ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2.
Available from http://www.i6doc.com/en/.

Automatic Miscalibration Diagnosis: Interpreting Probability Integral Transform (PIT) Histograms

Ondřej Podsztavek[1], Alexander I. Jordan[2], Pavel Tvrdík[1], and Kai L. Polsterer[2] *

1- Czech Technical University in Prague - Faculty of Information Technology
Thákurova 9, 160 00 Prague 6 - Czechia

2- Heidelberg Institute for Theoretical Studies
Schloss-Wolfsbrunnenweg 35, 691 18 Heidelberg - Germany

Abstract. Quantifying the predictive uncertainty of a model is essential for risk assessment. We address the proper calibration of the predictive uncertainty in regression tasks by employing the probability integral transform (PIT) histogram to diagnose miscalibration. PIT histograms are often difficult to interpret, and therefore we present an approach to an automatic interpretation of PIT histograms based on an interpreter trained with a synthetic data set. Given a PIT histogram of a model and a data set, the interpreter can estimate the data-generating distribution of the data set with the main purpose of identifying the cause of miscalibration.

1 Introduction

Predictive (especially machine learning) models are prevalent in real-world applications. Although these models are useful, the task of making perfect predictions remains unattainable. To allow risk assessment, we have to quantify the *predictive uncertainty* that is generally represented by a probability distribution. Assessing the quality of those uncertainties is an essential task that we address in this paper.

A key to this task is the paradigm of maximising the *sharpness* of predictive distributions subject to their *calibration* [1]. Here, we focus on regression tasks and use the probability integral transform (PIT) histogram as a tool for *miscalibration diagnosis*. In the machine learning literature, the *calibration plot* (also known as the *reliability diagram*) is a common tool to diagnose miscalibration.[1] One should be able to diagnose miscalibration by visually inspecting a PIT histogram or calibration plot. However, understanding the *cause* of miscalibration requires a lot of experience. Scalar scores such as the calibration error [2] express only the degree of miscalibration, not its cause.

*Ondřej Podsztavek, Alexander I. Jordan, and Kai L. Polsterer gratefully acknowledge the generous and invaluable support of the Klaus Tschira Foundation. Ondřej Podsztavek acknowledges the support of his co-supervisor Petr Škoda, and the Grant Agency of the Czech Technical University in Prague (No. SGS23/209/OHK3/3T/18).

[1]These two tools are equivalent because both display an estimate of the PIT distribution: the PIT histogram shows a density estimate, whereas the calibration plot displays an estimate of the cumulative distribution function.

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

2 Calibration, PIT histograms & proper scoring rules

Following [1], at instance $i \in \{1, \ldots, n\}$, nature chooses a true data-generating distribution G_i and a predictive model picks a predictive cumulative distribution function (CDF) F_i. Both G_i and F_i might depend on stochastic parameters. The predictive distributions are *probabilistically calibrated* relative to the true data-generating distributions if $\frac{1}{n}\sum_{i=1}^{n} G_i \circ F_i^{-1}(p) \to p$ for all $p \in (0, 1)$ where the arrow denotes the almost sure convergence as $n \to \infty$. This definition is equivalent to the uniformity of PIT values $\{p_i = F_i(y_i) \mid i \in \{1, \ldots, n\}\}$, where an outcome y_i is a random number with distribution G_i. The PIT is translation- and scale-invariant. We diagnose miscalibration by visualising the histogram of PIT values (hereafter the *PIT histogram*) and inspecting its shape.

Simple causes of miscalibration (*bias*, *underdispersion* and *overdispersion*) can be identified easily. They express themselves respectively as a PIT histogram with a single peak at an edge, a U-shaped and a bell-shaped PIT histogram. However, if the cause of miscalibration is not a simple one or multiple causes co-occur, potential shapes of PIT histograms cannot be easily enumerated, which makes their interpretation difficult or even impossible for inexperienced users. Therefore, we provide a user-friendly interpretation of PIT histograms, from which users can recognise causes of miscalibration. Subsequently, users can deal with those causes and get more reliable predictive distributions.

The PIT histogram is a useful diagnostic tool, but unsuitable for comparing two predictive models. To compare predictive models, we employ *proper scoring rules*. A scoring rule is a loss function for predictive distributions, as opposed to point predictions. It is *proper* if it has the property that a predictive distribution that matches the true data-generating distribution minimises the expected score. Implicitly, that property means that a proper scoring rule measures calibration and sharpness jointly. The two most used proper scoring rules are the *negative log-likelihood* $\mathrm{NLL}(f_i, y_i) = -\log f_i(y_i)$, where f_i denotes the probability density function corresponding to CDF F_i, and the *continuous ranked probability score*, $\mathrm{CRPS}(F_i, y_i) = \int (F_i(a) - \mathbf{1}_{a \geq y_i})\, da$. We need to use a proper scoring rule to measure and confirm the improvement in predictive performance that can be achieved by dealing with causes of miscalibration.

3 Automatic interpretation of PIT histograms

To facilitate an interpretation of a PIT histogram, we propose to perform a decomposition into a data-generating and a predictive distribution. These distributions allow us to reconstruct a PIT histogram that is close to the original PIT histogram. We achieve this decomposition using a machine learning model called an *interpreter*. Because the PIT is translation- and scale-invariant, an interpreter trained on a *synthetic data set of PIT histograms* can interpret a given PIT histogram independently of the original translation and scale of data-generating and predictive distribution pairs. Given the PIT histogram of a predictive model and data set, its interpretation allows us to diagnose miscali-

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

bration of the model by comparing the estimated data-generating and predictive distribution.

3.1 Synthetic data set of PIT histograms

A synthetic data set has to be relevant to the particular application, i.e. relevant to expected data-generating and predictive distributions. The synthetic data set consists of m PIT histograms with b bins, each generated from n pairs of data-generating and predictive distributions.

We generate the j-th PIT histogram, where $j \in \{1, \dots, m\}$, by first generating a set of PIT values, and then assigning these PIT values to the predefined bins. Technically, that means choosing a pair of predictive and data-generating CDFs ($F_i^{(j)}$ and $G_i^{(j)}$) for each $i \in \{1, \dots, n\}$, sampling an outcome $y_i^{(j)}$ from $G_i^{(j)}$, and computing $p_i = F_i^{(j)}(y_i^{(j)})$. Then, we assign the PIT values into b bins, and calculate the corresponding relative frequencies, such that the area under the histogram integrates to 1 and is therefore independent of n.

3.2 Interpreter

The input of the interpreter is a PIT histogram, and its output estimates the data-generating distribution that led to the PIT histogram. In particular, because a mixture of normal distributions can approximate any data-generating distribution if it has enough components, the interpreter is a mixture density network (MDN) [3]. To allow data-generating distributions of the synthetic data set to be from any family of distributions, the interpreter is trained with a Monte Carlo approximation to 1-Wasserstein distance between true $G_i^{(j)}$ and predicted $\hat{G}^{(j)}$ data-generating CDFs:

$$\frac{1}{m}\sum_{j=1}^{m}\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{o}|G_i^{(j)}(a_k) - \hat{G}^{(j)}(a_k)|,$$

where $a_1 < \cdots < a_o$ are equally spaced real numbers, a_1 and a_o are chosen according to the domain of the data-generating CDF $G_i^{(j)}$, and o is large enough to get a sufficiently accurate approximation.

4 Experiments

In probabilistic modelling, unimodal predictive distributions are often used to model multimodal data-generating distributions (e.g. [4, 5]). Therefore, we choose to experiment with a simple synthetic data set based on the normal family. For the j-th PIT histogram, every outcome $y_i^{(j)}$ is a random number from a data-generation distribution $G_i^{(j)}$. For simplicity, we assume that $G_i^{(j)}$ is the same for all i. Specifically, $G^{(j)}$ is a mixture of two normal distributions, i.e. $y_i^{(j)}$ takes a random value from $\mathcal{N}(-d^{(j)}/2, t^{(j)})$ with probability $w^{(j)}$ or $\mathcal{N}(d^{(j)}/2, v^{(j)})$ with probability $1 - w^{(j)}$. By manipulating the parameters

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

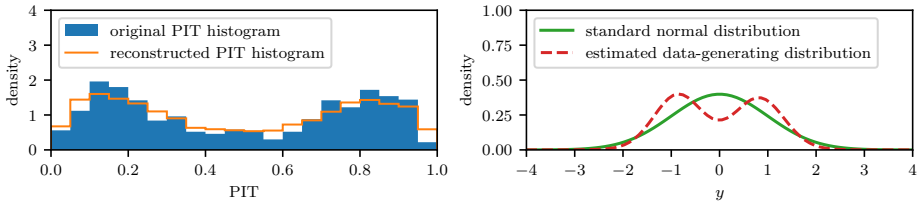

Fig. 1: The PIT histogram (left) of a density network (DN) trained to solve the simple synthetic inverse problem and its interpretation (right).

separation $d^{(j)}$, weight $w^{(j)}$, and variances $t^{(j)}$ and $v^{(j)}$, we can obtain PIT histograms of predictive models that are calibrated, under- and overestimated, under- and overdispersed, or have an incorrect number of modes. For simplicity, we fix the predictive distribution $F_i^{(j)}$ to $\mathcal{N}(0,1)$ for all i and j. On the simple synthetic inverse problem and real-world data sets in sections 4.1 and 4.2, we observe that reconstructed PIT histograms match the original PIT histograms. This is already possible with the current choices of the fixed predictive distribution and the family of data-generating distributions. We will experiment with further distributions from various families with even more modes in the future.

In order to have a wide range of visually distinct PIT histograms in the synthetic data set of the interpreter, we decided to 1. define separation $d^{(j)} = 2(1 - a^{(j)}a^{(j)})$, where $a^{(j)}$ is sampled from the continuous uniform distribution $U(0.1, 1)$, 2. define variances $t^{(j)} = 2^{b^{(j)}}$ and $v^{(j)} = 2^{c^{(j)}}$, where $b^{(j)}$ and $c^{(j)}$ are sampled from $U(-2, 2)$, and 3. sample weight $w^{(j)}$ from $U(0, 1)$. Each generated PIT histogram has $b = 20$ bins containing a total of $n = 10^4$ PIT values per histogram.

Our experimental interpreter has a single hidden layer with 16 neurons and outputs a mixture of five normal distributions, which gives the interpreter enough flexibility with respect to our experimental synthetic data set.[2]

4.1 Evaluation on a simple synthetic inverse problem

First, we present a simple synthetic inverse problem for which a bimodal predictive distribution is adequate. The corresponding data set consists of 10^4 input-outcome pairs (x_i, y_i), where $x_i = u_i'^2$, u_i' is sampled from $U(-1, 1)$, $y_i = u_i' + 0.25\epsilon_i$, and ϵ_i is sampled from $\mathcal{N}(0, 1)$. We train on it a *density network* (DN) [6] as a simple model with a unimodal normal predictive distribution.

Figure 1 displays the PIT histogram of the DN and the interpretation of the PIT histogram. The non-uniform PIT histogram reveals that the DN is miscalibrated. The interpretation clearly shows that the cause of miscalibration is that a unimodal predictive distribution is used to model a bimodal data-generating distribution.

[2]For more details, see `https://github.com/podondra/calibration`.

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

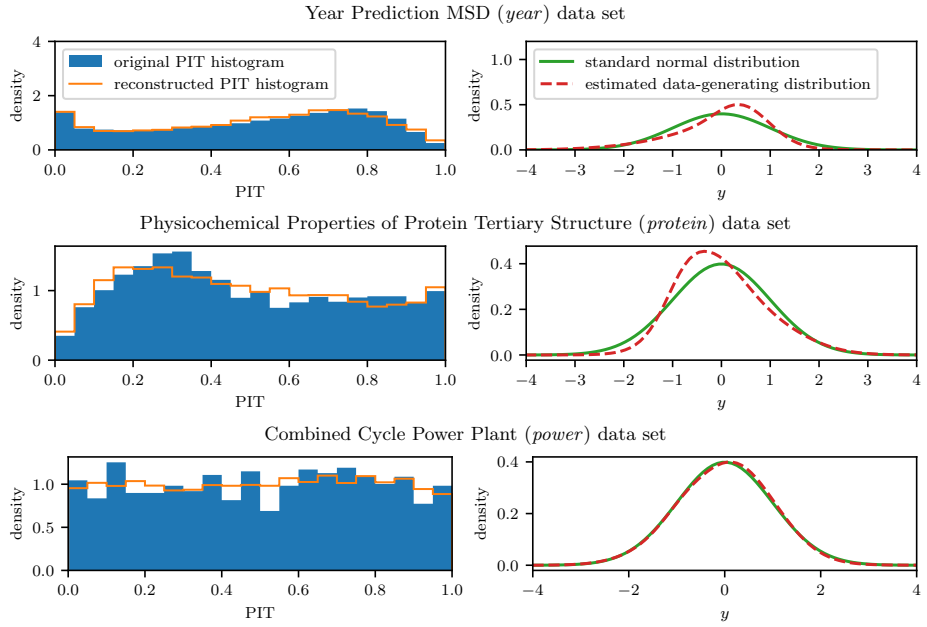

Fig. 2: PIT histograms (left) of DNs trained on the data sets from UCI Machine Learning Repository and interpretations of those PIT histograms (right).

4.2 Evaluation on real-world data sets

We choose the Year Prediction MSD, Physicochemical Properties of Protein Tertiary Structure, and Combined Cycle Power Plant (hereafter *year*, *protein*, and *power*, respectively) data sets from UCI Machine Learning Repository, because they are commonly used for the evaluation of predictive uncertainties (e.g. [4, 5]).

Figure 2 displays PIT histograms of DNs trained on the data sets and their interpretations. In the case of the *year* data set, the PIT histogram of the DN is not uniform, indicating miscalibration, and its cause is more easily identified with the proposed decomposition. Our interpreter suggests that the normal predictive distribution is insufficiently flexible in its shape to model the data-generating distribution, and that it would be better to use a right-skewed predictive distribution. On the *protein* data set, the decomposition is similar to the one of the *year* data set. However, on the *power* data set, we observe that the PIT histogram of the DN exhibits some noise but is uniform. It is plausible that the data-generating distribution deviates only slightly from a normal distribution.

Table 1 reiterates the well-known fact that dealing with causes of miscalibration leads to tangible improvements in the predictive performance. We deal with the skewness by training MDNs that output mixtures of five normal distributions for simplicity. In real applications, an appropriate simple predictive distribution inferred from the interpretation should be used, not a complex mixture of many

ESANN 2024 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 9-11 October 2024, i6doc.com publ., ISBN 978-2-87587-090-2. Available from http://www.i6doc.com/en/.

data set	model	mean NLL	mean CRPS
year	DN	3.373 ± 0.003	4.322 ± 0.013
	MDN	**3.094 ± 0.002**	**4.040 ± 0.007**
protein	DN	2.805 ± 0.039	2.342 ± 0.025
	MDN	**2.086 ± 0.017**	**1.940 ± 0.019**
power	DN	2.795 ± 0.018	2.175 ± 0.030
	MDN	**2.673 ± 0.023**	**2.093 ± 0.042**

Table 1: Comparison of models in terms of the mean NLL and mean CRPS.

distributions. We report the mean NLL and mean CRPS as performance metrics, accompanied by standard errors that are estimated from splitting the data sets into five train-test folds. The gap in predictive performance between DNs and MDNs is large for the *year* and *protein* data sets. This gap is mainly due to miscalibration when assuming a symmetric predictive distribution. For the *power* data set, the gap is small because both models are almost calibrated.

5 Discussion

In probabilistic machine learning, we often focus solely on improving predictive models with respect to metrics such as NLL or CRPS (e.g. with a Bayesian network [5] or deep ensemble [4]), frequently at the cost of neglecting causes of miscalibration (e.g. whether the family of predictive distributions is underspecified). The proposed approach yields plots that essentially contain the same information as PIT histograms or calibration plots, but in a form that makes causes of miscalibration more obvious. By dealing with the causes of miscalibration, we get predictive models that are more reliable, and output calibrated predictive distributions. In turn, the overall predictive performance of these models is superior in terms of metrics such as the NLL and CRPS.

References

[1] T. Gneiting, F. Balabdaoui, and A. E. Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 2007.

[2] V. Kuleshov, N. Fenner, and S. Ermon. Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[3] C. M. Bishop. Mixture density networks. Technical report, Aston University, 1994.

[4] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30*, 2017.

[5] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning*, 2016.

[6] D. A. Nix and A. S. Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 IEEE International Conference on Neural Networks*, 1994.