibration mapping* $\mathsf{R}(p) = \mathbb{P}\big[[\mathsf{M}(X)](Y) \leq p\big]$*,* $\mathsf{R} \circ \mathsf{M}$ *is quantile calibrated*

Note that learning the mapping R reduces to density estimation of $\mathbb{P}\big[[\mathsf{M}(X)](Y) \leq p\big]$, which is a hard problem in itself. With this insight, and using the fact that the mapping is monotonically increasing, (Kuleshov et al., 2018) use Isotonic Regression to learn the mapping on a separate calibration dataset. Similar theorems hold for other notions of calibration, too.

2.2 ISOTONIC CALIBRATION

We next describe Isotonic Calibration which is based on Isotonic Regression. Given ordered inputs $x_1, x_2, \ldots x_m$ i.e., $x_1 \leq x_2 \cdots \leq x_m$ and corresponding target variables $y_1, y_2, \ldots y_m$, Isotonic Regression finds $z_1, z_2, \ldots z_m$ by solving the following optimization problem

$$\begin{aligned} \min_{\mathbf{z}} \quad & \frac{1}{m}\sum_{i=1}^{m}(z_i - y_i)^2 \\ \text{s.t.} \quad & z_1 \leq z_2 \cdots \leq z_m \end{aligned} \tag{2}$$

(Mair et al., 2009) provides a nice survey of algorithms for Isotonic Regression. In particular Scikit-learn (Pedregosa et al., 2011) uses Pool Adjacent Violaters Algorithm (PAVA) of complexity $\mathcal{O}(m)$, . Given a calibration dataset $\{\mathbf{x}_i, y_i\}_{i=1}^{m}$ Isotonic Calibration first builds a recalibration set as $\mathcal{D} = \left\{ \Big(\; M(\mathbf{x}_i)[y_i] \;\; , \;\; \frac{1}{m}\sum_{j=1}^{m} \mathbb{I}\Big[\; M(\mathbf{x}_j)[y_j] \leq M(\mathbf{x}_i)[y_i] \; \Big] \Big)_{i=1}^{m} \right\}$, and then applies Isotonic Regression on the dataset $\mathcal{D}$ to get isotonic mapping with linear interpolation.

3 ANALYSIS OF ISOTONIC CALIBRATION

Let $\{(\mathbf{x}_k, y_k)\}_{k=1}^{N}$ be a calibration dataset. Let $\{(\mu_k, \sigma_k^2)\}_{k=1}^{N}$ be means and variances predicted by model trained. Suppose we have obtained CDF values at actual outputs

$(y_1, y_2, \cdots, y_n)$. Let these be $(c_1, c_2, \cdots, c_n)$. Then the calibration dataset is constructed as $\mathcal{D} = \{(c_{(1)}, \frac{1}{n}), (c_{(2)}, \frac{2}{n}), \cdots, (c_{(n)}, 1)\}$ where $\mathcal{C} = (c_{(1)}, c_{(2)}, \cdots c_{(n)})$ is obtained by sorting $(c_1, c_2, \cdots c_n)$ in ascending order. Now we have to fit isotonic regression on this dataset. Since $(\frac{1}{n}, \frac{2}{n}, \cdots, 1)$ are already in increasing order, isotonic regression does not modify values. So we don't even need to fit per say except for basic interpolation. Complete Mapping is obtained by adding $(0, 0)$ to $\mathcal{D}$ and connecting these points. We first give conditions under which smoothness is lost and later derive the correction one has to use after isotonic calibration. Note that all proofs are present in supplementary material.

Claim 1 *Let Y be a random variable with CDF F, and let $G = \mathsf{R} \circ F$ be its CDF after composing with mapping R obtained from isotonic regression characterized by $\mathcal{C} = \{c_{(1)}, c_{(2)}, \cdots, c_{(n)}\}$. If there exist $i-1, i, i+1 \in \{0, n\} \wedge c_{(i)} - c_{(i-1)} \neq c_{(i+1)} - c_{(i)}$ then the CDF G is not differentiable and its corresponding probability density function g is not continuous at $F^{-1}(c_{(i)})$*

Because of this, PDF of transformed r.v becomes discontinuous and spiky (see Fig. 1d). Update likelihood inversely depends on $n(c_{(i)} - c_{(i+1)})$. In many of cases it becomes so small that somtimes updated likelihood increases by factor of 10^2 to 10^5 for single point there by completely destroying what average log likelihood represents . To show this, we report maximum likelihood attained for single test time point after isotonic calibration in UCI experiments (see Table 1, Table 2)

Now we derive analytical expression for updated mean after Isotonic Calibration.

Claim 2 *Let Y_{iso} be transformed random variable after applying isotonic mapping R on random variable Y. Then the expectation of Y_{iso} is as follows.*

$$\mathbb{E}[Y_{iso}] = \mu - \frac{\sigma^2}{n} \sum_{i=0}^{n-1} \frac{f(F^{-1}(c_{(i+1)})) - f(F^{-1}(c_{(i)}))}{(c_{(i+1)} - c_{(i)})} \tag{3}$$

Summation involves both recalibration dataset and test time prediction. Now we will use properties of quantile functions to decouple the dependency , by which summation just depends on recalibration dataset.

Claim 3 *let $c_{(i)} = F_{\mu_{(i)}, \sigma_{(i)}}(y_{(i)})$ and $p_{(i)} = f_{\mu_{(i)}, \sigma_{(i)}}(y_{(i)})$*

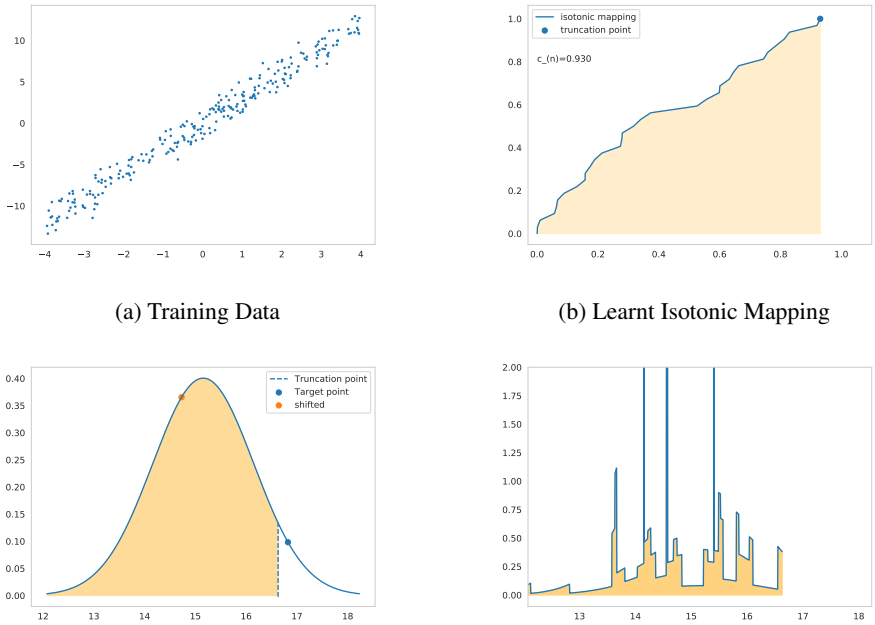
$$\mathbb{E}[Y_{iso}] = \mu - \sigma \underbrace{\sum_{i=0}^{n} \frac{1}{n} \frac{\sigma_{(i+1)} p_{(i+1)} - \sigma_{(i)} p_{(i)}}{c_{(i+1)} - c_{(i)}}}_{\delta} \tag{4}$$

Now δ can be computed *once*, so at test time given μ, σ as model prediction, updated mean after isotonic calibration is $\mu - \delta.\sigma$. This way the time required is reduced from $\mathcal{O}(m)$ to $\mathcal{O}(1)$. Also, the construction of recalibration dataset as suggested in (Kuleshov et al., 2018) can result in truncation of support of updated r.v (see fig. 1d). One way to remedy is to use $(\mathbf{x}, \infty)$ to calibration dataset for any $\mathbf{x} \in \mathbb{R}^n$. So recalibration dataset becomes $\{(c_{(1)}, \frac{1}{n+1}), \ldots, (c_{(n)}, \frac{n}{n+1}), (1, 1)\}$.

To illustrate all these , we use Bayesian linear regression with 256 training examples generated from $y = 3x + \epsilon$ where $\epsilon \sim \mathcal{N}(0, 1)$. and $x \in [-4, 4]$ randomly sampled and calibration dataset of size 32.

4 Quantile Regularization

In quantile calibration, we want $\mathbb{P}\big[[\mathsf{M}(X)](Y) \leq p\big] = p \; \forall p \in [0, 1]$. Our method is based on the idea that both right and left hand sides can be viewed as CDF. Let $R(p) = \mathbb{P}\big[[\mathsf{M}(X)](Y) \leq p\big]$ and $S(p) = p$. Here R can be seen as the CDF of $[\mathsf{M}(X)](Y)$ while S can be seen as the CDF of Uniform[0,1]. Quantile calibration mandates these two CDFs to be equal. So, for a perfectly calibrated quantile model M, we have that $[\mathsf{M}(X)](Y)$ is a *uniform* distribution. We seek to penalize the model this r.v deviates from a uniform distribution which gives us a calibration metric that can be used as a regularizer with any regression loss function, achieving highly desirable calibration during



(a) Training Data

(b) Learnt Isotonic Mapping

(c) Predicted PDF

(d) PDF after Isotonic Calibration

Figure 1: (A) shows 256 training data points. (B) shows the calibration mapping learnt using Isotonic Calibration with 32 samples. (C) shows predicted PDF at test time before applying Isotonic Calibration and truncation and shift in mean that happens if the mapping in (B) is applied . It also how such truncation can affect the support. (D) shows resultant PDF after Isotonic Calibration

training itself. We use the KL divergence as distance measure. Note that the KL divergence between any distribution on $[0, 1]$ and the uniform distribution is negative of the differential entropy, which provides a very intuitive interpretation for the Quantile Regularization (QR) based loss. Essentially, to improve calibration while training, QR maximizes the differential entropy of $[M(X)](Y)$ i.e., the predicted cumulative density of target value. We formalize the statement for completeness below.

Claim 4 *Let* M *be any regression model. Then* M *is perfectly quantile calibrated* iff

$$\mathrm{KL}\Big([\mathsf{M}(X)](Y) || \mathrm{U} \Big) = 0 \tag{5}$$

4.1 Differential Entropy Estimation

We need the differential entropy estimator for deriving our calibration loss function. There is a rich literature for entropy estimation. A brief overview about non-parametric entropy estimation can be found at (Beirlant & Dudewicz, 1997). We use sample-spacing entropy estimation originally proposed in (Vasicek, 1976). An m spacing of random variable is defined as amount of probability mass between ordered samples that are $m - 1$ samples apart. Let S be a one-dimension r.v. with CDF F and assume we are given n samples $\{s_i\}_{i=1}^n \sim X$ and m s.t. $1 \leq m \leq n$. Sample spacing entropy estimation is based idea on following observation of m-spacings of random variable

$$\mathbb{E}\Big[F(s_{(i+m)}) - F(s_{(i)})\Big] = \frac{n+1}{m}$$

There are many formulations of sample spacing entropy estimators but we use the one in (Learned-Miller et al., 2003). Let S be one dimension r.v and assume we are given n samples $\{s_i\}_{i=1}^n \sim X$ and m s.t. $1 \leq m \leq n$, then the sample spacing entropy estimator is given as follows

$$\hat{\mathrm{H}}(S) = \frac{1}{n-m} \sum_{i=1}^{n-m} \log \left[\frac{n+1}{m} (s_{(i+m)} - s_{(i)}) \right] \tag{6}$$

4.2 CALIBRATION LOSS FUNCTION

In our case, the random variable is $[\mathsf{M}(X)](Y)$. Given training data (x_k, y_k) we need to get samples $[\mathsf{M}(x_i)](y_i)$, and compute above expression $\hat{\mathrm{H}}(S)$ and maximize it. Note that,we want to make this part of training loop to achieve implicit calibration. To do so, we need *ordered* samples to compute the Entropy in Eq. 6. However, inherently, sorting is not a differentiable operation. We use NeuralSort (Grover et al., 2019) as a differentiable relaxation to sorting. Justification for the sample spacing entropy estimator despite its need for differentiable sorting is that the r.v. under consideration has support $[0, 1]$ and one has to use beta kernel (Chen, 1999) whose entropy doesn't have a simple form while the spacing estimator does. We summarize our Quantile Regularization algorithm below

Algorithm 1 Quantile Regularization

Precondition: $(\mathbf{x}_k, y_k)$ are n i.i.d training instances and $\mu_k, \sigma_k = \text{MODEL}_{\mathbf{w}}(\mathbf{x}_k)$ and DIFFSORT is any differentiable relaxation to sorting operation.

function CALIBRATION LOSS FUNCTION($\mathbf{y}, \boldsymbol{\mu}, \boldsymbol{\sigma}$)

  for $k \leftarrow 1$ to n **do**
    $\Phi_k \leftarrow \Phi(\mu_k, \sigma_k)$
    $c_k \leftarrow \Phi_k[y_k]$
  end for
  $\mathbf{s} \leftarrow \text{DIFFSORT}(\mathbf{c})$
  $m \leftarrow \sqrt{n}$
  $e \leftarrow \frac{1}{n-m} \sum_{i=1}^{n-m} \log \left[\frac{n+1}{m} (s_{i+m} - s_i) \right]$
  return e

end function

Assume that $\mathbf{X}, \mathbf{y}$ is the training data. Let $(\boldsymbol{\mu}_{\mathbf{w}}, \boldsymbol{\sigma}_{\mathbf{w}}) = \text{MODEL}_{\mathbf{w}}(\mathbf{X})$, where $\mathbf{w}$ denotes the parameters of the model, $\mathcal{NLL}(\mathbf{y}, \boldsymbol{\mu}_{\mathbf{w}}, \boldsymbol{\sigma}_{\mathbf{w}})$ be the negative log likelihood and $\mathcal{CL}(\mathbf{y}, \boldsymbol{\mu}_{\mathbf{w}}, \boldsymbol{\sigma}_{\mathbf{w}})$ be the quantile regularization based calibrated loss computed by Algorithm 1. The overall loss function can be written as , where the hyperparameter λ controls calibration.

$$\mathcal{L}(\mathbf{X}, \mathbf{y}, \boldsymbol{\mu}_{\mathbf{W}}, \boldsymbol{\sigma}_{\mathbf{W}}) = \mathcal{NLL}(\mathbf{y}, \boldsymbol{\mu}_{\mathbf{w}}, \boldsymbol{\sigma}_{\mathbf{w}}) + \lambda \times \mathcal{CL}(\mathbf{y}, \boldsymbol{\mu}_{\mathbf{w}}, \boldsymbol{\sigma}_{\mathbf{w}}) \tag{7}$$

4.3 DEGENERATE BUT PERFECTLY QUANTILE CALIBRATED MODEL

Every notion of calibration has examples of models such that it is perfectly calibrated according to that specific notion but far from the ideal model. We prove below that Quantile Calibration is no different. Specifically, we give conditions under which is model is quantile calibrated and then use it to build a model that is degenerate in the sense that the model does not depend on the inputs while predicting output, but is a perfectly quantile calibrated model.

Claim 5 *Let f be the marginal distribution of Y and $F_x = M[x]$ be the model's predicted cumulative distribution for $x \in \mathcal{X}$ and f_x be corresponding predicted probability density then if following holds, M is Quantile Calibrated*

$$f[F_x^{-1}(p)] = f_x(F_x^{-1}(p)) \;\; \forall x, \forall p$$

Now if we set $f = f_x \; \forall x$ above condition easily holds. So, a model that outputs marginal distribution of Y for every input $x \in \mathcal{X}$, is perfectly quantile calibrated. *Note that this observation is general*

Dataset	Heteroscedastic Dropout VI ($\lambda = 1$)							
	Calibration Error(%)		NLL		RMSE		time	
	base	QR	base	QR	base	QR	base	QR
Air Foil	14.69 $\pm$ 2.55	**7.04 $\pm$ 0.82**	**2.69 $\pm$ 0.01**	2.77 $\pm$ 0.03	3.66 $\pm$ 0.09	3.95 $\pm$ 0.10	1.56 $\pm$ 0.03	2.20 $\pm$ 0.06
Boston Housing	23.90 $\pm$ 6.35	**23.41 $\pm$ 3.10**	**3.24 $\pm$ 0.09**	3.28 $\pm$ 0.07	**4.54 $\pm$ 0.27**	4.61 $\pm$ 0.16	0.57 $\pm$ 0.01	0.73 $\pm$ 0.01
Concrete Strength	28.74 $\pm$ 4.70	**18.59 $\pm$ 2.22**	**3.62 $\pm$ 0.04**	**3.62 $\pm$ 0.03**	**8.74 $\pm$ 0.20**	8.80 $\pm$ 0.12	1.01 $\pm$ 0.01	1.43 $\pm$ 0.03
Fish Toxicity	2.64 $\pm$ 0.43	**2.00 $\pm$ 0.38**	**1.24 $\pm$ 0.00**	**1.24 $\pm$ 0.00**	**0.93 $\pm$ 0.00**	**0.93 $\pm$ 0.00**	1.09 $\pm$ 0.02	1.45 $\pm$ 0.03
Kin8nm	7.58 $\pm$ 0.55	**4.34 $\pm$ 0.45**	**-0.87 $\pm$ 0.01**	**-0.87 $\pm$ 0.01**	**0.09 $\pm$ 0.00**	0.10 $\pm$ 0.00	7.17 $\pm$ 0.04	10.10 $\pm$ 0.06
Protein Structure	2.65 $\pm$ 0.26	**1.07 $\pm$ 0.49**	**2.88 $\pm$ 0.0**	**2.88 $\pm$ 0.0**	4.64 $\pm$ 0.01	4.67 $\pm$ 0.01	41.44 $\pm$ 0.03	56.64 $\pm$ 0.03
Red Wine	3.01 $\pm$ 0.50	**2.09 $\pm$ 0.73**	**0.98 $\pm$ 0.00**	0.99 $\pm$ 0.00	**0.65 $\pm$ 0.00**	**0.65 $\pm$ 0.00**	1.57 $\pm$ 0.01	2.10 $\pm$ 0.03
White Wine	3.49 $\pm$ 0.62	**3.30 $\pm$ 0.14**	**1.10 $\pm$ 0.01**	1.12 $\pm$ 0.01	**0.73 $\pm$ 0.00**	**0.73 $\pm$ 0.00**	4.47 $\pm$ 0.00	6.12 $\pm$ 0.00
Yacht Hydrodynamics	33.03 $\pm$ 0.75	**22.96 $\pm$ 2.99**	**2.27 $\pm$ 0.03**	2.46 $\pm$ 0.03	**4.18 $\pm$ 0.38**	4.53 $\pm$ 0.39	1.01 $\pm$ 0.00	2.41 $\pm$ 0.02
Year Prediction MSD	5.12 $\pm$ NA	**1.98 $\pm$ NA**	**3.45 $\pm$ NA**	**3.45 $\pm$ NA**	9.12 $\pm$ NA	9.28 $\pm$ NA	547.41 $\pm$ NA	733.48 $\pm$ NA

Dataset	Heteroscedastic Dropout VI							
	Calibration Error(%)			RMSE			max likelihood	
	base+iso	QR	QR+iso	base+iso	QR	QR+iso	base+iso	QR+iso
Air Foil	17.02 $\pm$ 2.39	**7.04 $\pm$ 0.82**	9.63 $\pm$ 2.29	**3.58 $\pm$ 0.06**	3.95 $\pm$ 0.10	3.92 $\pm$ 0.09	5040.50	1406.11
Boston Housing	34.84 $\pm$ 6.44	**23.41 $\pm$ 3.10**	31.41 $\pm$ 6.4	4.55 $\pm$ 0.24	4.61 $\pm$ 0.16	4.55 $\pm$ 0.16	12252.25	2404.26
Concrete Strength	37.50 $\pm$ 2.52	**18.59 $\pm$ 2.22**	22.13 $\pm$ 3.47	**8.74 $\pm$ 0.18**	8.80 $\pm$ 0.12	8.73 $\pm$ 0.14	8133.59	1014.12
Fish Toxicity	1.56 $\pm$ 0.18	2.00 $\pm$ 0.38	**1.50 $\pm$ 0.11**	**0.93 $\pm$ 0.00**	**0.93 $\pm$ 0.00**	**0.93 $\pm$ 0.00**	20453.87	621.84
Kin8nm	**4.34 $\pm$ 0.45**	**-0.87 $\pm$ 0.01**	**-0.87 $\pm$ 0.01**	**0.09 $\pm$ 0.00**	0.10 $\pm$ 0.00	0.10 $\pm$ 0.00	7492.71	11211.06
Protein Structure	**0.05 $\pm$ 0.00**	1.07 $\pm$ 0.49	**0.05 $\pm$ 0.00**	4.64 $\pm$ 0.01	4.67 $\pm$ 0.01	**4.63 $\pm$ 0.01**	1744.95	1619.98
Red Wine	2.93 $\pm$ 0.05	**2.09 $\pm$ 0.73**	**2.09 $\pm$ 0.10**	**0.65 $\pm$ 0.00**	**0.65 $\pm$ 0.00**	**0.65 $\pm$ 0.00**	11736.34	1441.88
White Wine	4.44 $\pm$ 0.30	**3.30 $\pm$ 0.14**	4.09 $\pm$ 0.17	**0.73 $\pm$ 0.00**	**0.73 $\pm$ 0.00**	**0.73 $\pm$ 0.00**	4357.62	4858.45
Yacht Hydrodynamics	16.26 $\pm$ 2.94	22.96 $\pm$ 2.99	**15.56 $\pm$ 3.10**	4.20 $\pm$ 0.26	4.53 $\pm$ 0.39	4.69 $\pm$ 0.41	5881.62	170320.73
Year Prediction MSD	**0.02 $\pm$ NA**	1.98 $\pm$ NA	**0.02 $\pm$ NA**	**9.09 $\pm$ NA**	9.28 $\pm$ NA	9.27 $\pm$ NA	105.72	174.52

Table 1: Base Model is Dropout-VI model and QR is when Base Model is trained with Quantile Regularization. (Base/QR) +iso represents after applying Isotonic Calibration. We use $\lambda = 1$

that it does not require normal likelihood, for it to be true. As a simple example, consider $f(x, y) = \mathcal{N}(y|5x, 1)\mathcal{N}(x|0, 4)$. then the marginal distribution of y is $f(y) = \int_{-\infty}^{\infty} \mathcal{N}(y|5x, 1)\mathcal{N}(x|0, 4)dx = \mathcal{N}(y|5.0, 5.4^2.5 + 1^2) = \mathcal{N}(y|0, 401)$ so the model which predicts $\mathcal{N}(y|0, 401)$ regardless of input is perfectly quantile calibrated, but the true model is $\mathcal{N}(y|5x, 1)$. So, a model can predict good confidence intervals despite being far from ideal. Hence we need model to be both well-calibrated and sharp.

5 Experiments

We evaluate our approach on various regression datasets in terms of the calibration error as well as other standard metrics, such as root-mean-squared-error (RMSE) and negative log-likelihood (NLL).

5.1 l_2 Quantile Calibration Error

Given any model $\mathrm{M} : \mathcal{X} \to (\mathcal{Y} \to [0, 1])$, we define the l_2 calibration error as follows. Let us choose m equidistant points $\{p_m\}_{m=1}^{M}$ in $(0, 1]$ with $p_M = 1$. Given a test set $\{x_n, y_n\}_{n=1}^{N}$, whose predictions are $F_n = F(x_n)$, the M-bin estimator of above integral will give us metric used in (Kuleshov et al., 2018).

$$\mathcal{CE}(F) = \int_0^1 \left(\mathbb{P}\big[[\mathrm{M}(X)](Y) \leq p\big] - p\right)^2 dp \approx \frac{1}{M} \sum_{i=1}^{M} \Big[\sum_{j=1}^{N} \frac{1}{N} \mathrm{I}[F_j(y_j) \leq p_i] - p_i \Big]^2 \tag{8}$$

5.2 UCI Data Experiments

we consider two different architectures Dropout VI (Gal, 2016) and Deep Ensembles(Lakshminarayanan et al., 2017). The dataset sizes ranges from 308 to 515345 and input feature dimensions ranges from 6 to 91. Every dataset, except Year Prediction MSD, is divided into 5 splits

Dataset	Deep Ensembles with Adversarial Training ($\lambda = 4$)							
	Calibration Error(%)		NLL		RMSE		time	
	base	QR	base	QR	base	QR	base	QR
Air Foil	23.01 ± 3.53	$\mathbf{19.04 \pm 1.44}$	2.84 ± 0.09	$\mathbf{2.82 \pm 0.05}$	$\mathbf{3.16 \pm 0.05}$	3.43 ± 0.03	8.75 ± 0.03	10.95 ± 0.04
Boston Housing	30.92 ± 4.73	$\mathbf{31.22 \pm 4.09}$	4.56 ± 0.17	$\mathbf{4.41 \pm 0.46}$	4.70 ± 0.12	$\mathbf{4.46 \pm 0.09}$	2.99 ± 0.02	3.71 ± 0.01
Concrete Strength	39.95 ± 1.06	$\mathbf{30.85 \pm 2.40}$	4.56 ± 0.10	$\mathbf{4.32 \pm 0.06}$	8.86 ± 0.18	$\mathbf{8.70 \pm 0.01}$	6.01 ± 0.02	7.39 ± 0.05
Fish Toxicity	3.48 ± 0.62	$\mathbf{2.62 \pm 0.24}$	$\mathbf{1.44 \pm 0.03}$	1.47 ± 0.03	0.92 ± 0.01	$\mathbf{0.91 \pm 0.00}$	5.67 ± 0.04	7.15 ± 0.02
Kin8nm	$\mathbf{0.87 \pm 0.28}$	1.04 ± 0.16	$\mathbf{-1.35 \pm 0.00}$	$\mathbf{-1.35 \pm 0.01}$	$\mathbf{0.07 \pm 0.00}$	$\mathbf{0.07 \pm 0.00}$	40.46 ± 0.14	49.64 ± 0.14
Protein Structure	2.48 ± 0.14	$\mathbf{0.57 \pm 0.15}$	2.62 ± 0.00	2.67 ± 0.00	4.44 ± 0.40	$\mathbf{4.12 \pm 0.00}$	226.21 ± 0.82	282.43 ± 1.57
Red Wine	6.26 ± 0.29	$\mathbf{5.84 \pm 0.95}$	1.57 ± 0.05	$\mathbf{1.39 \pm 0.02}$	0.68 ± 0.00	$\mathbf{0.66 \pm 0.00}$	9.14 ± 0.03	11.32 ± 0.01
White Wine	7.69 ± 0.43	$\mathbf{7.03 \pm 0.18}$	1.52 ± 0.01	$\mathbf{1.38 \pm 0.01}$	0.76 ± 0.00	$\mathbf{0.74 \pm 0.00}$	25.09 ± 0.10	31.07 ± 0.03
Yacht Hydrodynamics	28.82 ± 3.20	$\mathbf{25.86 \pm 3.92}$	$\mathbf{2.40 \pm 0.03}$	2.45 ± 0.11	$\mathbf{3.72 \pm 0.27}$	4.04 ± 0.12	2.73 ± 0.03	3.34 ± 0.02
Year Prediction MSD	$2.55 \pm$ NA	$\mathbf{0.85 \pm}$ **NA**	$\mathbf{3.33 \pm 0.00}$	3.35 ± 0.00	$\mathbf{8.67 \pm 0.00}$	8.72 ± 0.00	$2814.87 \pm$ NA	3465.02 ± 0.00

Dataset	Deep Ensembles with Adverserial Training							
	Calibration Error(%)			RMSE			max likelihood	
	base+iso	QR	QR+iso	base+iso	QR	QR+iso	base+iso	QR+iso
Air Foil	32.92 ± 0.82	$\mathbf{19.04 \pm 1.44}$	24.01 ± 3.19	$\mathbf{3.17 \pm 0.07}$	3.43 ± 0.03	3.41 ± 0.02	17898.23	3621.50
Boston Housing	42.51 ± 3.91	$\mathbf{31.22 \pm 4.09}$	33.15 ± 1.37	4.67 ± 0.09	4.46 ± 0.09	$\mathbf{4.42 \pm 0.09}$	71276.14	3234.50
Concrete Strength	47.52 ± 2.54	$\mathbf{30.85 \pm 2.40}$	30.90 ± 1.29	8.89 ± 0.10	8.70 ± 0.01	$\mathbf{8.68 \pm 0.06}$	340786.93	262215.09
Fish Toxicity	4.43 ± 0.19	2.62 ± 0.24	$\mathbf{1.89 \pm 0.03}$	0.92 ± 0.01	$\mathbf{0.91 \pm 0.00}$	$\mathbf{0.91 \pm 0.00}$	226521.01	175425.39
Kin8nm	3.81 ± 0.12	$\mathbf{1.04 \pm 0.16}$	1.77 ± 0.03	$\mathbf{0.07 \pm 0.00}$	$\mathbf{0.07 \pm 0.00}$	$\mathbf{0.07 \pm 0.00}$	15236.93	14855.14
Protein Structure	$\mathbf{0.06 \pm 0.00}$	0.57 ± 0.15	$\mathbf{0.06 \pm 0.00}$	4.44 ± 0.40	$\mathbf{4.12 \pm 0.00}$	$\mathbf{4.12 \pm 0.00}$	10572.97	10629.57
Red Wine	14.09 ± 0.61	5.84 ± 0.95	$\mathbf{4.93 \pm 0.60}$	0.68 ± 0.00	$\mathbf{0.66 \pm 0.00}$	$\mathbf{0.66 \pm 0.00}$	287680.56	225613.04
White Wine	16.67 ± 0.52	$\mathbf{7.03 \pm 0.18}$	9.93 ± 0.16	0.76 ± 0.00	$\mathbf{0.74 \pm 0.00}$	$\mathbf{0.74 \pm 0.00}$	42681.30	44115.77
Yacht Hydrodynamics	80.15 ± 8.59	$\mathbf{25.86 \pm 3.92}$	37.86 ± 6.28	$\mathbf{3.76 \pm 0.25}$	4.04 ± 0.12	4.04 ± 0.09	2072.53	2493.95
Year Prediction MSD	$0.06 \pm$ NA	$0.85 \pm$ NA	$\mathbf{0.04 \pm}$ **NA**	$\mathbf{8.67 \pm 0.00}$	8.72 ± 0.00	8.72 ± 0.00	284.35	365.21

Table 2: Base Model is Deep Ensemble model and QR is when Base Model is trained with Quantile Regularization. (Base/QR) +iso represents after applying Isotonic Calibration. We use $\lambda = 4$

whereas for Year Prediction MSD there is a single split where we train on 463715 points and test on 51630 points. This experiment is repeated 3 times and averages are reported except for year prediction MSD. We use 2 hidden layer network with 128 units with ReLU activation, and trained with Adam Optimizer with a learning rate of 10^{-2} . Results are presented in Table 1 and 2. Calibration plots and proofs for all claims can be found Supplementary in material. Code can be found at `https://github.com/occam-ra-zor/QR`

5.3 Discussion

Both, in case of Dropout VI and Deep Ensembles, calibration error improves when trained with quantile regularizer. Also in some cases, Isotonic Calibration increases the calibration error (Ex. for dropout VI see Concrete Strength, Yacht Hydrodynamics and for Deep Ensembles see Bouston Housing and Yacht Hydrodynamics). The reason is that unlike classification calibration , *recalibration dataset* in case of regression calibration already satisfies monotonicity constraint, So isotonic regression completely overfits in case of smaller calibration datasets. For pretty large calibration datasets like protein structure (calib dataset size $m = 36,584$) and Year Prediction MSD (calib dataset size $m = 463715$), Isotonic Regression has better improvement . Although, it can work well in such setting, PDF after calibration becomes unusable. QR not only works well on small datasets but also works well in very large datasets like Year prediction MSD(nearly 60 % improvement in both architectures) . Also, there is small reduction in RMSE,NLL in models trained with QR, but again it is negligible. The reason for this is that if we plainly optimize for calibration there exists a non-degenerate case as solution apart from true distribution as we discussed in Sec. 4.3

6 Conclusion

We have proposed a black-box calibration loss function that can be used in any probabilistic regression models. Unlike current methods for quantile calibration, our method is *implicit* in nature and does not require an additional calibration dataset and, more importantly, the smoothness of the PDF is not lost. We conduct experiments to show effectiveness of our proposed method.

REFERENCES

Jan Beirlant and Edward J Dudewicz. Nonparametric entropy estimation: An overview. *International Journal of Mathematical and Statistical Sciences*, 6(1):17–39, 1997.

Glenn W Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

Song Xi Chen. Beta kernel estimators for density functions. *Computational Statistics & Data Analysis*, 31(2):131–145, 1999.

Tom Fawcett and Alexandru Niculescu-Mizil. Pav and the roc convex hull. *Machine Learning*, 68 (1):97–106, 2007.

Yarin Gal. Uncertainty in deep learning. *University of Cambridge*, 1:3, 2016.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(2): 243–268, 2007.

Aditya Grover, Eric Wang, Aaron Zweig, and Stefano Ermon. Stochastic optimization of sorting networks via continuous relaxations. *arXiv preprint arXiv:1903.08850*, 2019.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1321–1330. JMLR. org, 2017.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Gil Keren, Nicholas Cummins, and Björn Schuller. Calibrated prediction intervals for neural network regressors. *IEEE Access*, 6:54033–54041, 2018.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. *arXiv preprint arXiv:1807.00263*, 2018.

Meelis Kull, Telmo Silva Filho, and Peter Flach. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In *Artificial Intelligence and Statistics*, pp. 623–631, 2017.

Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. In *Advances in Neural Information Processing Systems*, pp. 12295–12305, 2019.

Ananya Kumar, Percy S Liang, and Tengyu Ma. Verified uncertainty calibration. In *Advances in Neural Information Processing Systems*, pp. 3787–3798, 2019.

Aviral Kumar, Sunita Sarawagi, and Ujjwal Jain. Trainable calibration measures for neural networks from kernel mean embeddings. In *International Conference on Machine Learning*, pp. 2805–2814, 2018.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems*, pp. 6402–6413, 2017.

Erik G Learned-Miller et al. Ica using spacings estimates of entropy. *Journal of machine learning research*, 4(Dec):1271–1295, 2003.

Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for bayesian uncertainty in deep learning. In *Advances in Neural Information Processing Systems*, pp. 13132–13143, 2019.

Patrick Mair, Kurt Hornik, and Jan de Leeuw. Isotone optimization in r: pool-adjacent-violators algorithm (pava) and active set methods. *Journal of statistical software*, 32(5):1–24, 2009.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems*, pp. 4696–4705, 2019.

Allan H Murphy. Scalar and vector partitions of the probability score: Part i. two-state situation. *Journal of Applied Meteorology*, 11(2):273–282, 1972.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017.

John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

Hao Song, Tom Diethe, Meelis Kull, and Peter Flach. Distribution calibration for regression. *arXiv preprint arXiv:1905.06023*, 2019.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Sunil Thulasidasan, Gopinath Chennupati, Jeff A Bilmes, Tanmoy Bhattacharya, and Sarah Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 13888–13899, 2019.

Juozas Vaicenavicius, David Widmann, Carl Andersson, Fredrik Lindsten, Jacob Roll, and Thomas B Schön. Evaluating model calibration in classification. *arXiv preprint arXiv:1902.06977*, 2019.

Oldrich Vasicek. A test for normality based on sample entropy. *Journal of the Royal Statistical Society. Series B (Methodological)*, 38(1):54–59, 1976. ISSN 00359246. URL `http://www.jstor.org/stable/2984828`.

Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, volume 1, pp. 609–616. Citeseer, 2001.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 694–699, 2002.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.