Calibrating Uncertainty Estimates for Machine Learning Models

Yuchen Niu
Imperial College London
yuchen.niu21@imperial.ac.uk

Abstract

Recent advances in machine learning have dramatically improved the ability of neural networks to solve real-world problems. Machine learning methods are also more frequently deployed in critical scenarios. This requires that machine learning models need to be both accurate and reliable. So, calibrating a model to create a rigorous confidence set or estimates that represent the actual correctness likelihood is essential. In this report, we review the theoretical and practical concepts of uncertainty estimation and calibration. The history of popular calibration models and their application are also covered, which will help understand concepts more intuitively. We conduct experiments on the uncertainty estimation of neural networks before and after applying various post-processing calibration methods. Empirical results show that modern neural networks are often overconfident and poorly calibrated. Temperature scaling is straightforward and surprisingly efficient at calibrating confidence for classification models. Conformal prediction is a non-parametric distribution-free calibration method for regression models. However, we notice that distribution shifts and biases can easily influence the confidence interval for predictions from conformal inference. Ensembling can be an implicit calibration method that can mitigate the overconfidence of neural networks and provide stable predictions.

Contents

1 Introduction

The neural network has gained popularity in the community of machine learning and it is now an entrusted method for solving complex real-world problems due to its ability of implicit pattern discovery and universal function approximation (1). We can categorise neural networks into classifiers and regressors. Assume we have a set of data points $S = \{x_i, y_i\}, i = 1, 2, 3, ..., N$, where x_i is the input and y_i is the target class. Classifiers output a conditional probability distribution of target categories given the data and pick the one with the largest probability as final predictions:

$$\hat{y_i} = \arg\max_{y_i} P(y_i|x_i)$$

The classification networks are usually optimised by negative log-likelihood as a proxy method for maximum likelihood estimation to increase the conditional probability of respective target classes. For regressors, the target y_i becomes a real value and networks approximate a function that maps x to y:

$$\hat{y_i} = f(x_i)$$

Regressors are optimised by minimising the distance between $\hat{y_i}$ and y_i. Commonly used loss functions are Laplace (L1) loss and mean squared (L2) loss. The L1 loss measures the absolute difference

$$l_1 = \frac{1}{N}\sum_{i=1}^{N} |\hat{y_i} - y_i|$$

that yields the medians of targets, whereas L2 loss calculates the mean squared norm to estimate the means of targets

$$l_2 = \frac{1}{N}\sum_{i=1}^{N} ||\hat{y} - y||_2^2$$

However, the neural network is an old idea that has gone in and out of fashion since the late 20th century. Hardware innovations throughout the decades and the breakthrough from Deep Brief Network (2) made its application possible and rapidly developed in the last decade. Convolutional Neural Network (CNN) (3) has been widely used for classification tasks (4, 5). The unique feature of the Recurrent Neural Network (RNN) (6) makes it become an ideal model for language and speech processing. More recent, Transformer (7) has shown the tendency to unify the deep learning community since it achieves state-of-the-art results on many complex tasks (8, 9).

It is a common desire for human kinds to make a liable prediction for an uncertain future. As the neural network is usually treated as a black-box method, the trustworthiness of a neural network has received attention in some critical scenarios. Consider an autonomous vehicle that uses a neural network to detect obstructions. If the network predicts an object as not an obstruction with a probability close to 50%, the decision of whether break should rely more on the other information or sensors. Automated disease diagnosis in health care is another scenario in which critical decisions must be made with confidence. A doctor needs to intervene in the outcome from a neural network to ensure the patient's diagnosis is correct. So, a well-calibrated confidence estimation is often required in addition to the prediction from neural networks.

In the project, we present uncertainty estimation and calibration on neural networks in classification and regression, respectively. The structure of this report is listed below:

- Section 2 reviews statistical concepts of probabilistic forecasting and calibration concepts for machine learning.
- Section 3 explains in detail the design of the experiments and compares the performance of calibration methods on the neural networks[1].
- Section 4 concludes the achievements of the project and proposes suggestions for future work.

[1]The code is available at https://github.com/NiuJ1ao/Calibrating-Uncertainty-Estimates

2 Background

2.1 Probabilistic Forecasting

Probabilistic forecasting is a popular paradigm that predicts the uncertainty of future events. The general goal of probabilistic forecasting is maximising the sharpness of the predictive distribution subject to calibration (10). Ideally, we would like the forecast matches the realisation. For the instance of biased coin tossing, if there is 90% of chance that a toss is a head, the forecaster supposes to predict a Bernoulli distribution with parameter $p = 90\%$ and concentrate on the two ends of the distribution rather than a point estimate of being head.

The predictive distribution from a forecaster needs to be assessed on the basis of prediction spaces, which is a joint distribution $P(F, Y)$ of the probabilistic forecast F and the event that materialise Y, where F is a CDF-valued random quantity that utilises some information set $\mathcal{A}$ (11). Gneiting et al. (10) stated that the F is ideal if $F = \mathcal{L}(Y|\mathcal{A})$, where $\mathcal{L}$ denotes a probability distribution. So, an ideal forecaster needs to make the best use of the information given by nature.

To evaluate a forecaster, the probability integral transform (PIT) $Z_F = F(Y)$, where Z_F is the random variable, has formed a cornerstone of forecast evaluation (12). If quantity F is ideal and continuous, then Z_F has a uniform distribution. However, Hamill (13) aimed to show that the uniformity of the PIT values is only a necessary condition for a probabilistic forecast to be ideal. To evaluate probability forecast properly, Gneiting et al. (14) proposed a diagnostic approach to assess the predictive performance based on calibration and sharpness from the general goal of the probabilistic forecast.

2.1.1 Calibration

Calibration refers to the statistical consistency between predictive distribution and realisation and is a joint property of the forecasts and the events that materialise regardless of their origins (10). Given a given sequence of true probability distributions $(G_t)_{t=1,2...}$ from the nature, there are three modes of calibration (14):

1. The sequence of forecast F_t is probabilistically calibrated if

$$\frac{1}{T}\sum_{t=1}^{T} G_t \circ F_t^{-1}(p) \to p \quad \forall p \in (0, 1)$$

2. The sequence of forecast F_t is exceedance calibrated if

$$\frac{1}{T}\sum_{t=1}^{T} G_t^{-1} \circ F_t(y) \to y \quad \forall y \in \mathbb{R}$$

3. The sequence of forecast F_t is marginally calibrated if

$$\bar{G}(y) = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} G_t(y) \quad \texttt{and} \quad \bar{F}(y) = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} F_t(y)$$

   exist and equal each other for all $y \in \mathbb{R}$ and if the common limit distribution places all mass on finite values.

The probabilistic calibration is equivalent to the uniformity of PIT values and the marginal calibration can be written in a simpler form $\mathbb{E}[F(y)] = P(Y \leq y)$ (10). Both marginal calibration and probabilistic calibration highlight the role of ideal forecasts relative to some information set (11). Since exceedance calibration was not discussed in the literature, we will focus on assessing probabilistic calibration and marginal calibration.

Examining histograms of the PIT values is now an approach for assessing the probabilistic calibration of a forecaster instead of assessing the predictive distribution. A probabilistically

calibrated forecast should provide a uniform histogram of PIT values. Hump-shaped histograms indicate overdispersed predictive distribution, whereas U-shaped histograms indicate underdispersed predictive distributions (12, 13, 14).

For assessing marginal calibration, Gneiting et al. (14) proposed a visual inspection by plotting the difference of the two CDFs $\bar{F}_T(y)$ and $\bar{G}_T(y)$. By the definition of marginal calibration, minor fluctuations around zero only are expected.

2.1.2 Sharpness

Sharpness refers to the concentration of the predictive distribution and is a property of the forecasts only (10). The concentration of a predictive distribution is basically equivalent to the prediction interval. So, the mean widths of these intervals from forecasts should be as short as possible, subject to the empirical coverage being at a nominal level (10). Therefore, the ideal forecaster forms the lower bound of the sharpness for any sufficient calibrated forecaster (14).

2.1.3 Scoring Rules

A scoring rule $S : \mathcal{F} \times \mathbb{R} \to \bar{\mathbb{R}}$, where $\bar{\mathbb{R}}$ is the extended real line $[-\infty, \infty]$, is a function that assigns numerical values to forecasts that represents predictive performance and addresses calibration and sharpness simultaneously (14). It provides a piece of summary information for the evaluation of probability forecaster. From the perspective of convexity, scoring rules are usually considered penalties that we would like to minimise. If there is a best belief G and a probabilistic forecast F, where $G, F \in \mathcal{F}$ and $\mathcal{F}$ is a convex class of probability distribution in $\mathbb{R}$, a event x is materialised from G, then the penalty for the forecast is $S(F, x) \in \bar{\mathbb{R}}$. So, $S(F, G)$ can be written as the expected value of $S(F, \cdot)$ under G (15). To optimise the estimation of a predictive distribution, the scoring function needs to be proper, and the score of an ideal distribution should be the smallest: $S(G, G) \leq S(F, G)$. The scoring rule is strictly proper if the equality only holds when $F = G$ (15). Practically, minimising tailored strictly scoring functions is required in solving estimation problems.

2.2 Uncertainty

We shall notice that a predictor, like a neural network, is usually trained as a point estimator instead of a distribution estimator. Some real-world applications, like medical diagnosis and automated vehicles, have low toleration of mistakes. Models for these tasks must not only be accurate but also indicate how likely outcomes are reliable. Therefore, the uncertainties of a model should be estimated jointly with predictions before applying calibration. The confidence of a classifier is simply the conditional probability of the target class from the models. If a model is certain, the conditional probability of the respective target class should be high, and those of other classes should be low. For regressors, a range of estimates needs to be evaluated to determine how likely a sample lies in the range. Confidence interval (CI) and quantile regression are two notions commonly used for uncertainty estimation.

2.2.1 Confidence Interval

Confidence interval is a common measurement of uncertainty. It constructs a symmetric band around estimated mean $\hat{x}$ with the width of the multiplication of the standard deviation σ:

$$CI = \hat{x} \pm z\frac{\sigma}{\sqrt{n}}$$

where z is a scale of confidence level and n is the sample number. The most common confidence interval is 95%. Since it is reasonable to assume that a model with L2 loss estimates the mean of a normal distribution: $\mathbb{E}_{\mathcal{N}(\mu,\sigma^2)}[y_i|x_i]$, 95% confidence level is roughly equivalent to $\hat{y} \pm 2\sigma$. However, a drawback of a mean squared estimator is that it is sensitive to outliers. The extreme sensitivity to contamination by a small number of outliers makes the model poorly estimated in non-Gaussian distributions (16).

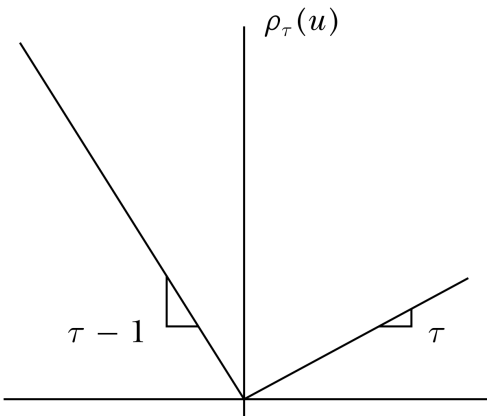


Figure 1: ρ function for quantile regression (18)

2.2.2 Quantile Regression

Quantile regression is a regression method that makes no assumption on the global distribution and enables us to estimate quantiles without modifying model architecture (17). It replaces the loss function by quantile loss

$$\sum_{i=1}^{N} \rho_\tau(y_i - f(x_i, \theta))$$

where τ is a quantile and the function $\rho_\tau(\cdot)$ is a tilted absolute value function as shown in Figure 1. Let τ equal to 0.5 for now and the quantile loss is equivalent to the L1 loss. As minimising the sum of absolute residuals provides the median of samples, the symmetry of the absolute function implies that the number of positive residuals and negative residuals must be equal to minimise the loss function:

$$\frac{\partial l_1}{\partial \hat{y}} = \frac{1}{N} \sum_{i=1}^{N} sgn(\hat{y} - y_i) = 0$$

where $\hat{y}$ is an estimation and $y_{i=1,...,N}$ are samples. Other quantiles follow a similar intuition by replacing the symmetrical absolute function with an asymmetrically weighted absolute function, i.e., quantile loss, which assigns different weights to positive and negative residuals to estimate the expected quantile.

The optimality condition for quantile regression is referred to as Subgradient condition (18). Since the objective function of quantile regression is piece-wise linear and continuous, it is differentiable except at the points where residuals are zero. By the first-order necessary condition, the directional derivatives at a point are non-negatives if the point minimises the function. During the optimisation, the quantile regression identifies p-element subsets that best describe the conditional quantile function. These subsets are also referred to as basic solutions in terms of linear programming. Let $h \in \mathcal{H}$ index basis solutions of the first n integers, $\mathcal{N} = \{1, 2, ..., n\}$. Elements $h \in \mathcal{H}$ and its relative complement $\bar{h} = \mathcal{N} - h$ serve to partition y and X to get sub-vectors and sub-matrices, respectively. Koenker et al. (16) stated that the solution of quantile regression problems has a basic form $b(h) = X(h)^{-1}y(h)$ and there exists a solution in this form if and only if, for some $h \in \mathcal{H}$,

$$-\tau 1_p \leq \xi(h) \leq (1 - \tau)1_p$$

where $\xi(h)$ is derived from the directional derivative of the quantile objective function. Moreover, $b(h)$ is unique if and only if the inequalities are strict, otherwise, the solution set is a convex hull of all solutions in the basic form (18).

Regression quantiles have four equivariance properties (16). Let $\hat{\beta}(\tau; y, X)$ denotes the τth regression quantile based on observations (y, X),

1. Scale equivariance 1: $\hat{\beta}(\tau; \alpha y, X) = \alpha\hat{\beta}(\tau; y, X)$

2. Scale equivariance 2: $\hat{\beta}(\tau; -\alpha y, X) = -\alpha\hat{\beta}(1-\tau; y, X)$

3. Shift equivariance: $\hat{\beta}(\tau; y + X\gamma, X) = \hat{\beta}(\tau; y, X) + \gamma$

4. Equivariance to reparameterisation of design: $\hat{\beta}(\tau; y, XA) = A^{-1}\hat{\beta}(\tau; y, X)$

Such changes listed above will not affect the estimation of regression quantiles.

Apart from equivariance properties, the robustness of quantile regression can also be formally evaluated by the influence function, which offers a description of an estimator $\hat{\theta}$ at a distribution F is influenced by correspondingly contaminating distribution F_ϵ (19). The influence function of $\hat{\theta}$ at F can be expressed as (18)

$$IF_{\hat{\theta}(y,F)} = \lim_{\epsilon \to 0} \frac{\hat{\theta}(F_\epsilon) - \hat{\theta}(F)}{\epsilon}$$

For a quantile τ, the influence function is $IF_{\hat{\theta}(y,F)} = \text{sgn}(y - \hat{\theta}(F))/f(F^{-1}(\tau))$ as $\hat{\theta}(F_\epsilon) = F_\epsilon^{-1}(\tau)$. When the distribution F is extended as the joint distribution (x, y), the influence function becomes

$$IF_{\hat{\beta}_F(\tau)}((y, x), F) = Q^{-1}\ x\ \text{sgn}(y - x^T\hat{\beta}_F(\tau))$$

where $Q = \int xx^T f(x^T\hat{\beta}_F(x))dG(x)$ and $dF = dG(x)f(y|x)dy$ (18). From above influence function and subgradient condition, y only appears in the $\text{sgn}(\cdot)$ function and the affection of a point to the subgraident does not rely on y_i as long as $\text{sgn}(y - x^T\hat{\beta}_F(\tau))$ does not change. So, y_i can be freely moved up and down as long as it stays on the same side of the fitted plane. Recall the optimality of quantile regression, the estimator selects p-element subsets. The robustness clarifies that other samples are never ignored, rather they participate equally in determining representations (18).

2.3 Calibration Methods

Post-hoc calibration is a common calibration approach because calibrators are trained independently on a small validation dataset which is computationally cheap and avoids unintentional bias from training data. Also, the inference model remains the same during calibration. We will discuss several post-hoc calibration methods for classification and regression. Most of the calibration methods are originally proposed for binary classification tasks. Recent work has adapted some methods for multi-class classification.

2.3.1 Classification Calibration

Zadrozny et al. (20) introduced the histogram method to calibrate naive Bayesian classifiers. Naive Bayesian classifiers are based on the conditional independence assumption - features are mutually independent given the class. This strong independent assumption leads the classifier to inaccurate probability estimation (21). Histogram binning split predictions into mutually exclusive bins based on their scores. The bin boundaries are usually equal length intervals or equal to the number of samples in the bin. Then, the calibrated estimation is obtained as the fraction of samples in the bin that belongs to the target class. The variance of the binned probability estimates is correlated to the number of bins. So, the number of bins must be small to reduce the variance of predictions by increasing the number of samples whose positive and negative classes are averaged inside each bin (20).

Zadrozny et al. (22) then proposed another non-parametric calibration method based on Isotonic regression (23). It is a more general method that calibrates decision trees, naive Bayesian and SVMs. It is also an intermediary of the sigmoid fitting and binning process. This method restricts the mapping function from scores to calibrated probabilities to be isotonic (i.e. non-decreasing). Pair-adjacent violator (PAV) (24) is a common approach to finding a stepwise constant Isotonic function that best fits the data by optimising mean squared error. The details of the PAV algorithm are shown in Table 1

Algorithm 1 PAV algorithm (24)

Input: (p_i, y_i) set, where p_i is sorted predictions and y_i is targets.
  $m_{i,i} = y_i$ ▷ m is an Isotonic function
  while $\exists i$ *s.t.* $m_{k,i-1} \geq m_{i,l}$ **do**
    $w_{k,l} = w_{k,i-1} + w_{i,l}$
    $m_{k,l} = (w_{k,i-1} m_{k,i-1} + w_{i,l} m_{i,l})/w_{k,l}$
    $m_{k,i-1} = m_{k,l}$
  end while
Output: Stepwise constant function $m(p) = m_{i,j}$ for $p_i < p \leq p_j$, where $m = \arg\min_z \sum (y_i - z(p_i))^2$

Platt (25) proposed a parametric calibration method that maps SVM outputs to posterior probabilities. For the binary classification task, the prior $P(y = 1|f)$ is assumed to be monotonic in f since the SVM is trained to separate most positive samples away from negative ones. However, the strong prior is violated if we fit probabilities to the outputs of an SVM by Bayes' rule, which can be written as an analytic function of f:

$$P(y = 1|f) = \frac{1}{1 + \exp(af^2 + bf + c)}$$

which is non-monotonic, and the assumption of Gaussian class-conditional density is violated (25). To address these problems, Platt scaling estimates the posterior $P(y = 1|f)$ directly instead of the class conditional density $P(f|y)$. As suggested by two exponentials of the Bayes' rule, a parametric sigmoid model should be applied on the top of a SVM:

$$P(y = 1|f) = \frac{1}{1 + \exp(af + b)}$$

where a and b are model parameters trained discriminatively and the model assumes that the output of the SVM is proportional to the log probability of a positive sample. The sigmoid model is optimised by the negative log-likelihood loss (NLL) on the validation dataset.

Recent work (26) have extended Platt scaling to multiclass models on the neural networks. The simplest extension of Platt scaling is Temperature scaling. Instead of linearly transforming logits z_i, they are divided by a global parameter T before passing through a softmax layer:

$$\hat{q}_i = \max \sigma_{SM}(z_i/T)^{(k)}$$

The temperature softens the probability of each class, which is equivalent to increasing the outputs' entropy. Since it does not change the maximum outcomes, the predictive class will remain the same. Matrix scaling is another implementation that follows a similar procedure as Platt scaling. The model takes the logits z_i and applies a linear transformation to them:

$$\hat{q}_i = \max \sigma_{SM}(W z_i + b)^{(k)}$$

where W is a weight matrix and b is a bias. If W is restricted to a diagonal matrix, the method is called vector scaling instead.

2.3.2 Conformal Prediction

Conformal prediction (27, 28) is an emerging inference method that quantifies model uncertainty without distribution assumption. Let $\{(x_i, y_i)\}$ for $i = 1, 2, ..., n$ be a training set of i.i.d. samples of a random variable. For another data sample (x_{test}, y_{test}) from the same i.i.d. distribution, conformal prediction estimates a confidence set $\mathcal{T}(x_{test})$ based on the training samples. The test sample must be in the set with a user-chosen error rate α for a well-calibrated regressor:

$$P(y_{test} \in \mathcal{T}(x_{test})) \geq 1 - \alpha$$

The general approach of conformal prediction follows (29):

1. Tailor a score function $s(x, y) \in \mathbb{R}$ that represents the distance between the sample and predictive boundaries.
2. Retrieve uncertainty estimates from a model.
3. Compute the $\lceil \frac{(n+1)(1-\alpha)}{n} \rceil$ quantile $\hat{q}$ of calibrated scores $s_i = s(x_i, y_i)$ for $i = 1, 2, ..., n$
4. Construct the confidence set for x_{test}: $\mathcal{T}(x_{test}) = \{y : s(x_{test}, y) \leq \hat{q}\}$

The choice of the score function is important engineering work. The score function plays a major role in the usefulness of the prediction set since the score function encodes the model performance that easy inputs should construct a smaller set and hard inputs should construct a bigger set (29).

The original conformal prediction method is computationally intensive because the estimator needs to be retrained and recompute the quantile of calibration scores every time a new sample is encountered. A cheaper and more common approach is split conformal prediction (28). The samples are split as training-test split fashion. One part is used to fit the estimator, while another is used to compute calibration scores. Then, the predictive interval for each sample is adjusted by the quantile of scores $\hat{q}$. Intuitively, the split conformal prediction expands or shrinks the prediction bands estimated from training samples by $\hat{q}$ to achieve expected coverage.

The full conformal prediction and split conformal prediction tend to generate predictive interval $\mathcal{T}(x)$ with roughly constant width over $x \in \mathbb{R}^d$. In some scenarios, the prediction bands should adjust correspondingly as x varies. Lei et al. (28) introduced locally-weighted conformal inference that scales the score function by an estimated error spread. For example, the score function is defined as the absolute residual of true value y and estimated value $\hat{f}(x)$, the fitted residuals should be scaled by an estimated conditional mean absolute deviation $\hat{\rho}_y(x)$ (MAD) of $(Y - f(X))|X$:

$$s_i = \frac{|y_i - \hat{f}(x_i)|}{\hat{\rho}_y(x_i)}$$

where $\hat{\rho}_y(x_i)$ can be estimated jointly with $\hat{f}(x_i)$ or the desired value $\hat{f}(x_i)$ is estimated first, then estimates error spread $\hat{\rho}_y(x)$ using the absolute residuals.

Previous conformal methods assume that all samples are drawn from the same distribution. However, the test samples may come from a different distribution than training samples in real-world applications. Covariate shift is a type of distribution shift in which the distribution P_X shifts to P'_X but the relation $P_{Y|X}$ retains. Formally,

$$(x_i, y_i) \overset{\text{i.i.d.}}{\sim} P = P_X \times P_{Y|X},\ i = 1, ..., n$$
$$(x_{test}, y_{test}) \overset{\text{i.i.d.}}{\sim} P' = P'_X \times P_{Y|X}$$

Tibshirani et al. (30) addressed covariate shift by weighted conformal prediction. In this case, we consider the weight as the likelihood ratio that a sample is from the new distribution P'_X than the old distribution P_X:

$$p_i^w(x) = \frac{w(x_i)}{\sum_{j=1}^n w(x_j) + w(x)},\ i = 1, ..., n$$
$$p_{test}^w(x) = \frac{w(x)}{\sum_{j=1}^n w(x_j) + w(x)}$$

where $w(x_i) = \mathbf{d}P'_X(x_i)/\mathbf{d}P_X(x_i)$. Then, we can compute score quantile of a reweighted distribution as a function of x:

$$\hat{q}(x) = \inf\{s_j : \sum_{i=1}^{j} p_i^w(x) \mathbb{1}\{s_i \leq s_j\} \geq 1 - \alpha\}$$

The quantile is no longer the same as the quantiles from previous conformal inferences. If the covariate shift makes easier values more likely by putting more weight on small scores, the quantile should be reduced (29). Therefore, the choice of quantile is an essential step under covariate shift.

2.4 Calibration evaluation

So far, we have discussed popular methods that can calibrate neural networks. However, whether a model is calibrated correctly remains unknown. In this section, we will detail three evaluation metrics for calibration.

2.4.1 Reliable Diagrams

Reliable diagrams (31, 32) are a visual interpretation of model calibration. In the context of binary classification, these diagrams partition predictions into bins with equal intervals and calculate the average predicted value for each bin. Then, the average expected values are plotted against the fraction of positive labels in the bins. The reliable diagram of a well-calibrated classifier should form a diagonal line.

For multi-class classification, the reliable diagrams plot the accuracies with respect to the confidence in the bins, where accuracy is the portion of correctly predicted samples in a bin and confidence is the average probability of correctness in a bin (e.g., Figure 3 top) (26). Since the probability of a class from a well-calibrated classifier should reflect its ground truth correctness likelihood, the accuracy and the average confidence in a bin must be equal for a well-calibrated classifier. For instance, there are 100 samples in a bin, and the confidence of every sample is 90%. The number of correctly predicted samples must be 90.

2.4.2 Calibration Error

Since using a scalar statistic to evaluate calibration is more convenient, two simple calibration errors are proposed that follow the intuition of reliable diagrams. These metrics are Expected Calibration Error (ECE) and Maximum Calibration Error (MCE) (33, 26). The first step of both calibration metrics is as same as the reliable diagrams, which partition predictions into equally-width bins. Then, they evaluate the gap between accuracy and the average confidence in each bin (or the average prediction value and the fraction of positive labels in a bin). The ECE computes the weighted average of the gaps in the bins:

$$ECE = \sum_{i=1}^{M} \frac{n_i}{N} |acc_i - conf_i|$$

where N is the total number of samples, M is the number of bins and in the bin i, n_i is the number of samples, acc_i and $conf_i$ denote the accuracy and average confidence. Similar to ECE, MCE takes the largest calibration gap:

$$MCE = \max_{i \in \{1,\dots,M\}} |acc_i - conf_i|$$

For a well-calibrated classifier, MCE and ECE are both zero.

2.4.3 Correct Coverage

Previous evaluation methods are for classification tasks. Recall the goal of conformal prediction, it guarantees that a sample will fall into an estimated conformal set within a user-defined error rate. Therefore, the coverage of samples can be implemented as an evaluation method for regressors:

$$Coverage = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}\{y_i \in \mathcal{T}(x_i)\}$$

where N is the number of samples. If one chooses a 10% error rate, the coverage of the conformal band should be 90% on the total samples for a well-calibrated regressor.

Dataset	Training	Validation	Test
CIFAR-10	40000	10000	10000
SVHN	73257	20000	26032

Table 1: Statistics of classification datasets.

Dataset	Input Dim.	Training	Validation	Test
Solar	1	203	88	100
Housing	12	247	106	152
Concrete	7	504	217	309
Energy	8	375	162	231
Power	3	4687	2010	2871
Red wine	10	782	336	480
Yacht	5	149	65	92

Table 2: Statistics of regression datasets.

3 Experiment

3.1 Datasets

We choose CIFAR-10 (34) and SVHN (35) as classification datasets. The number of samples in the training, validation and test set is shown in Table 1. For regression tasks, time-series is a suitable task to evaluate uncertainty. We conduct experiments on the solar irradiance dataset (36), in which all targets are standardized, and five interpolated segments of data are used as test samples. This one-dimensional dataset can help us visualise the fitted curves and estimated uncertainty intervals between $x = -230$ and $x = 230$ (Figure 7 & 8). Moreover, 6 UCI datasets (37) are used for multivariate regression. Among those datasets, we randomly select 30% of total samples as test data and then retrieve 30% from the training data as validation data. The details of the statistics of the UCI dataset are shown in Table 2.

3.2 Model Architectures

We use Torchvision built-in ResNet-101 (4) as the uncalibrated classifier. Following prior work (26) and practical consideration, a ResNet is trained for 1000 epochs with a batch size of 256. The models are optimised for Cross-Entropy loss by stochastic gradient descent with 0.1 learning rate and 0.9 momentum. To avoid overfitting on the training set, weight decay is set to 0.0001, and the best model is selected based on the best accuracy on the validation set.

Matrix scaling and temperature scaling are implemented as calibration models. Matrix scaling is just an affine layer (10-10) followed by a softmax layer. The affine layer is trained for 250 epochs. Temperature scaling only needs to train for 100 epochs since it has only one trainable parameter. All calibration models are trained on the 70% of the validation set. The rest 30% is the validation set of validation set that selects the best calibrator based on the lowest ECE. The optimiser is Adam (38) with a learning rate of 0.01 for fast convergence, and Cross-Entropy loss is the optimisation criterion. During the training of calibration models, the original ResNets are frozen.

For regression, we use multi-layer perceptrons (MLP) to fit curves. In the experiments, we implement two different architectures (Figure 2). One assumes that the data are from Gaussian distribution and estimates means and variances by optimising Pytorch build-in Gaussian negative log-likelihood loss. We refer to this method as linear regression (L.R.). Another one estimates three quantiles with a user-defined error rate α by optimising respective quantile losses, which is referred to as quantile regression (Q.R.). Since the solar irradiance dataset is only one dimensional, regressors have 40 hidden sizes. They are trained for 5000 epochs with Adam optimiser of 0.001 learning rate and $\frac{0.1}{\sqrt{N}}$ weight decay, where N is the data size. For UCI datasets, regressors have 200 hidden sizes due to larger input dimensions and are trained for 1000 epochs

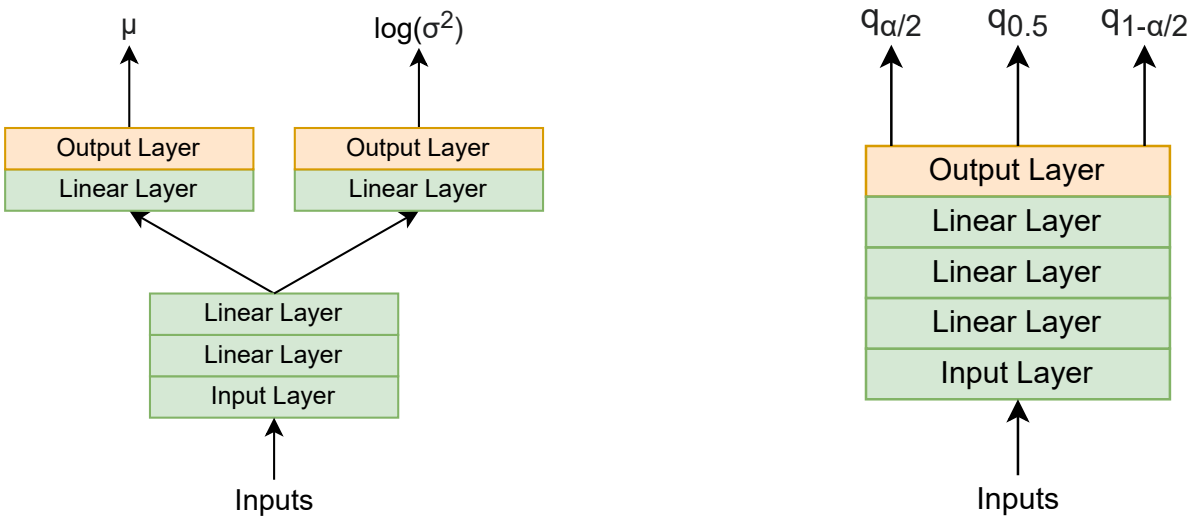


Figure 2: Model architectures of multi-layer perceptrons

with the same optimiser configuration. Note that the green blocks in Figure 2 also consist of ReLU activation layers and dropout layers with a 0.01 dropout rate.

Not only the calibration performance of a single model is evaluated, but we also assess the performance of ensemble models. We train 5 models on each dataset with different random seeds. To retrieve ensemble outputs, the logits from models can be either averaged or weighted by the models' accuracies (or other heuristic weights). Then, the calibration methods are applied on top of ensemble logits. So, the calibrator is aware of the uncertainty of all models. In this project, we uniformly average logits since the performance of 5 individual models are similar.

3.3 Evaluation

3.3.1 Classification

We characterise the performance of classifiers as measured by accuracy, ECE, MCE and visualised by reliable diagrams before and after applying calibration approaches. The number of bins is 10 for three calibration evaluation methods. Since the reliable diagram cannot provide information about the confidence level of a model, histograms of confidence are also drawn.

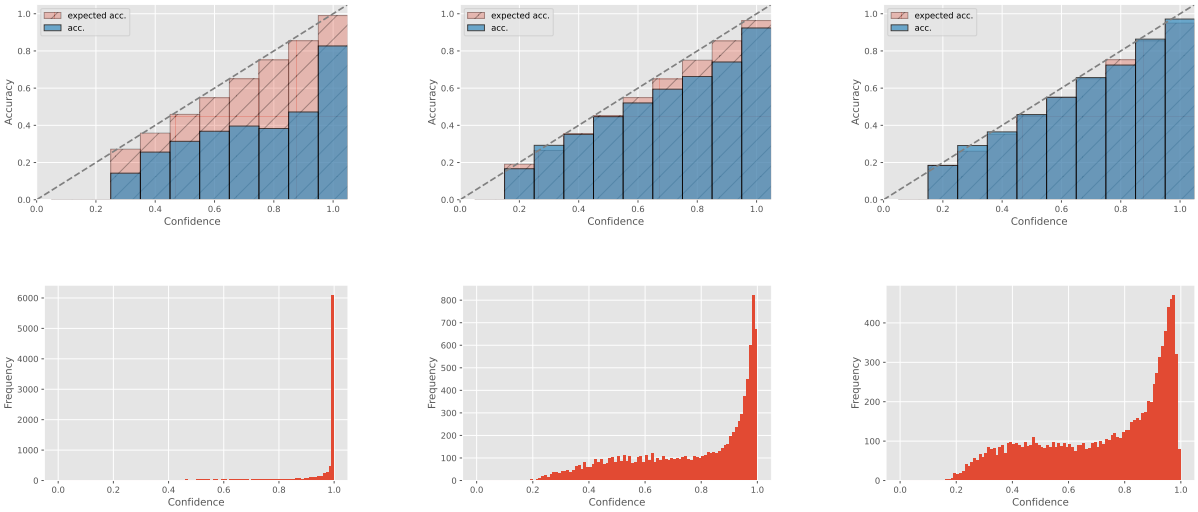
The original ResNet-101 achieves more than 70% accuracy on the CIFAR-10 dataset but experiences some degree of miscalibration, where the ECE is near 20%, and the MCE is around 38% (Table 3). As shown in Figure 3a top, the accuracy in each bin is significantly lower than the average confidence in the bin, especially as the confidence is lower. From Figure 3a bottom, we observe that the original ResNet is highly confident about its predictions since most of the predictive probabilities are close to 1.

Matrix scaling and temperature scaling have significantly improved ECE and MCE (Table 3). From Figure 3b and 3c, more confidences spread at the lower regions. The modes of frequencies of confidences also shift to the left. Calibration methods make the model less assured for better calibration errors. However, matrix scaling reduces the accuracy to 71.7% on the CIFAR-10. Temperature scaling does not change the prediction and achieves the best calibration performance on the CIFAR-10.

The ResNet-101 on the SVHN has much better performance in both prediction accuracy and calibration errors than on the CIFAR-10. The huge gap at 0.2 confidence is caused by outliers (Figure 4 top) since it is possible that the model has a low belief in an outlier but still predicts correctly. Both approaches mitigate the miscalibration of the original model. Table 3 shows that two methods have similar ECE on the SVHN and the temperature scaling achieves the best MCE on the SVHN. The accuracy of matrix scaling is negligently higher than the original

Dataset	Model	Acc.	ECE	MCE
CIFAR-10	Uncalibrated	73.32%	18.96%	38.28%
	Matrix Scaling	71.7%	4.95%	11.3%
	Temp. Scaling	73.32%	1.54%	3.07%
SVHN	Uncalibrated	92.5%	5.1%	80.6%
	Matrix Scaling	92.62%	2.15%	30.4%
	Temp. Scaling	92.5%	2.31%	22.03%

Table 3: Calibration performance of single ResNet-101.



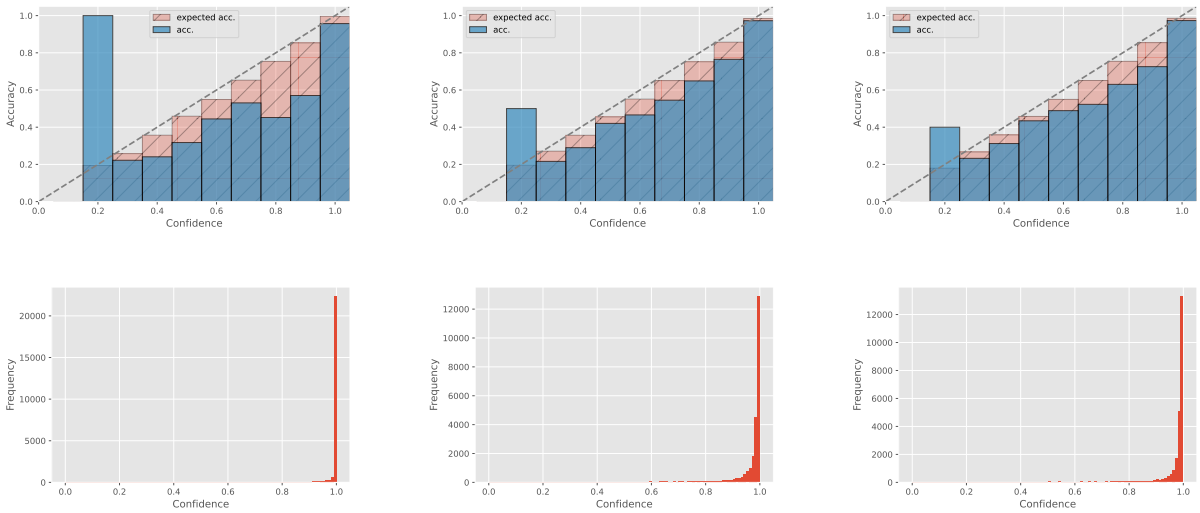
(a) Uncalibrated (b) Matrix Scaling (c) Temp. Scaling

Figure 3: Reliable diagrams and confidence histograms for single ResNet-101 on CIFAR-10.

model. After both calibration approaches, models are still confident (Figure 4 bottom). So, SVHN is relatively easy for ResNet-101 since it can obtain high accuracy and low errors without post-processing.

Table 4 demonstrates the accuracies and calibration errors from ensemble models. Some degree of calibration is observed from the outcomes of ensemble models compared with the single models (Table 3). The ECE of an ensemble ResNets decreases dramatically to 2.21% and 1.26% on the CIFAR-10 and SVHN, respectively. Likewise, the accuracy increases by 6% on the CIFAR-10 and 2% on the SVHN. Even though the classifiers still become less confident after calibration by inspecting the confidence histograms (Figure 5 & 6 bottom), the calibration errors are not necessarily lower. In this case, matrix scaling negatively affects accuracy, ECE, and MCE on the CIFAR-10, which almost tripled calibration errors. But, matrix scaling on SVHN illustrates opposite results that reduce the ECE to 0.67% and the MCE by 1%. ECEs on the CIFAR-10 and SVHN decrease by almost half for temperature scaling. However, MCEs become worse on both datasets. As shown in Figure 5c and Figure 6c top, the large gap between the average confidence and accuracy happens at 0.2 confidence, where the accuracy in the bin is much lower than the averaged confidence in the bin. The accuracies on two datasets for temperature scaling increase slightly because the logits from 5 models are averaged before applying softmax and the calibration methods use averaged logits as inputs.

From the experiments on two classification datasets, matrix scaling can calibrate classifiers. But, the accuracy is affected since the logits are linearly transformed, and we select the best model based on calibration performance instead of the accuracy. So, there is a risk that a calibration model overfits the validation set for lower calibration error. The temperature scaling achieves the best calibration performance in most cases on both datasets without disturbing the accuracy. Our results of temperature scaling verify the prior work (26). Ensembling can be treated as the third calibration method. The ensemble outputs are more stable and trustworthy. However, post-hoc calibration may not work on the ensemble models.



(a) Uncalibrated (b) Matrix Scaling (c) Temp. Scaling

Figure 4: Reliable diagrams and confidence histograms for single ResNet-101 on SVHN.

Dataset	Model	Acc.	ECE	MCE
CIFAR-10	Uncalibrated	79.27%	2.21%	7.3%
	Matrix Scaling	71.54%	6.15%	12.73%
	Temp. Scaling	79.97%	1.42%	10.92%
SVHN	Uncalibrated	94.71%	1.26%	8.62%
	Matrix Scaling	94.87%	0.67%	7.41%
	Temp. Scaling	94.91%	0.58%	18.3%

Table 4: Calibration performance of ensemble ResNet-101.

3.3.2 Regression

We evaluate the performance of split conformal prediction due to the benefit of being computationally cheap. In the experiments, we use standard deviations and quantiles as two heuristic notions of uncertainty. Following prior work (29), two scoring functions are implemented, respectively. For the confidence interval with 1 confidence level $CI = \mu \pm \sigma$, a locally-weighted scoring rule is formulated:

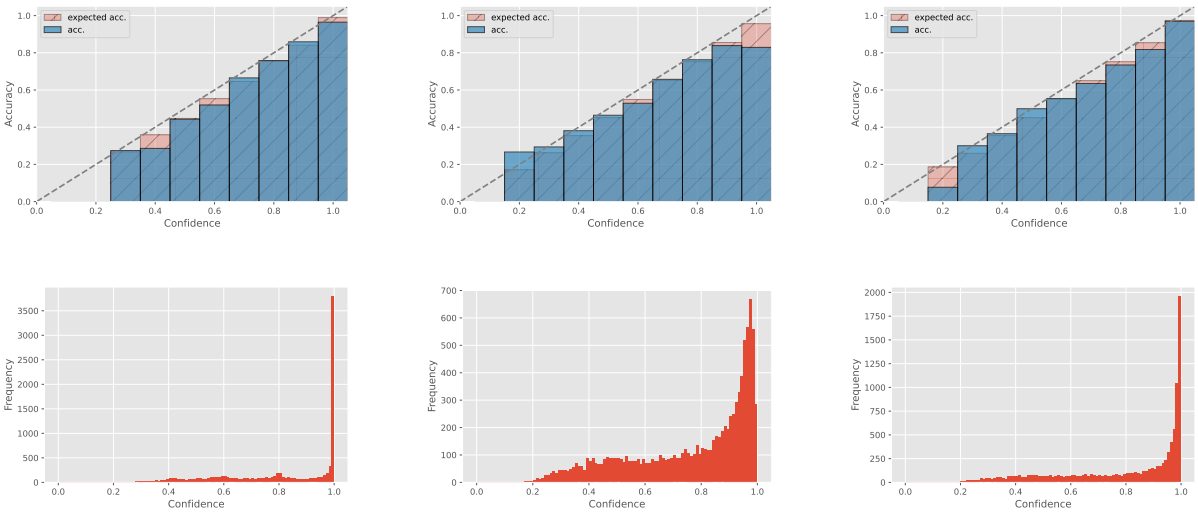
$$s(x, y) = \frac{|y - \hat{\mu}(x)|}{\hat{\sigma}(x)}$$

Then, the empirical quantile $\hat{q}$ of scores is used as a multiplicative factor to calibrate conformal band $\mathcal{T}(x) = [\hat{\mu}(x) \pm \hat{\sigma}(x)\hat{q}]$. Since we assume the data are from Gaussian distribution, the ideal coverage of this confidence interval is around 68%. For quantile regression, the scoring function is the project distance from the true value y onto the lower quantile $\hat{\tau}_{\alpha/2}(x)$ and upper quantile $\hat{\tau}_{1-\alpha/2}(x)$:

$$s(x, y) = \max\{\hat{\tau}_{\alpha/2}(x) - y, y - \hat{\tau}_{1-\alpha/2}(x)\}$$

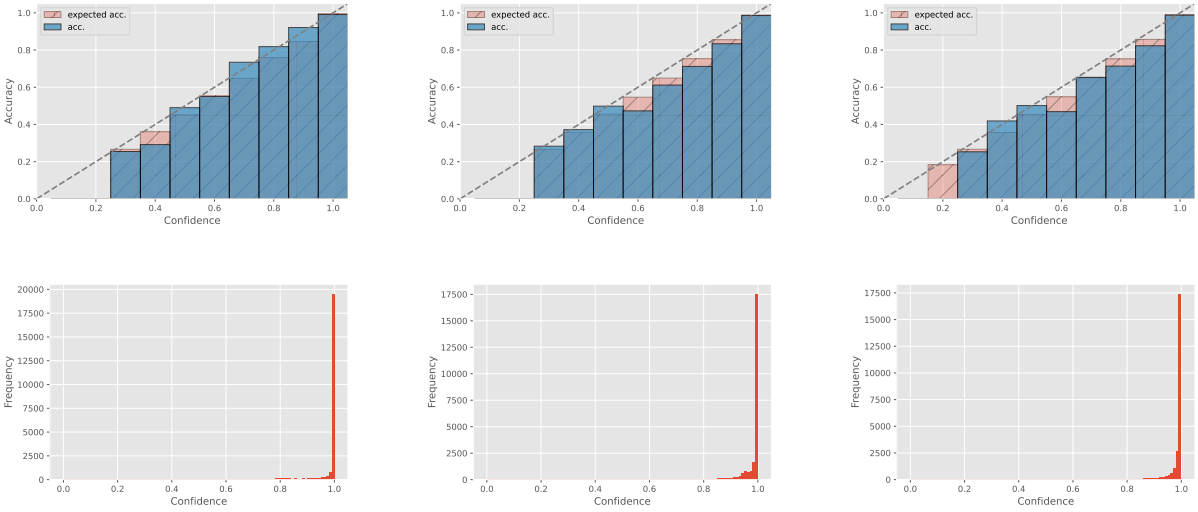
After computing $\hat{q}$ as usual, the conformal band simply grows or shrinks by $\hat{q}$ on two sides: $\mathcal{T}(x) = [\hat{\tau}_{\alpha/2}(x) - \hat{q}, \hat{\tau}_{1-\alpha/2}(x) + \hat{q}]$. The sharpness of a conformal band is also evaluated as the variance of the gap between conformal boundaries, which is $\mathbb{V}[\hat{q}\sigma^2]$ for linear regression ($\hat{q}$ is 1 before calibration) and the variance of distances between two quantiles for quantile regression.

The conformal prediction does not necessarily adjust the boundaries towards the correct coverage. The quantile of scores $\hat{q}$, the only variable of moving edges, solely depends on the scores of validation data. So, the width of conformal bonds is adjusted to cover the ideal portion of validation data. In the solar irradiance dataset, the conformal prediction increases the coverage on test data by 1% to 65% for the linear regression. It decreases the coverage to 75% for the quantile regression that the expected coverage should be around 90% (Table 5). However, the coverage on the validation set for the linear regression changes from 65.91% to



(a) Uncalibrated (b) Matrix Scaling (c) Temp. Scaling

Figure 5: Reliable diagrams and confidence histograms for ensemble ResNet-101 on CIFAR-10.



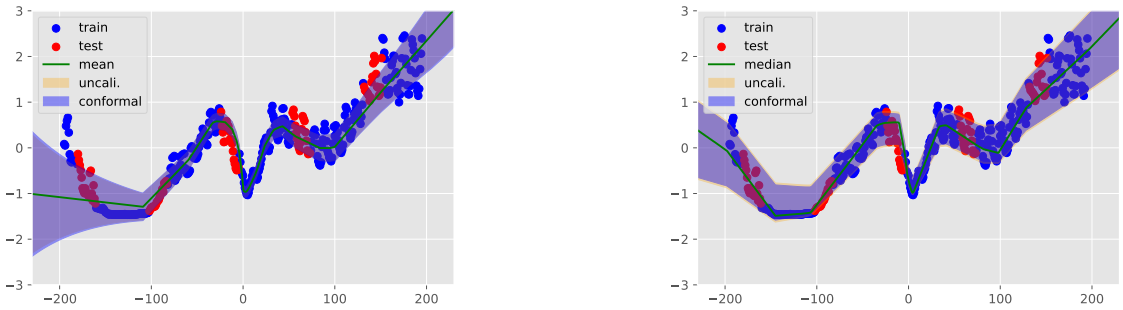
(a) Uncalibrated (b) Matrix Scaling (c) Temp. Scaling

Figure 6: Reliable diagrams and confidence histograms for ensemble ResNet-101 on SVHN.

69.32% and that for the quantile regression decreases from 94.32% to 92.04% as expected. This proves the implementation of conformal prediction is correct. As shown in Figure 7, the test data are interpolated segments. Therefore, we speculate that the distribution shift between calibration data and test data is why the coverage on test data does not change in the same direction as the coverage on validation data.

However, the conformal bands are adjusted in the right direction in most cases on the UCI datasets (Table 5). We believe that this is the validation data and test data are sampled randomly from datasets. So, they should have similar distribution and the distribution shift should not be a significant problem. Nevertheless, for linear regression, the coverage on the Red wine is over-calibrated from 67.5% to 73.75% after conformal prediction. The coverages of quantile regressors on the Energy and Yacht datasets are not updated correctly. We think overfitting causes problems since the conformal prediction can adapt the bias in the finite calibration set.

The sharpnesses of a single regressor are shown in Table 6. The sharpness of a linear regressor is changed according to how the conformal band is adjusted. Since the quantiles from quantile regression are added or subtracted by $\hat{q}$, the sharpnesses before and after conformal prediction are the same. The solar irradiance dataset has a relatively large sharpness since there is no data



(a) Linear Regression

(b) Quantile Regression

Figure 7: Conformal prediction of a single model on the solar irradiance dataset.

	L.R. (68%)		**Q.R.** (90%)	
Dataset	**Uncali.**	**Conformal**	**Uncali.**	**Conformal**
Solar	64%	65%	85%	75%
Housing	57.24%	63.16%	76.97%	84.21%
Concrete	55.35%	58.25%	77.67%	79.41%
Energy	55.41%	57.14%	85.71%	84.85%
Power	71.68%	70.11%	90.77%	90.8%
Red wine	67.5%	73.75%	83.75%	88.54%
Yacht	81.52%	75%	94.57%	94.57%

Table 5: Coverage of single multi-layer perceptrons on test sets.

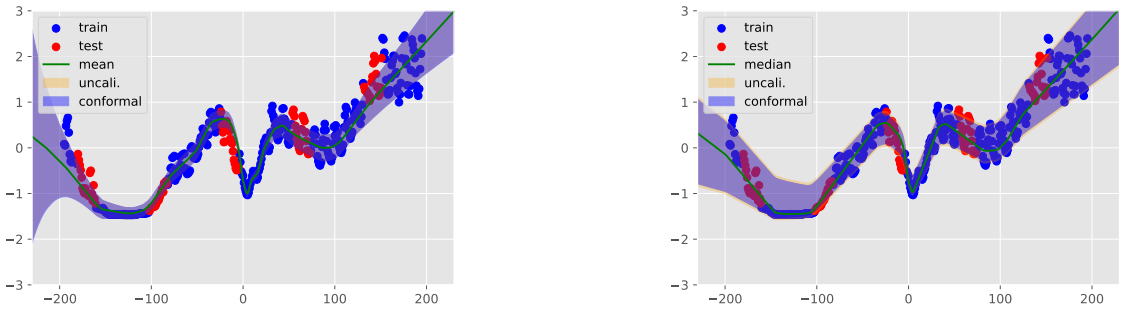
in $(-230, -200) \cup (200, 230)$. The model is very uncertain in these regions. However, we do not observe any relation between sharpness and coverage. Regardless of regression methods, the conformal prediction can adjust the intervals towards the expected coverage on either dataset with large sharpness, like Housing and Red wine, or a dataset with small sharpness, like Energy and Power. For those miscalibrations in the quantile regression, the Solar dataset has a much smaller sharpness than the Energy dataset. Therefore, the conformal prediction can work no matter how the data are sparse or noised.

The number of data also plays an important role. The Power dataset has the most data 2. An uncalibrated regressor on this dataset can obtain the coverage on the test set that is close to the expected value. After conducting conformal prediction, the coverage is still around the expected value. For small datasets, such as Solar and Yacht, the model is more easily interfered with by the outliers in the data. Therefore, the coverage of an uncalibrated model has a significant discrepancy with the expected value, and the degree of calibration of conformal prediction is limited or even wrong.

As shown in Figure 8, the curve fitting of ensemble models is better than these of a single model (Figure 7), especially on the left-hand sides of graphs. On the solar irradiance dataset, the uncalibrated ensemble linear regressor becomes over-confident that the coverage is only 56%

	L.R.		**Q.R.**
Dataset	**Uncali.**	**Conformal**	**Uncali. & Conformal**
Solar	0.0957	0.1073	0.1614
Housing	0.1227	0.1353	0.0252
Concrete	0.0826	0.0920	0.1635
Energy	0.0001	0.0001	0.0096
Power	0.0007	0.0007	0.0028
Red wine	0.0037	0.0049	0.3824
Yacht	0.1535	0.0908	0.032

Table 6: Sharpness of single multi-layer perceptrons.



(a) Linear Regression

(b) Quantile Regression

Figure 8: Conformal prediction of ensemble models on the solar irradiance dataset.

	L.R. (68%)		**Q.R.** (90%)	
Dataset	**Uncali.**	**Conformal**	**Uncali.**	**Conformal**
Solar	56%	56%	87%	83%
Housing	71.05%	68.42%	80.92%	86.18%
Concrete	60.19%	58.9%	78.96%	79.94%
Energy	66.67%	61.04%	86.15%	84.85%
Power	71.37%	70.53%	90.35%	90.53%
Red wine	69.17%	72.71%	86.25%	90%
Yacht	81.52%	78.26%	94.57%	95.65%

Table 7: Coverage of ensemble multi-layer perceptrons on test sets.

and its sharpness increases dramatically to 0.3723 (Table 6). The quantile ensemble regressor has similar characteristics as the single model. For the UCI datasets, Table 7 demonstrates that the uncalibrated ensemble models are less certain than the single models, where a single regressor is usually over-confident (Table 5). So, there is an implicit calibration by ensembling models, and conformal prediction of ensemble models shrinks the boundaries reasonably. Since the calibration set is limited and possibly biased, the ensemble linear regressor on the Concrete and Energy dataset and the ensemble quantile regressor on the Energy dataset are miscalibrated after conformal prediction.

4 Conclusion

In this project, we review uncertainty and calibration from theoretical and empirical perspectives. Classic and popular calibration methods are also re-implemented for neural networks in the context of classification and regression. The following observations are made: 1). The uncalibrated neural network is usually overconfident and badly calibrated. 2). Matrix scaling can calibrate confidence with the risk of jeopardizing accuracy. But, temperature scaling is a remarkably efficient method with no such concern. 3). Conformal prediction is distributional-free

	L.R.		**Q.R.**
Dataset	**Uncali.**	**Conformal**	**Uncali. & Conformal**
Solar	0.3723	0.3925	0.1953
Housing	0.0518	0.0409	0.0726
Concrete	0.0594	0.0559	0.1826
Energy	0.0001	0.0001	0.0156
Power	0.0006	0.0006	0.0026
Red wine	0.0235	0.0275	0.2951
Yacht	0.1541	0.0837	0.0353

Table 8: Sharpness of ensemble multi-layer perceptrons.

but the outcome intervals can be affected by either the outliers in the finite calibration set or the distribution shift between the calibration set and test set. 4). Ensembling is another simple and weak calibration method that depends on ensemble methods. But, it is computationally expensive. Future work can investigate other variations of conformal prediction that mitigate distribution shift and evaluation methods for conformal prediction.

References

[1] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(89)90020-8. URL https://www.sciencedirect.com/science/article/pii/0893608089900208.

[2] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 18(7):1527–1554, 07 2006. ISSN 0899-7667. doi: 10.1162/neco.2006.18.7.1527. URL https://doi.org/10.1162/neco.2006.18.7.1527.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

[5] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2016. URL https://arxiv.org/abs/1608.06993.

[6] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. 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/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[8] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. URL https://arxiv.org/abs/2010.11929.

[10] Tilmann Gneiting and Matthias Katzfuss. Probabilistic forecasting. *Annual Review of Statistics and Its Application*, 1(1):125–151, 2014. doi: 10.1146/annurev-statistics-062713-085831. URL https://doi.org/10.1146/annurev-statistics-062713-085831.

[11] Tilmann Gneiting and Roopesh Ranjan. Combining predictive distributions. *Electronic Journal of Statistics*, 7(none):1747 – 1782, 2013. doi: 10.1214/13-EJS823. URL https://doi.org/10.1214/13-EJS823.

[12] Francis X. Diebold, Todd A. Gunther, and Anthony S. Tay. Evaluating density forecasts with applications to financial risk management. *International Economic Review*, 39(4): 863–883, 1998. ISSN 00206598, 14682354. URL http://www.jstor.org/stable/2527342.

[13] Thomas M. Hamill. Interpretation of rank histograms for verifying ensemble forecasts. *Monthly Weather Review*, 129(3):550 – 560, 2001. doi: 10.1175/1520-0493(2001)129⟨0550:IORHFV⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/mwre/129/3/1520-0493_2001_129_0550_iorhfv_2.0.co_2.xml.

[14] 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. doi: https://doi.org/10.1111/j.1467-9868.2007.00587.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00587.x.

[15] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437. URL https://doi.org/10.1198/016214506000001437.

[16] Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1):33–50, 1978. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1913643.

[17] Roger Koenker and Kevin F. Hallock. Quantile regression. *Journal of Economic Perspectives*, 15(4):143–156, December 2001. doi: 10.1257/jep.15.4.143. URL https://www.aeaweb.org/articles?id=10.1257/jep.15.4.143.

[18] Roger Koenker. *Quantile Regression*. Econometric Society Monographs. Cambridge University Press, 2005. doi: 10.1017/CBO9780511754098.

[19] Frank R. Hampel. The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393, 1974. ISSN 01621459. URL http://www.jstor.org/stable/2285666.

[20] Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 609–616, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1558607781.

[21] Pedro Domingos and Michael Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2):103–130, Nov 1997. ISSN 1573-0565. doi: 10.1023/A:1007413511361. URL https://doi.org/10.1023/A:1007413511361.

[22] 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*, KDD '02, page 694–699, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 158113567X. doi: 10.1145/775047.775151. URL https://doi.org/10.1145/775047.775151.

[23] Tim Robertson, F. T. Wright, and Richard L. Dykstra. *Order restricted statistical inference*. Wiley, Chichester, 1988.

[24] Miriam Ayer, H. D. Brunk, G. M. Ewing, W. T. Reid, and Edward Silverman. An Empirical Distribution Function for Sampling with Incomplete Information. *The Annals of Mathematical Statistics*, 26(4):641 – 647, 1955. doi: 10.1214/aoms/1177728423. URL https://doi.org/10.1214/aoms/1177728423.

[25] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

[26] 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*, ICML'17, page 1321–1330. JMLR.org, 2017.

[27] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

[28] Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression, 2016. URL `https://arxiv.org/abs/1604.04173`.

[29] Anastasios Nikolas Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2021. URL `https://arxiv.org/abs/2107.07511`.

[30] Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/8fb21ee7a2207526da55a679f0332de2-Paper.pdf`.

[31] Morris H. DeGroot and Stephen E. Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 32(1/2):12–22, 1983. ISSN 00390526, 14679884. URL `http://www.jstor.org/stable/2987588`.

[32] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 625–632, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595931805. doi: 10.1145/1102351.1102430. URL `https://doi.org/10.1145/1102351.1102430`.

[33] Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2901–2907. AAAI Press, 2015. ISBN 0262511290.

[34] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[35] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. URL `http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf`.

[36] Judith Lean, Juerg Beer, and Raymond Bradley. Reconstruction of solar irradiance since 1610: Implications for climate change. *Geophysical Research Letters*, 22(23):3195–3198, 1995.

[37] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

[38] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. URL `https://arxiv.org/abs/1412.6980`.