

# Building Calibrated Deep Models via Uncertainty Matching with Auxiliary Interval Predictors

Jayaraman J. Thiagarajan<sup>†\*</sup>, Bindya Venkatesh<sup>‡</sup>, Prasanna Sattigeri<sup>+</sup>,  
Peer-Timo Bremer<sup>†</sup>

<sup>†</sup>Lawrence Livermore National Laboratory, <sup>‡</sup>Arizona State University, <sup>+</sup>IBM Research AI  
jjayaram@llnl.gov, bindya.venkatesh@asu.edu, psattig@us.ibm.com, bremer5@llnl.gov

## Abstract

With rapid adoption of deep learning in critical applications, the question of when and how much to trust these models often arises, which drives the need to quantify the inherent uncertainties. While identifying all sources that account for the stochasticity of models is challenging, it is common to augment predictions with confidence intervals to convey the expected variations in a model’s behavior. We require prediction intervals to be well-calibrated, reflect the true uncertainties, and to be sharp. However, existing techniques for obtaining prediction intervals are known to produce unsatisfactory results in at least one of these criteria. To address this challenge, we develop a novel approach for building calibrated estimators. More specifically, we use separate models for prediction and interval estimation, and pose a bi-level optimization problem that allows the former to leverage estimates from the latter through an *uncertainty matching* strategy. Using experiments in regression, time-series forecasting, and object localization, we show that our approach achieves significant improvements over existing uncertainty quantification methods, both in terms of model fidelity and calibration error.

## Introduction

Data-driven learning methods, in particular deep learning, have led to incredible advances in a wide variety of commercial AI applications, and are rapidly being adopted in high impact applications such as healthcare, autonomous driving, and scientific discovery. Consequently, it has become critical to enable deep models to reliably assess their confidence in scenarios different from the training regime and to verify that predictions arise from generalizable patterns rather than artifacts in the training data. In practice, a variety of sources including data sampling errors, model approximations and the inherent data uncertainties can contribute to stochasticity in deep models (?). Hence, capturing these uncertainties when making predictions can shed light onto when and how much to trust the results (??; ??; ?;

?). Furthermore, accurate uncertainty estimates are essential for solving a variety of practical machine learning challenges including knowledge transfer (?), active learning (?), and anomaly detection (?).

In recent years, several classes of uncertainty estimation techniques have been proposed (??; ??; ??; ?). An inherent limitation of these uncertainty quantification (UQ) techniques is the lack of suitable validation strategies. A typical workaround is to evaluate prediction intervals (PIs) via their calibration, without distinguishing the uncertainty sources (?): A PI is well calibrated if the likelihood of true target falling in the interval is consistent with the confidence level of the interval. In the context of calibration, most existing UQ methods in deep learning are not inherently calibrated (?).

In this paper, we argue that producing well-calibrated prediction intervals requires the identification of informative and uninformative uncertainties. When a model is equally confident in all parts of the data space, the uncertainties are considered uninformative. In practice, it is typical to use an additional recalibration dataset to refine the uncertainty estimates. In contrast, we propose to effectively leverage the informative uncertainties from a model through the use of an auxiliary interval estimator that is exclusively optimized for the calibration objective, and a novel *uncertainty matching* process. More specifically, we use a primary network for producing mean estimates with prediction uncertainties and an auxiliary network for obtaining calibrated PIs, and formulate a bi-level optimization approach for jointly training the two networks. To this end, we develop two *uncertainty matching* strategies to leverage the intervals during the mean estimation process, namely *Sigma Fit* and *IQR Fit*. By incorporating the calibration step into the training process, our method is able to produce predictive models with significantly improved accuracy as well as better calibration.

## Contributions:

- A novel framework to produce reliable prediction intervals in continuous-valued regression, that are sharp and well calibrated for any required confidence level;
- An approach to estimate means and prediction intervals as a bi-level optimization and an alternating optimization

<sup>\*</sup>This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

strategy for model inference;

- A new regularization for estimating means along with prediction uncertainties based on *uncertainty matching*;
- Two different *uncertainty matching* strategies, namely *Sigma Fit* and *IQR Fit*; and
- A rigorous evaluation on different use-cases and model architectures including regression with FCNs, time-series forecasting with LSTMs and object localization with CNNs, all producing better generalization performance and calibration error compared to the state of the art.

## Related Work

Broadly, there are two important sources of prediction uncertainties in deep models: epistemic uncertainty, also known as model uncertainty that can be eliminated given enough training data, and aleatoric uncertainty, which depends on stochasticity inherent to the data. A variety of estimators have been proposed in the literature for measuring these uncertainties, most often with classification models. For example, Bayesian neural nets (?), Monte-Carlo dropout (?), concrete dropout (?) and ensemble techniques (?) are commonly utilized to estimate the epistemic uncertainties. On the other hand, Tagasovska *et al.* recently proposed a conditional quantile based estimator for measuring aleatoric uncertainties (?).

Due to the lack of suitable evaluation mechanisms for validating the quality of these estimates, it is common to utilize calibration as a quality metric. Interestingly, it has been reported in several studies that these estimators are not inherently calibrated (?). Consequently, a large class of techniques that are aimed at calibrating pre-trained classifiers has been developed (??; ??; ?). In comparison, calibration for regression tasks has received limited attention (??; ?). For example, a two-stage recalibration technique was proposed in (?) which re-calibrates the PI estimates from the classifier model. This approach requires sufficient i.i.d. data to perform recalibration and can arbitrarily alter the true uncertainties to achieve the calibration objective (?). Another important limitation of recalibration methods is that they do not distinguish between informative and uninformative uncertainties. In order to circumvent this challenge, recently, Levi *et al.* proposed to match the heteroscedastic variances to the residuals from the predictions (?), through a strategy similar to temperature scaling (?). In contrast, our approach attempts to utilize the informative uncertainties for producing calibrated intervals, through an auxiliary interval estimator model and a novel *uncertainty matching* strategy.

## Proposed Approach

In this section, we describe the proposed approach (see Figure 1) for building deep predictive models, which employs a novel *uncertainty matching* strategy to produce estimators that are both accurate and well-calibrated.

**Notations.** Denoting the input data by  $\mathbf{x} \subset \mathcal{X} \in \mathbb{R}^d$ , our goal is to build a predictive model  $\mathcal{F} : \mathbf{x} \mapsto \mathbf{y}$ , where  $\mathbf{y} \subset \mathcal{Y} \in \mathbb{R}$  and an interval estimator  $\mathcal{I} : \mathbf{x} \mapsto \mathbb{R}^2$ . Here,  $\mathcal{X}$  and  $\mathcal{Y}$  indicate spaces of inputs and outputs respectively.

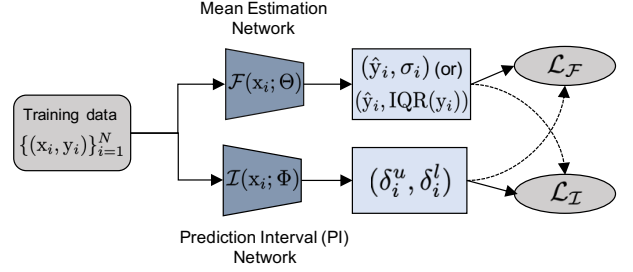


Figure 1: An illustration of our approach to produce PI, for any specified confidence  $\alpha$ , and perform PI-regularized mean estimation via a bi-level optimization technique.

Note, the model  $\mathcal{I}$  produces  $\delta^u$  and  $\delta^l$ , which denote widths of the interval in the positive/negative direction respectively, thus producing a total interval width  $w = \delta^u + \delta^l$ . The models  $\mathcal{F}$  and  $\mathcal{I}$  are described using parameters  $\Theta$  and  $\Phi$  respectively. A labeled dataset of  $N$  samples is comprised of tuples  $\{(\mathbf{x}_i, \mathbf{y}_i)\}, i = 1, \dots, N$ . We denote all estimated quantities with a hat symbol, e.g.  $\hat{\mathbf{y}} = \mathcal{F}(\mathbf{x})$ . Though our focus is on predicting continuous-valued outputs, the proposed techniques can be easily generalized to classification problems.

## Formulation

In this paper, we hypothesize that integrating the task of calibration into the predictive modeling process can provide useful regularization to its optimization, thus producing higher fidelity models that are also well-calibrated. To this end, our approach employs two separate models – one for predicting the target and the other for estimating the prediction intervals, and pose a bi-level optimization formulation that allows the mean estimator to identify prediction uncertainties that are the most informative for matching the intervals from the PI estimator. Let  $\mathcal{F}(\mathbf{x}; \Theta)$  denote the mean estimator and  $\mathcal{I}(\mathbf{x}; \Phi^*)$  denote the PI estimator parameterized by  $\Theta$  and  $\Phi$  respectively. The overall objective is

$$\begin{aligned} \min_{\Theta} \mathcal{L}_{\mathcal{F}}(\Theta; \mathbf{x}, \mathbf{y}, \mathcal{I}(\mathbf{x}; \Phi^*)), \\ \text{s.t. } \Phi^* = \arg \min_{\Phi} \mathcal{L}_{\mathcal{I}}(\Phi; \mathbf{x}, \mathcal{F}(\mathbf{x}; \Theta)). \end{aligned} \quad (1)$$

Here, the first term represents an improved mean estimator that takes the intervals into account for identifying informative uncertainties, and the second term exclusively optimizes for calibration, based on the current mean estimator. Intuitively, the auxiliary task of obtaining prediction intervals, i.e., minimizing  $\mathcal{L}_{\mathcal{I}}$ , is used to regularize the primary task of fitting a predictive model by constraining its uncertainty estimates to match the estimated intervals. We refer to this process as *uncertainty matching*. Though this can be implemented with any existing uncertainty estimation technique, we use heteroscedastic neural networks (HNN) and conditional quantile estimators in our setup.

**PI Estimator Design.** While the mean estimate is a random variable, an interval estimate is a random interval. For an output variable  $y$ , let  $y^l$  and  $y^u$  denote its lower and upper bounds. While an interval does or does not contain a

certain value, a random interval has a certain probability of containing a value. Suppose that  $p(y^l \leq y \leq y^u) = \alpha$ , where  $\alpha \in [0, 1]$ , then the random interval  $[y^l, y^u]$  is referred as a  $100 \times \alpha\%$  interval of  $y$ . When this holds, we expect that when we sample a large number of  $x_i$  from a validation set and predict  $\mathcal{F}(x_i)$  and  $\mathcal{I}(x_i)$ , then  $y_i$  will be contained in  $\mathcal{I}(x_i)$   $100 \times \alpha\%$  of the trials.

The PI estimator network is designed to produce intervals that are well-calibrated and can reflect the underlying prediction uncertainties. In contrast to existing approaches which attempt to produce uncertainties that can be calibrated to any  $\alpha$  by applying appropriate scaling, our method operates with a specific  $\alpha$  and implements the calibration task as a differentiable function. As expressed in Eq. (1), the loss  $L_{\mathcal{I}}$  is constructed based on the current state of  $\Theta$  for the mean estimator network  $\mathcal{F}$ . We use the following term to optimize for the quality of calibration, which measures the empirical probability of the true target lying in the estimated intervals:

$$L_{emce} = \left| \alpha - \frac{1}{N} \sum_{i=1}^N \mathbb{1} \left[ (\hat{y}_i - \delta_i^l) \leq y_i \leq (\hat{y}_i + \delta_i^u) \right] \right|, \quad (2)$$

where  $\mathbb{1}$  is the indicator function and  $\hat{y}_i = \mathcal{F}(x_i; \Theta)$  are the mean predictions. We implement the indicator function as  $\text{Sigmoid}(\eta \times (y_i - y_i^l)(y_i^u - y_i))$ , where  $\hat{y}_i^l = \hat{y}_i - \delta_i^l$ ,  $\hat{y}_i^u = \hat{y}_i + \delta_i^u$ , and  $\eta$  is a large scaling factor. This empirical objective does not require any distributional assumption on data. Next, in order to ensure that the intervals reflect the underlying uncertainty, the widths  $w_i$  are matched to the residual  $r$  from  $\mathcal{F}$ .

$$L_{noise} = \sum_{i=1}^N \left\| \left( 0.5 * w_i - |r_i| \right) \right\|_1, \quad (3)$$

where  $\|\cdot\|_1$  denotes the  $\ell_1$  norm,  $w_i = \delta_i^u + \delta_i^l$  and the residual  $r_i = y_i - \mathcal{F}(x_i; \Theta)$ . Finally, the widths of estimated intervals are regularized to avoid non-trivial solutions of arbitrarily large intervals.

$$L_{sharp} = \sum_{i=1}^N \left\| \hat{y}_i^u - y_i \right\|_1 + \left\| y_i - \hat{y}_i^l \right\|_1. \quad (4)$$

Hence the overall objective can be written as:

$$\begin{aligned} \Phi^* &= \arg \min_{\Phi} L_{\mathcal{I}} \\ &= \arg \min_{\Phi} L_{emce} + \beta_n L_{noise} + \beta_s L_{sharp}, \end{aligned} \quad (5)$$

where  $\beta_n, \beta_s$  are penalty terms.

**Mean Estimation with Uncertainty Matching.** As formulated in Eq. (1), the goal is to leverage the calibrated intervals from the PI estimator, through uncertainty matching, to optimize for parameters  $\Theta$ . The  $\mathcal{F}$  network is designed to predict the target along with uncertainty estimates, and the loss function  $L_{\mathcal{F}}$  is designed such that the fidelity of the predictions is measured with respect to the true targets, while the uncertainties are guided by the interval estimates from  $\mathcal{I}$ . We find that, by allowing the structure of estimated uncertainties to follow the pseudo-supervisory signal from  $\mathcal{I}$ , one can obtain higher quality estimates  $\hat{y}$ . Now, we explain the two proposed strategies for uncertainty matching.

**Algorithm 1:** Building calibrated deep predictive models.

**Input:** Labeled data  $\{(x_i, y_i)\}_{i=1}^N$ , Desired calibration level  $\alpha$ , Number of epochs  $n_m$  and  $n_c$ .

**Output:** Trained mean and interval estimators  $\mathcal{F}$  and  $\mathcal{I}$

**Initialization:** Randomly initialize parameters

$\Theta^*, \Phi^*$ ;

**while not converged do**

**for**  $n_m$  epochs **do**

    Compute intervals  $\delta_i^u, \delta_i^l = \mathcal{I}(x_i; \Phi^*)$ ;

    Compute loss function  $L_{\mathcal{F}}$  using Eq. (7) for

*Sigma Fit* or Eq. (10) for *IQR Fit*;

    Update  $\Theta^* = \arg \min_{\Theta} L_{\mathcal{F}}$ ;

**end**

**for**  $n_c$  epochs **do**

    Obtain predictions  $\hat{y}_i = \mathcal{F}(x_i; \Theta^*)$ ;

    Compute loss function  $L_{\mathcal{I}}$  using Eq. (5);

    Update  $\Phi^* = \arg \min_{\Phi} L_{\mathcal{I}}$ ;

**end**

**end**

**a. Sigma Fit:** In this variant, the predictive model  $\mathcal{F}$  is designed as a heteroscedastic neural network, i.e. it returns mean and variance estimates  $(\hat{y}_i, \hat{\sigma}_i^2)$  for a given sample  $x_i$ , and is optimized using the heteroscedastic loss (?). Note that, this loss function is defined by assuming the predictions to follow a Gaussian distribution, and the estimated variances from the HNN can be used to estimate intervals for different levels of calibration  $\alpha$ ,

$$\text{PI} = [\hat{y} - z_{(1-\alpha)/2} \hat{\sigma}, \hat{y} + z_{(1-\alpha)/2} \hat{\sigma}], \quad (6)$$

where  $z$  indicates the  $z$ -score (i.e.  $z_{(1-\alpha)/2} = 1.96$ , for 95% calibration). We enforce the estimated  $\hat{\sigma}$  to match the appropriately scaled version of the interval width  $w$ , based on the calibration level achieved by  $\mathcal{I}$  on the training dataset. Formally, the optimization objective can be written as:

$$L_{\mathcal{F}} = \frac{1}{N} \sum_{i=1}^N \frac{\|y_i - \mathcal{F}(x_i; \Theta)\|_2^2}{2\sigma_i^2} + \frac{1}{2} \log \sigma_i^2 + \lambda_m \|\sigma_i - \gamma \frac{w_i}{2}\|_1, \quad (7)$$

where  $\lambda_m$  is a penalty term, the scale  $\gamma = 1/z_{(1-\alpha_v)/2}$  with  $\alpha_v$  being the calibration level achieved by  $\mathcal{I}$ .

**b. IQR Fit:** In this strategy, we propose to obtain uncertainty estimates based on conditional quantiles of the target  $y$  and enforce them to match the intervals from the PI estimator. Let  $F(y) = P(y \leq y)$  be the strictly monotonic cumulative distribution function of  $y$  assuming real values  $y$  and  $F^{-1}(\tau) = \inf\{y : F(y) \geq \tau\}$  denotes the quantile distribution function for  $0 \leq \tau \leq 1$ . Quantile regression is aimed at inferring a desired quantile  $\tau$  for the target  $y$ , when the input  $x = x$ , i.e.,  $F^{-1}(\tau|x = x)$ . This model can be estimated using the *pinball* loss (?):

$$L_{\tau}(y, \hat{y}) = \begin{cases} \tau(y - \hat{y}), & \text{if } y - \hat{y} \geq 0, \\ (1 - \tau)(\hat{y} - y), & \text{otherwise.} \end{cases} \quad (8)$$

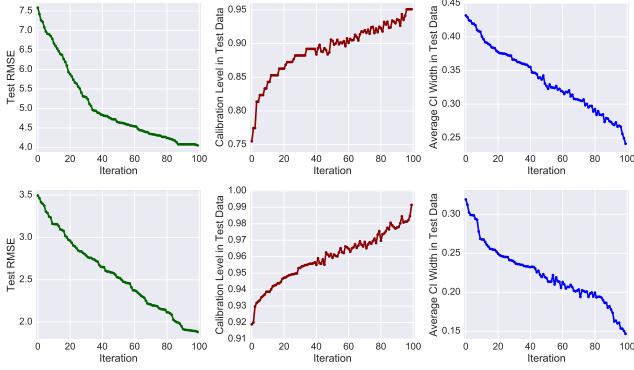


Figure 2: Convergence behavior of the proposed approach on the held-out test data for the Boston housing (top) and Parkinsons (bottom) datasets, for a target calibration level of 95% and 99% respectively : As the algorithm progresses, we observe that both the RMSE and calibration consistently improve. At the same time, the interval widths become sharp even while achieving high calibration.

Recently, (?) showed that the *inter-quantile range* (IQR) between upper and lower quantiles can produce calibrated estimates of the aleatoric uncertainties. Hence, we propose an uncertainty estimator similar to (?), where the  $\alpha$  prediction interval can be obtained around the median as:

$$\text{IQR}(x; \tau^u, \tau^l) = \mathcal{F}(x|\tau = \tau^u, \Theta) - \mathcal{F}(x|\tau = \tau^l, \Theta), \quad (9)$$

$$\tau^u = (1 + \alpha)/2, \tau^l = (1 - \alpha)/2.$$

Given the interval estimates from the PI network, the loss function  $L_{\mathcal{F}}$  can be constructed as:

$$L_{\mathcal{F}} = \frac{1}{N} \sum_{i=1}^N \|y_i - \mathcal{F}(x_i; \Theta)\|_2^2 + \lambda_u L_{\tau^u}(y_i, \mathcal{F}(x_i; \Theta)) + \lambda_l L_{\tau^l}(y_i, \mathcal{F}(x_i; \Theta)) + \lambda_m \|\text{IQR}(x_i) - w_i\|_1. \quad (10)$$

## Algorithm

The overall algorithm, which repeats the regularized mean estimation and the interval prediction in an alternating fashion, is provided in Algorithm 1. The mean estimator takes the intervals into account for its update. Similarly, we include the residual structure from the mean estimator for interval estimation. For a fixed interval, the improved mean estimate can increase the calibration error (CE) by achieving a higher likelihood even for smaller  $\alpha$ . However, when re-updating the interval estimator, we expect the widths to become sharper to reduce CE. This collaborative process thus leads to superior quality mean estimates and highly calibrated intervals. While each of the steps are run for  $n_m$  and  $n_c$  epochs respectively, the outer loop is executed until convergence. In all our experiments, both hyper-parameters  $n_m$  and  $n_c$  are set at 10 and the algorithm returns optimized parameter estimates for  $\mathcal{F}$  and  $\mathcal{I}$ . It is important to note that, we do not require any additional recalibration step for the validation/testing stages, in contrast to approaches such as (?; ?). Figure 2 illustrates the convergence behavior of the proposed approach for a regression task on two datasets. We

observe that, as the algorithm progresses, the proposed approach produces well-calibrated intervals that are also sharp (measured using the average of interval widths  $w_i$ ). More importantly, the uncertainty matching based regularization produces much improved quality predictors, as evidenced by the improvements to the prediction quality (measured using RMSE).

## Empirical Studies

In this section, we present empirical studies to compare the proposed approach against popularly adopted methods for producing prediction intervals. More specifically, we considered three different use cases where the goal was to predict continuous valued target variables – regression, forecasting with time-series data and object localization in images (i.e. predicting bounding boxes).

### Experiment Setup

**Hyper-parameter Choices:** We used Algorithm 1 to solve the optimization problem in Eq. (1), based on the two proposed strategies, namely *Sigma Fit* and *IQR Fit*. We used the following hyper-parameter choices for all experiments: The penalties for the loss  $\mathcal{L}_{\mathcal{I}}$  in Eq. (5) were fixed at  $\beta_n = 0.1$  and  $\beta_s = 0.3$  respectively. For the *Sigma Fit* method, the penalty for the uncertainty matching term was set at  $\lambda_m = 0.5$ . Similarly, the hyper-parameters for constructing the loss  $\mathcal{L}_{\mathcal{F}}$  for *IQR Fit*, we used  $\lambda_m = 0.4, \lambda_u = \lambda_l = 0.3$ . While the outer loop in Algorithm 1 was run until convergence, both the mean estimator and the PI estimator networks were trained for  $n_m = n_c = 10$  epochs in each iteration.

**Evaluation Metrics:** For both regression and time-series forecasting applications, we used the standard root mean square error (RMSE) metric to evaluate the quality of mean predictions. In the case of object localization, we used the intersection-over-union (IoU) metric, that measures the ratio between the area under intersection of the true and predicted bounding boxes, and the area under union of both bounding boxes. Since our primary objective is to produce calibrated estimators, we evaluate the quality of prediction intervals on the held-out validation set using the calibration error metric,  $CE_{\alpha}$ , which is the same as the  $L_{emce}$  term in Eq. (2). In order to ensure that the PI estimator network does not produce a degenerate solution (arbitrarily large intervals that are always calibrated), we included the  $L_{sharp}$  term in Eq. (5). Hence, we also evaluate how well that constraint is met using the average width metric:  $AW_{\alpha} = \frac{1}{N} \sum_{i=1}^N (\delta_i^l + \delta_i^u)$ .

**Baselines:** We compare against the following baseline techniques that are commonly adopted to produce prediction intervals, in lieu of simple point-estimates. Though there exists an entire class of recalibration methods (?), which utilizes an additional recalibration dataset to refine the learned estimator with a calibration objective, our focus is on single-shot calibration methods.

**MC Dropout (?):** In this approach, dropout in deep neural networks is cast as an approximate Bayesian inference in deep Gaussian processes. Using dropout during both train and test phases simulates Monte-Carlo sampling from the network with non-deterministic weights, thus captur-

Table 1: Test RMSE from the regression experiments on 6 UCI benchmark datasets. For all cases, the results showed are for the case of  $\alpha = 0.9$ , i.e. 90% calibration. The results reported were obtained by averaging over 5 random trials.

Method	Crime	Red Wine	White Wine	Parkinsons	Boston	Auto MPG
MC Dropout	<b>0.139</b>	0.686	0.748	2.78	5.51	4.61
Concrete Dropout	0.144	0.745	0.764	3.1	5.88	5.19
Quantile Estimator	0.152	0.737	0.782	2.91	4.802	4.32
BNN	0.159	0.823	0.871	2.493	4.025	<b>3.07</b>
HNN	<b>0.138</b>	0.714	0.757	2.842	4.762	3.98
Sigma Fit	<b>0.138</b>	0.599	0.702	<b>2.017</b>	4.088	3.23
IQR Fit	<b>0.138</b>	<b>0.581</b>	<b>0.691</b>	2.206	<b>4.011</b>	3.18

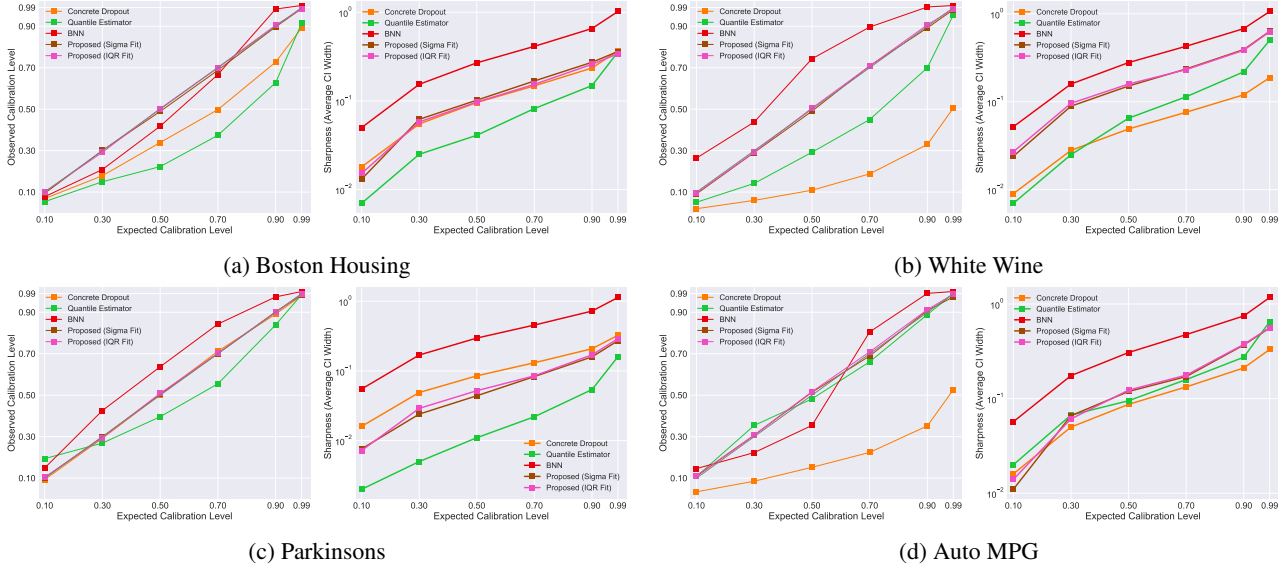


Figure 3: Evaluating the quality of estimated intervals - For each dataset, we plot the expected calibration vs observed calibration on test data to demonstrate the generality of the proposed PI estimator, when compared to baselines. Further, we plot expected calibration vs average PI width to show how the estimated intervals compare to existing uncertainty estimators.

ing epistemic uncertainties. For all experiments, the dropout probability was set at 0.5.

*Concrete Dropout* (?): This was designed to automatically tune the dropout probability based on data characteristics, and hence it is expected to produce better calibrated intervals in comparison to MC Dropout.

*Bayesian Neural Networks (BNN)* (?): In this approach, the underlying distribution on network weights are approximated with a Gaussian (variational approximation), and the posterior inference based on observed data is then carried out by sampling from this approximate distribution, thus capturing epistemic uncertainties.

*Quantile Estimator* (?): Inter-quantile range from conditional quantiles of predictors can be used to estimate the aleatoric uncertainties and hence prediction intervals for a given  $\alpha$  confidence can be obtained by choosing appropriate upper and lower quantiles.

*Heteroscedastic Neural Networks (HNN)* (?): Using the Gaussian likelihood formulation, the data dependent observation noise, i.e. aleatoric uncertainties, can be captured

through the heteroscedastic loss.

## Regression using Fully Connected Networks

In this experiment, we considered the problem of regressing to a continuous target variable from high-dimensional features. For each dataset, we used a random 80/20 split for train/test, and we report the averaged results obtained from 5 random trials. As a preprocessing step, we rescaled the target data into the range  $[0, 1]$  before training the models.

**Datasets:** We considered 6 datasets from the UCI repository (?), which ranged between 7 and 124 in their dimensionality, and between 391 and 4898 in terms of sample size: crime, red wine quality, white wine quality, parkinsons, boston housing and auto mpg.

**Model Architecture:** For all approaches, we used a fully connected network with 5 linear layers, each followed by a ReLU activation function, and the MSE loss for training.

**Results:** A meaningful interval is one that reflects the informative uncertainties in data or modeling process and is well calibrated. Hence, we measure the quality of calibra-

Table 2: Results from the time-series forecasting experiments - We report the test RMSE, calibration error and average width metrics for each of the cases, when  $\alpha$  was set to 0.95.

Method	Metric	NSW2013	Bike Sharing	TAS2016	Traffic	Beijing	Air Quality
MC Dropout	RMSE	125.48	62.68	22.12	2.03E+08	22.42	0.66
	CE <sub>0.95</sub>	0.04	0.30	0.196	0.037	0.177	0.02
	AW <sub>0.95</sub>	587.54	90.94	50.37	8.39E+08	38.53	2.52
Concrete Dropout	RMSE	153.16	64.81	23.11	2.25E+08	22.46	0.65
	CE <sub>0.95</sub>	0.046	0.277	0.145	0.073	0.12	0.105
	AW <sub>0.95</sub>	696.19	96.67	53.58	8.93E+08	48.42	2.73
Quantile Estimator	RMSE	131.18	65.19	23.44	2.07E+08	22.45	0.77
	CE <sub>0.95</sub>	0.07	0.22	0.19	0.07	0.11	0.241
	AW <sub>0.95</sub>	533.31	99.34	49.17	8.6E+08	49.17	2.71
BNN	RMSE	153.16	64.81	23.11	2.25E+08	22.46	0.65
	CE <sub>0.95</sub>	0.046	0.277	0.145	0.073	0.12	0.105
	AW <sub>0.95</sub>	696.19	96.67	53.58	8.93E+08	48.42	2.73
HNN	RMSE	124.59	64.38	22.81	1.36E+08	22.83	0.72
	CE <sub>0.95</sub>	0.008	0.004	0.013	0.014	0.028	0.041
	AW <sub>0.95</sub>	452.05	199.64	91.24	5.62E+08	98.95	6.16
Sigma Fit	RMSE	<b>90.31</b>	58.97	17.95	1.27E+08	<b>18.65</b>	<b>0.53</b>
	CE <sub>0.95</sub>	<b>0.002</b>	0.005	0.009	0.006	0.011	0.007
	AW <sub>0.95</sub>	424.59	169.81	77.71	5.49E+08	88.19	2.99
IQR Fit	RMSE	94.85	<b>53.65</b>	<b>17.64</b>	<b>1.21E+08</b>	19.43	0.62
	CE <sub>0.95</sub>	0.003	<b>0.002</b>	<b>0.005</b>	<b>0.003</b>	<b>0.01</b>	<b>0.004</b>
	AW <sub>0.95</sub>	421.63	182.34	80.44	5.43E+08	87.48	3.11

tion achieved using the proposed approach at different calibration levels  $\alpha$  - (i) *expected calibration vs observed calibration*; and (ii) *expected calibration vs average PI width*. While the former evidences the generality of the PI estimator to unseen test data, the latter shows how the estimated intervals compare to conventional uncertainty estimators. Similar to the findings of existing studies, we also find that existing approaches such as MC Dropout, Concrete Dropout and BNN are not well calibrated. In comparison, we found the heteroscedastic neural networks to be better calibrated. Figure 3 shows the results for 4 different datasets (rest in supplementary) and we find that the proposed approach achieves significantly improved calibration, when compared to the baseline methods. Interestingly, from the *average PI width* plots, we find that the estimated intervals are typically larger than the *aleatoric* uncertainties from the quantile estimator, while being much lower than the estimates from BNNs. Finally, from Table 1, it is apparent the *uncertainty matching* process leads to improved regularization, thus producing higher-quality mean estimates.

### Time-series Forecasting using LSTMs

In this study, we consider the problem of forecasting in time-varying data by leveraging information from the past samples. More specifically, we attempt to look-ahead by one sample  $x[t+1]$  using the observations  $x[t-k:t+1]$ .

**Datasets:** We considered 6 time-series datasets: NSW2013 and TAS2016 annual electricity demand recordings from the

Australian Energy Market Operator; bike sharing, air quality, Beijing PM2.5 and network traffic datasets from UCI (?). In each case, we used  $k = 24$  steps in the past to look ahead, and overall we used the first 30% of the time-steps for training and the rest for testing.

**Model Architecture:** Our architecture comprised of a RNN with LSTM units (2 layers with 128 hidden dimensions) and a linear layer for making the final prediction.

**Results:** For this study, we ran all experiments with the desired level of calibration fixed at  $\alpha = 0.95$ , and report the RMSE,  $CE_{0.95}$  and  $AW_{0.95}$  metrics obtained using the different approaches in Table 2. As it can be seen, the proposed alternating optimization achieves the lowest calibration error in all cases, and more importantly, consistently produces higher-fidelity mean estimators. Figure 4 illustrates the prediction intervals obtained using *IQR Fit* for a few cases, where the quality of calibration is clearly evident.

### Object Localization using CNNs

The third study involved the application of the proposed approaches to object localization in images using convolutional neural networks. We posed the problem of localizing as predicting a bounding box around the object. Each bounding box is characterized by 4 regression targets –  $(x, y)$  coordinates of the left-bottom corner, height  $h$  and width  $w$ .

**Datasets:** For this experiment, we used the Caltech Birds (CUB) dataset (?), which is comprised of 11,788 images belonging 200 different categories of birds.



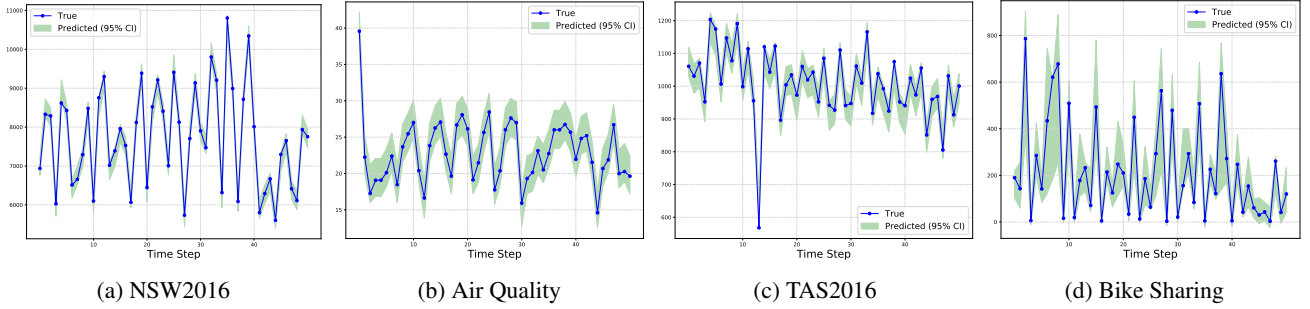


Figure 4: *Forecasting*: Prediction intervals obtained using the proposed *IQR Fit* method for the calibration level  $\alpha = 0.95$  on the test data. It can be seen that in all cases, our method produces sharp intervals that are well calibrated.

Table 3: Results from the object localization experiments - We report the performance of the best performing baseline and the proposed *IQR Fit* at  $\alpha = 0.95$

	HNN		IQR Fit	
	CE <sub>0.95</sub>	AW <sub>0.95</sub>	CE <sub>0.95</sub>	AW <sub>0.95</sub>
$x$	0.11	31.72	<b>0.01</b>	18.67
$y$	0.14	26.33	<b>0.02</b>	16.42
$w$	0.12	28.87	<b>0.04</b>	20.51
$h$	0.15	29.14	<b>0.04</b>	21.13
<b>IoU</b>	<b>0.468</b>		<b>0.557</b>	

**Model Architecture:** We performed transfer learning from a ResNet-18 model, pre-trained on imagenet classification, to carry out this prediction. We allowed fine-tuning of the last residual block and the final prediction layer during training.

**Results:** For brevity, we report results only from the best performing baseline method, HNN, and the proposed *IQR Fit* technique. Similar to the forecasting experiment, we used  $\alpha = 0.95$ . As seen in Table 3, the proposed approach significantly outperforms the baseline in terms of calibration error, while also meeting the sharpness constraint enforced in our optimization. Finally, the IoU score improvement clearly demonstrates the fidelity of the mean estimator. Figure 5 illustrates the prediction intervals (mean prediction showed in green) obtained using *IQR Fit* for a few example cases.

## Conclusions

In this work, we proposed a framework to build calibrated models for continuous-valued regression problems. By formulating the problems of estimating means and prediction intervals as a bi-level optimization, we showed that the model can effectively capture the informative uncertainties to produce well-calibrated intervals. Further, we found that, this alternating optimization also leads to better regularization of the mean estimator, thus improving its generalization significantly. From our empirical studies, we find that the proposed approach is effective across different data modalities, model architectures and applications. Though our approach was demonstrated using HNN-based and quantile-



Figure 5: Illustration of prediction intervals estimated using the proposed *IQR Fit* approach. In each case, the mean prediction is marked in green.

based aleatoric uncertainty estimators, any existing UQ technique can be integrated into our approach. This work also emphasizes the need to rigorously study the process of producing calibrated intervals based on prediction uncertainties.

Voluptate eligendi ab hic maxime quisquam totam eveniet perferendis harum possimus iusto, doloreque magnam sequi, totam enim maiores, nihil quis esse quibusdam amet explicabo iure ut modi laudantium, laboriosam asperiores ipsum nulla dicta aperiam laborum tempora?Libero laboriosam hic corrupti enim est rem recusandae, cupiditate saepe dignissimos facere error eaque odio modi reiciendis quod fugiat adipisci, dolorum quaerat doloribus provident eveniet quibusdam dicta?Unde libero totam cum, quidem sit a placeat iusto deleniti.Repellendus quo doloribus, quia dicta quidem similique veritatis sunt velit praesentium quod, porro rerum accusantium, voluptates voluptatem sunt odit accusamus at quibusdam assumenda repellendus eos atque, dignissimos voluptatem eaque atque fugiat sunt quos nam

esse neque aliquid reprehenderit?A amet saepe dolores architecto, magnam quidem eum corrupti, error cumque omnis dolore praesentium, et dicta voluptatum dignissimos quam?