Looking at the posterior: accuracy and uncertainty of neural-network predictions

(article starts on next page)

PAPER • OPEN ACCESS

Looking at the posterior: accuracy and uncertainty of neural-network predictions

View the article online for updates and enhancements.

MACHINE LEARNING
Science and Technology

PAPER

Looking at the posterior: accuracy and uncertainty of neural-network predictions

Hampus Linander[1,2,*] ⓘ, Oleksandr Balabanov[3] ⓘ, Henry Yang[1] and Bernhard Mehlig[1] ⓘ

1 Department of Physics, University of Gothenburg, 41296 Gothenburg, Sweden
2 Department of Mathematical Sciences, Chalmers University of Technology, University of Gothenburg, 41296 Gothenburg, Sweden
3 Department of Physics, Stockholm University, 10691 Stockholm, Sweden
* Author to whom any correspondence should be addressed.

E-mail: linander@chalmers.se

Abstract

Bayesian inference can quantify uncertainty in the predictions of neural networks using posterior distributions for model parameters and network output. By looking at these posterior distributions, one can separate the origin of uncertainty into aleatoric and epistemic contributions. One goal of uncertainty quantification is to inform on prediction accuracy. Here we show that prediction accuracy depends on both epistemic and aleatoric uncertainty in an intricate fashion that cannot be understood in terms of marginalized uncertainty distributions alone. How the accuracy relates to epistemic and aleatoric uncertainties depends not only on the model architecture, but also on the properties of the dataset. We discuss the significance of these results for active learning and introduce a novel acquisition function that outperforms common uncertainty-based methods. To arrive at our results, we approximated the posteriors using deep ensembles, for fully-connected, convolutional and attention-based neural networks.

1. Introduction

The user of an artificial neural-network wants to know when the prediction of the model is accurate and trustworthy. When target ground truth is unavailable, as is usually the case, one must instead rely upon surrogate measures that correlate with accuracy and trustworthiness in a robust way. Uncertainty quantification aims to provide such measures. Recently there has been an intensive effort towards a better understanding of uncertainty of neural-network predictions [1, 2]. To quantify this uncertainty in a way that informs on the efficacy of the model, and to identify its sources, is of key significance in many applications of machine-learning algorithms using neural networks, from real-time predictions to active learning [3–9].

When the outputs of neural networks can be viewed as probability distributions over possible output values, certain distributional measures naturally capture the uncertainty of the network predictions. For instance, if the output distribution is sharply peaked, one might expect the prediction to be accurate. To which extent this expectation is borne out, depends not only on the model architecture and parameters, but also on the input data (for example whether it is from a domain the model has knowledge about).

Bayesian inference [10] provides a theoretical framework to reason about the conditional distribution of model parameters, and of the model output, given the available training data. More precisely, given a neural network with parameters θ, a prior $p(\theta)$, and a training dataset $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots\}$ of pairs (*input, target*), Bayesian arguments determine a distribution over the neural-network parameters $p(\theta|\mathcal{D})$ [11]. This so-called posterior distribution tells us the probability of different model parameters given the training dataset. Using this posterior distribution for the parameters, a corresponding posterior distribution of the neural-network predictions,

$$p(y|\mathcal{D}, x) = \int_\theta p(y|\theta, x)\, p(\theta|\mathcal{D})\, \mathrm{d}\theta, \tag{1}$$

can be derived. The posterior predictive distribution in equation (1) is the marginalization over model parameters θ, conditioned on a particular input x that is either previously unseen or contained in the training dataset. Together, the posterior distribution of model parameters and the posterior predictive distribution characterize the knowledge of the model, and their entropies provide measures of uncertainty.

The entropy of the posterior parameter distribution measures the uncertainty of the model parameters, and as such is *epistemic*. Uncertainty stemming from the input data is often referred to as *aleatoric*, and it is related to the entropy of the posterior predictive distribution [12]. Bayesian modeling makes the distinction between epistemic and aleatoric uncertainty clear. The posterior probability $p(\theta|\mathcal{D})$ of the neural-network parameters θ given the observations $\mathcal{D}$ determines the epistemic uncertainty. The aleatoric part, by contrast, is determined by the likelihood $p(y|\theta, x)$ given model parameters θ and input data x. The entropy of the posterior predictive distribution in equation (1) contains through the two factors in the integrand a mixture of aleatoric and epistemic uncertainty, referred to as predictive uncertainty.

Hence, the decomposition of uncertainty into aleatoric and epistemic parts can be quantified by the entropy of the posterior predictive distribution in equation (1), containing both aleatoric and epistemic contributions, and the posterior parameter distribution $p(\theta|\mathcal{D})$. The epistemic uncertainty associated with a particular input x can then be quantified by the conditional mutual information $I(\theta; y|x, \mathcal{D}) = H(\theta|\mathcal{D}) - E_{p(y|x,\mathcal{D})}[H(\theta|(x,y), \mathcal{D})]$, where $H(\cdot|\cdot)$ denotes conditional entropy, between parameters and predictive distribution conditioned on the input x and training dataset $\mathcal{D}$ [11]. However, it turns out that this separation of uncertainty into aleatoric and epistemic parts can be hard to use in practice, as evident by the following two examples. The first one comes from active learning: It is found that using a decomposition of predictive uncertainty into aleatoric and epistemic parts does not necessarily improve sample selection [13–16]. This is surprising, since high epistemic uncertainty is a natural criterion for samples where the model can hope to learn something new. The second example relates to safety critical applications such as medical diagnosis. For medical image analysis in particular, uncertainty quantification has been used to improve model precision, and to guide clinical assessment [17–21]. By selecting for low epistemic and aleatoric uncertainty one could hope to increase prediction accuracy at the cost of recall. Again, marginalized measures in terms of epistemic and aleatoric parts do not seem to provide better precision over predictive uncertainty [22]. Why is it hard to use the decomposition into aleatoric and epistemic uncertainty effectively? To answer this question we evaluate the correlation between accuracy and uncertainty quantification using the joint distribution of predictive uncertainty and epistemic uncertainty, as measured by the entropies of the posterior predictive distribution and the posterior parameter distribution.

Our results show that the accuracy of a model has non-trivial correlations with the combination of predictive and epistemic uncertainty, and that this correlation depends on model architecture and dataset. We test these insights in application by proposing and evaluating a novel acquisition function based on expected accuracy. Using the joint distribution of predictive uncertainty and epistemic uncertainty, we quantify how the approximate posteriors of three common neural-network architectures for image classification differ from each other and how they depend on data-distributional shifts in the form of impulse noise [23] for MNIST [24] and CIFAR [25]. We conclude that the joint distribution contains important information regarding model accuracy, and that it needs to be calibrated for a particular model architecture and dataset.

2. Contributions

- We quantify the joint distribution of prediction accuracy, predictive uncertainty and epistemic uncertainty for the MNIST [24] and CIFAR [25] datasets. The non-trivial patterns we find makes it clear that model accuracy cannot be understood in terms of marginalized uncertainty measures.
- We introduce a novel acquisition function for active learning that outperforms acquisition using marginalized uncertainty distributions.
- We use the joint distribution of predictive entropy and conditional mutual information between parameters and targets to quantify the variability of uncertainty measures over different model architectures and data-distributional shifts.
- We show that neural networks with different architectures can disagree about the origin of uncertainty for data-distributional shifts in image classification tasks.

3. Related work

Depeweg *et al* [26] uses aleatoric and epistemic uncertainty separately for active learning on low dimensional regression and classification tasks, concluding that epistemic uncertainty provides a better criterion for selecting training samples than the predictive uncertainty. This provides a clear example of where the intuition of epistemic uncertainty being a good sample selection criteria holds. In contrast, this does not

extend to higher dimensional computer vision domains where epistemic uncertainty is no longer superior [16]. In the context of reinforcement learning, [26] uses a combination of aleatoric and epistemic uncertainty to find balanced policies. The joint distribution, architecture dependence, and dataset dependence is not considered. In [21], the correlation between accuracy and uncertainty is quantified for image classification. Correlation with accuracy is shown for predictive uncertainty and epistemic uncertainty separately, the architecture and dataset dependence is not discussed. The joint distribution of aleatoric and epistemic uncertainty is considered in [20], where the correlation between predictive probabilities and the joint uncertainty distribution is quantified in the context of medical image semantic segmentation. Using a fixed residual U-Net architecture and datasets for semantic segmentation it was shown that there is a correlation between predictive probabilities and the uncertainty measures, and that accuracy for their semantic segmentation model on the considered datasets shows correlation with epistemic uncertainty. In terms of open questions, [20] highlight the effect of data-distributional shifts, model architecture, and data modality on the quality of uncertainty quantification. These are in line with two of our target questions on how the perceived origin of uncertainty depends on model architecture and how the dataset impacts uncertainty quantification. Quantifying uncertainty under data-distributional shifts was investigated in [27] where accuracy, calibration and entropy of the posterior predictive is evaluated for different shifts introduced in [28]. Here, the epistemic uncertainty is not considered and the model dependence of the uncertainty measures is not analyzed.

4. Background

4.1. Bayesian inference

An artificial neural network f with parameters $\theta \in \mathbb{R}$ can be seen as a map from input space X to the output space Y [29], where we take Y to be the space of distributions over possible outcomes so that the likelihood is given by $p(y|x, \theta) = f(y, x; \theta)$. Bayesian inference [30] allows us to reason about uncertainty in terms of posterior distributions for the parameters of a model. With a training dataset $\mathcal{D} \in X \times Y$ corresponding to observations (x_t, y_t) with $t \in \{1 \ldots N\}$, the posterior distribution for the neural-network parameters $p(\theta|\mathcal{D})$ can be calculated using Bayes theorem in terms of the likelihood as

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta) p(\theta)}{p(\mathcal{D})}, \tag{2}$$

where $p(\mathcal{D}|\theta)$ (assuming independent samples in the dataset $\mathcal{D}$) can be expressed as $p(\mathcal{D}|\theta) = \prod_{i=1}^{N} f(y_i, x_i; \theta)$, $p(\theta)$ the parameter prior and the evidence $p(\mathcal{D})$ is the marginalization over the parameter prior.

A prediction, or more generally a distribution over possible outputs, is computed using the posterior predictive distribution in equation (1) by marginalizing over the parameters. Each parameter configuration is weighted by its posterior probability given the training data. For large neural-network architectures, it is in general very challenging to evaluate the high-dimensional integral over θ [31]. To implement Bayesian inference for neural networks, the posterior needs to be approximated.

There are a number of different methods available that range from computationally expensive Monte-Carlo simulations [10] to more efficient dropout approximations [32, 33], as well as simpler ensembling methods [34]. The most accurate approximations to the true posterior rely on the Hamiltonian Monte-Carlo (HMC) methods [35–37]. Through intensive computational efforts, HMC methods have recently been used to approximate the posteriors of larger convolutional neural networks such as a 20-layer ResNet [31]. The HMC computations show that simpler approximation schemes such as ensembling and variational inference can fail to accurately describe the true posterior, but that ensembles often provide more accurate posteriors than more advanced methods. Here we use deep ensembles, as described in appendix A, to approximate the Bayesian posteriors of the neural networks [38].

4.2. Uncertainty quantification

For classification over a discrete set of M classes, the entropy [39] of the predictive distribution

$$H(p) = -\sum_{c=1}^{M} p(c) \log(p(c)), \tag{3}$$

provides a measure of the information content and thus its uncertainty.

In terms of the ensemble approximations of the posterior predictive distribution, see appendix equation (A.1) for details, this takes the form

$$H(p(y|x,\mathcal{D})) = -\sum_{c=1}^{M}\left(\frac{1}{N}\sum_{i=1}^{N}f_c\left(x;\theta^{(i)}\right)\right)\log\left(\frac{1}{N}\sum_{i=1}^{N}f_c\left(x;\theta^{(i)}\right)\right), \tag{4}$$

where $H(p(y|x,\mathcal{D}))$ is the entropy of the posterior predictive distribution, referred to here as predictive uncertainty. The entropy of $p(y|x,\mathcal{D})$ in equation (4), as a measure of uncertainty, contains contributions that are both epistemic and aleatoric.

Epistemic uncertainty stems from uncertainty in model parameters. This is captured by the shape of the posterior distribution of these parameters. To quantify the epistemic uncertainty associated with a single data sample x, we can ask how this shape changes when the dataset is extended to include x [11]. The epistemic uncertainty associated with a data sample x can be quantified by the expected change in entropy of the model parameter posterior distribution $p(\theta|\mathcal{D})$ when x is added to the observations [11],

$$I(x) = H(p(\theta|\mathcal{D})) - E_{p(y|x,\mathcal{D})}\left[H(p(\theta|\mathcal{D},(x,y)))\right]. \tag{5}$$

The entropy difference in equation (5) is the conditional mutual information $I(\theta;y|x,\mathcal{D})$ between the parameter posterior and the predictive posterior distribution for the new sample point (x,y). Since mutual information is symmetric, this can be written in terms of the entropy of the predictive distribution instead:

$$I(x) = H(p(y|x,\mathcal{D})) - E_{p(\theta|\mathcal{D})}\left[H(p(y|\theta,x))\right], \tag{6}$$

where $H(p(y|\theta,x))$ is the entropy of the likelihood $p(y|\theta,x)$. See appendix B for an example of how this relation manifests in a toy model. The first term in equation (6) is the entropy of the posterior predictive distribution given the dataset $\mathcal{D}$, whereas the second term is the expected value of the likelihood entropy over the model posterior distribution.

Using the approximate posteriors from the ensemble method, the first term in equation (6) is given by equation (4) and the second term is given by

$$E_{p(\theta|\mathcal{D})}\left[H(p(y|\theta,x))\right] = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{M}f_c\left(x;\theta^{(i)}\right)\log\left(f_c\left(x;\theta^{(i)}\right)\right). \tag{7}$$

Together, equations (4) and (7) provide a concrete way to evaluate epistemic uncertainty as defined by equation (6) in practice.

The epistemic uncertainty in equation (6) is large when the posterior predictive entropy is large and the mean likelihood entropy is small. In terms of the ensemble members this corresponds to the situation where each member has a sharp distribution but they disagree about the mean. Large aleatoric uncertainty is ascribed to broad output distributions from the individual members $f(\cdot;\theta^{(i)})$, also implying a large posterior predictive entropy. Since entropy is positive, both the term $H(p(y|x,\mathcal{D}))$ and the expected entropy of the likelihood in equation (6) are positive, and so the entropy difference in equation (6) is bounded by the posterior entropy in equation (4), resulting in the inequality

$$I(x) \leqslant H(x), \tag{8}$$

i.e. a large epistemic uncertainty implies a large posterior predictive entropy. This can be seen by noting that a collection of sharp member distributions that disagree about the mean necessarily adds up to a broad mean distribution. One way to describe a sample with large epistemic uncertainty is that the model might fit the data well, but in many different ways. The parameters with high posterior probability could all result in sharp distributions, whereas the full posterior can be broad [40].

Since the epistemic uncertainty in equation (6) measures the change in posterior entropy, a model can have irrelevant parameters that contribute to a broad posterior distribution, but the change in posterior entropy can still be small when adding a given sample to the training set. If the parameter posterior for some irrelevant parameter is equally broad after we add sample x, then we do not want to consider this as a point of high epistemic uncertainty.

Even though entropic measures are a natural starting point for quantifying uncertainty of predictions, they have been shown to possess a number of unwanted properties such as ataining maximal values in un-intuitive scenarios [41]. In this work we take a pragmatic view and consider uncertainty quantification based on entropy as a flawed, but still practically useful tool to understand neural network predictions.
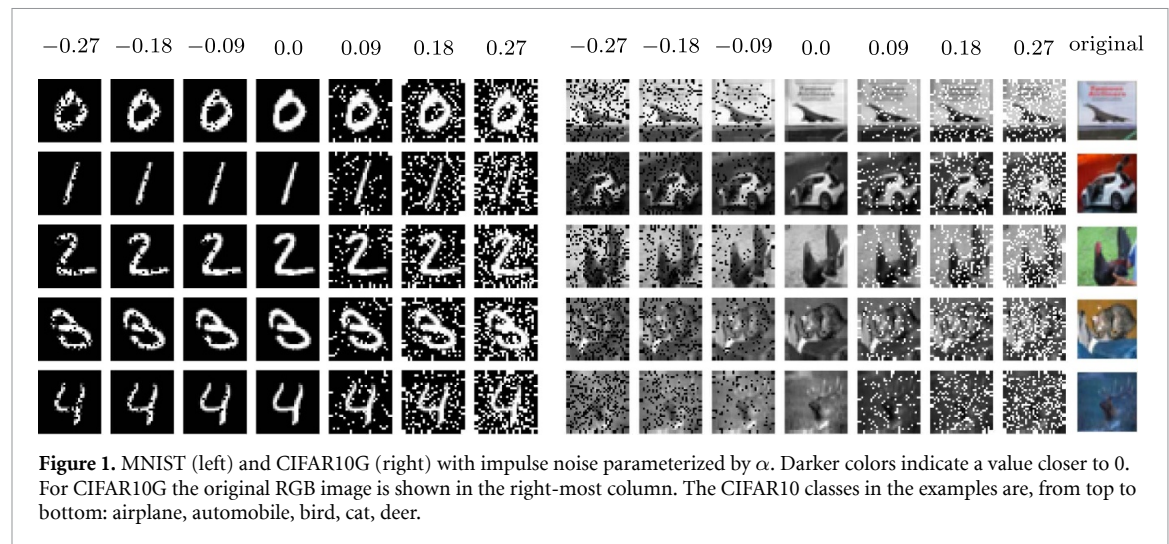
Figure 1. MNIST (left) and CIFAR10G (right) with impulse noise parameterized by α. Darker colors indicate a value closer to 0. For CIFAR10G the original RGB image is shown in the right-most column. The CIFAR10 classes in the examples are, from top to bottom: airplane, automobile, bird, cat, deer.

Table 1. Summary of the three different neural-network architectures used for comparisons. The accuracy for MNIST and CIFAR10G is over the validation datasets with standard deviation calculated for ten separate trainings. The dense model uses three layers (128, 128, 10) with ReLU activations, CNN is model A in [42] and Swin is a shifted window transformer [43] based on the KERAS vision implementation [44].

Model	Parameters	MNIST	CIFAR10G
Dense	118k	$99.5\% \pm 0.07\%$	$45.8\% \pm 0.5\%$
CNN	836k	$99.9\% \pm 0.06$	$74.1\% \pm 0.95\%$
Swin	147k	$97.3\% \pm 0.1\%$	$66.2\% \pm 1.3\%$

5. Methods

5.1. Datasets

Two datasets are used for all numerical experiments, MNIST [24] and a grayscale version of CIFAR10 [25]. We choose these simple datasets to be able to train ensembles efficiently and still be in the domain of realistic images with CIFAR10. To be able to compare relative shifts in uncertainty measures between the two datasets we choose to work with a grayscale version of CIFAR denoted CIFAR10G so that the data-distributional shift acts in exactly the same way for both MNIST and CIFAR10G. MNIST is an example of an image classification dataset with minimal complexity, whereas CIFAR10 provides a more realistic data distribution for image classification. The grayscale conversion for the RGB data from CIFAR10 is given by the standard BT.601 luminance $Y = 0.2989r + 0.5870g + 0.1140b$. We apply impulse noise [23], a common corruption present in digital images, to the original datasets (MNIST, CIFAR10G) where the strength of the perturbation is controlled by a noise parameter α. For 'salted' noise with parameter $\alpha > 0$ a random sample of pixels of size αN_{pixels} are given the maximum value 1.0, and for 'peppered' noise with $\alpha < 0$ a corresponding amount of pixels are set to 0. We pick N_α distinct values between $\alpha_{min} = -0.3$ and $\alpha_{max} = 0.3$.

MNIST consists of 60 000 grayscale images with resolution 28×28. The original CIFAR10 dataset consists of 60 000 RGB images with resolution 32×32 that we resample to a single grayscale channel. See figure 1 for examples of the different noise levels.

5.2. Network architectures

In order to quantify how the architecture affects the uncertainty estimates, we use three neural-network architectures: fully connected (Dense), convolutional (CNN) and attention-based (Swin) neural networks. The fully connected neural network has a three-layer architecture with two 128-neuron hidden layers using ReLU activations. The convolutional neural network is identical to model A in [42], a simple fully convolutional model with five layers using max-pooling for spatial down-sampling and ReLU activations. Finally, we also use a small version of a shifted windows transformer (Swin) [43], a popular attention-based model for computer vision. All architectures use a softmax output layer. See table 1 for a summary of the model sizes and baseline accuracy on the target datasets.

5.3. Active learning

One way of improving the accuracy of a neural-network model is to extend the training set. Active learning [45–48] makes use of the network predictions to choose inputs that contribute the most to increased

accuracy. Which inputs to choose can be phrased in terms of a so-called acquisition function [49], that determines how samples are picked from the pool of unlabelled data. Common acquisition functions include BALD scoring [50] that picks samples with highest epistemic uncertainty as measured by mutual information $I(x)$ in equation (6). Another common acquisition function is or to choose samples with highest predictive uncertainty [51] quantified by $H(y|x, \mathcal{D})$ in equation (4). We refer to these two methods as BALD and max entropy respectively. Based on our results for the correlation between accuracy and the joint distribution (H, I) we propose a novel acquisition function that picks samples with lowest expected accuracy in section 6.2.

6. Results

6.1. Accuracy
Figure 2 shows the accuracy of predictions of three neural networks for MNIST and CIFAR10G as a function of predictive and epistemic uncertainty, evaluated on the union of all noise levels including the uncorrupted test set.

For CIFAR10G in figure 2, the simple fully connected neural network in the first panel has a pronounced correlation between the predictive uncertainty and model accuracy. In the center panel, the convolutional model shows a different pattern where both epistemic and predictive uncertainty correlates equally with accuracy. For the attention-based model in the last panel, there is a more intricate relation. Decreasing epistemic uncertainty correlates with higher accuracy, but predictive uncertainty for a fixed epistemic value does not. In general, lower epistemic and aleatoric uncertainty does not always imply higher accuracy.

Turning to MNIST in figure 2, the first panel shows that the correlation between accuracy and predictive uncertainty is not as pronounced as in the first panel in figure 2. In the domain of MNIST where the dense model has higher accuracy there is instead a stronger correlation between decreasing epistemic uncertainty and prediction accurac. The center panel shows that for the convolutional neural network, for a fixed moderate predictive uncertainty, the accuracy does not monotonically increase with epistemic uncertainty. On the other hand, the attention-based Swin architecture in the right panel shows a monotonically increasing accuracy with decreasing epistemic uncertainty with similar structure as for CIFAR10G.

The moments of the joint uncertainty distribution for CIFAR10G at a fixed distributional shift are summarized in table 2 together with average prediction accuracies. For MNIST, the corresponding joint uncertainty distribution at fixed data-distributional shift is given in table 3. For reference, the moments and accuracy on the unshifted test set is shown in table 4. Comparing the average accuracy over the shifted dataset in table 2 with the accuracy on the validation set in table 4, we see that the dense model retains its accuracy better than the more complex architectures. For each individual architecture, a decrease in average predictive and epistemic uncertainty is accompanied by an increase in average accuracy.

6.2. Active learning
Given the patterns in the accuracy distributions in the (H, I) plane shown in figure 2, is it possible to use information on where the model is less accurate to improve training? According to the results in section 6.1, both the predictive uncertainty and the epistemic uncertainty are needed to parameterize the accuracy. Since standard acquisition functions, such as BALD [50] and max entropy [51], only refer to the marginal uncertainty distributions, we propose here an acquisition function that uses H as well as I to pick inputs from the pool of unlabelled data D_{pool} corresponding to low-accuracy regions in the (H, I) plane.

The goal is to increase the prediction accuracy of the model by choosing low-accuracy inputs from D_{pool} to label and train on. The problem is of course that the samples from D_{pool} are not labeled, therefore there is no direct way of determining the accuracy. The idea is to create a look-up table $EA(H, I)$ that parameterizes the relation between expected accuracy and (H, I). This is done using a labeled calibration dataset $D_{\text{calibration}}$, and discretising H as well as I to obtain the look-up table.

Using this look-up table an active-learning loop is constructed in the following way. For each iteration, the values of H and I are computed for all input samples in the unlabeled pool D_{pool}, and the expected accuracy $EA(H, I)$ is evaluated. The input samples are then ordered by their expected accuracy, and the 50 samples with lowest expected accuracy are labeled and added to the training dataset. A new ensemble of neural networks is trained using the updated training dataset, and a new look-up table is created. This loop is iterated 20 times such that a total of 1000 inputs from D_{pool} have been acquired.

To evaluate the proposed acquisition function LEA, we use a simple active learning setup [50] where an ensemble of neural networks is first trained on a small class-balanced subset of 20 inputs from the CIFAR10 training set. For the accuracy calibration dataset, we use a random subset of 10 000 samples from the CIFAR10 training dataset, corresponding to about 17% of the training data. We discretise the (H, I) plane into 16×16 regular bins limited by the minimum and maximum values of H and I. As the pool of unlabelled

Figure 2. Accuracy distribution in the (H, I)-plane, and marginal accuracy distributions (above and to the right of each panel), for a union over noise levels $-0.27 \leqslant \alpha \leqslant 0.27$ evaluated on CIFAR10G (top row) and MNIST (bottom row). Grey bins indicate less than ten inputs in the corresponding region. Prediction accuracy conditioned on the joint distribution of posterior predictive entropy H and epistemic uncertainty I is shown for the ensemble posterior.

Table 2. Mean and standard deviation for predictive and epistemic uncertainty for fully connected (Dense), convolutional (CNN) and attention-based (Swin) models with posterior approximations using ensembles evaluated on CIFAR10G at fixed data-distributional shift $\alpha = 0.09$.

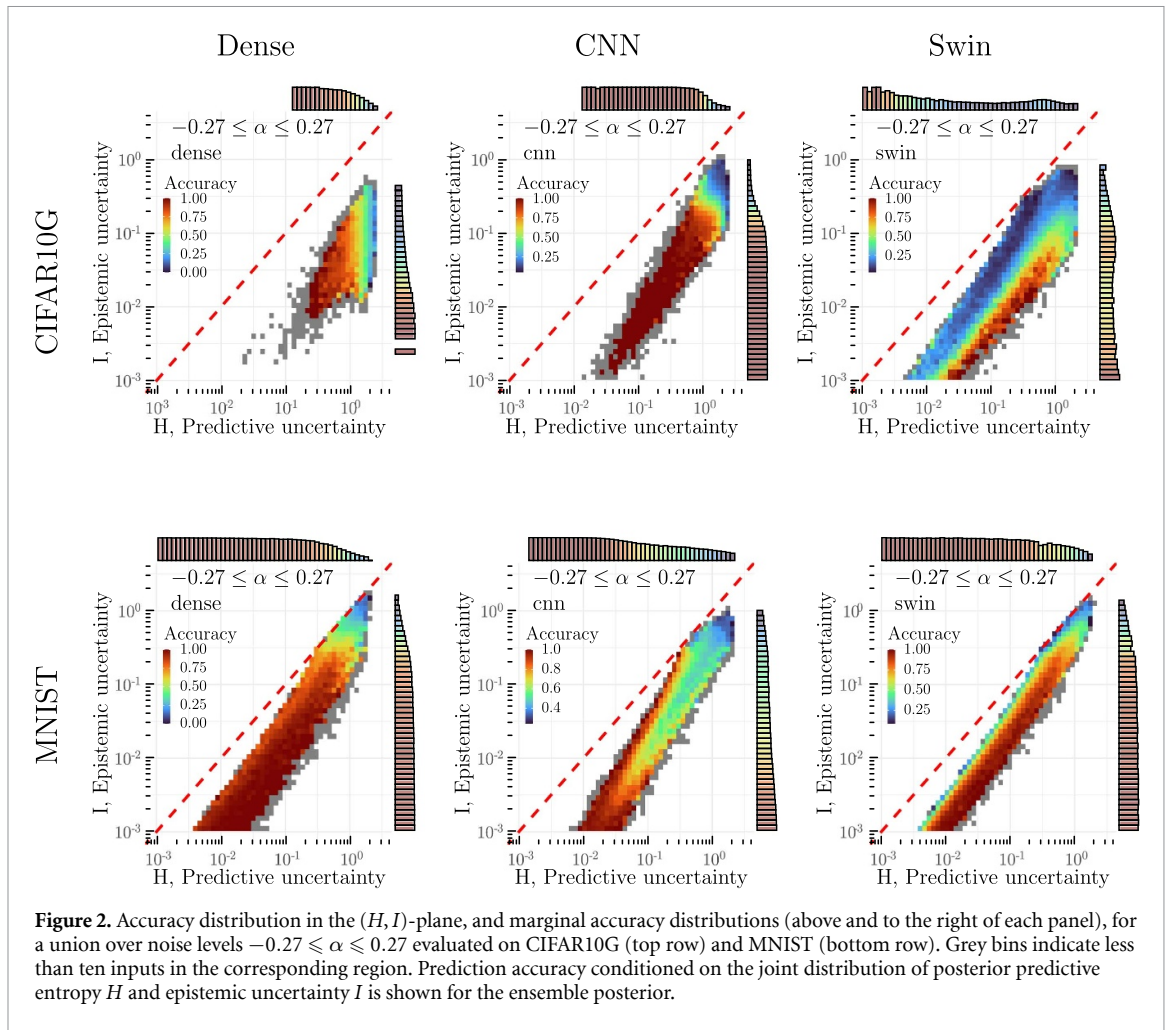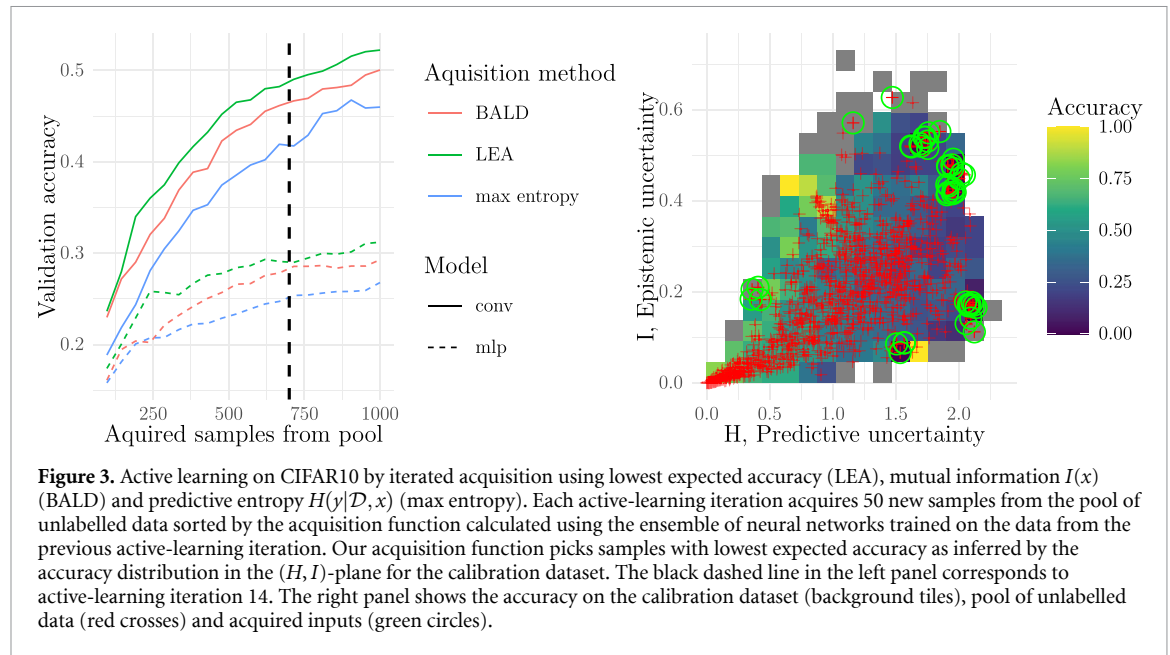| Architecture	Acc.	H	I
Dense	41%	1.66 ± 0.39	0.10 ± 0.04
CNN	20%	1.92 ± 0.22	0.39 ± 0.11
Swin	20%	1.12 ± 0.47	0.27 ± 0.14

Table 3. Mean and standard deviation for predictive and epistemic uncertainty for fully connected (Dense), convolutional (CNN) and attention-based (Swin) models with posterior approximations using ensembles for MNIST at fixed data-distributional shift $\alpha = 0.09$.

Architecture	Accuracy	H	I
Dense	65%	0.54 ± 0.47	0.35 ± 0.32
CNN	73%	0.67 ± 0.54	0.25 ± 0.22
Swin	58%	0.69 ± 0.44	0.44 ± 0.30

Table 4. Mean and standard deviation for predictive and epistemic uncertainty and accuracy for fully connected (Dense), convolutional (CNN) and attention-based (Swin) models with posterior approximations using deep ensembles evaluated on CIFAR10G without data-distributional shift.

Architecture	Acc.	H	I
Dense	45%	1.58 ± 0.41	0.07 ± 0.04
CNN	75%	1.07 ± 0.64	0.14 ± 0.10
Swin	65%	0.90 ± 0.55	0.08 ± 0.06

Figure 3. Active learning on CIFAR10 by iterated acquisition using lowest expected accuracy (LEA), mutual information $I(x)$ (BALD) and predictive entropy $H(y|\mathcal{D}, x)$ (max entropy). Each active-learning iteration acquires 50 new samples from the pool of unlabelled data sorted by the acquisition function calculated using the ensemble of neural networks trained on the data from the previous active-learning iteration. Our acquisition function picks samples with lowest expected accuracy as inferred by the accuracy distribution in the (H, I)-plane for the calibration dataset. The black dashed line in the left panel corresponds to active-learning iteration 14. The right panel shows the accuracy on the calibration dataset (background tiles), pool of unlabelled data (red crosses) and acquired inputs (green circles).

data we use the remaining CIFAR10 training dataset after the initial inputs and accuracy calibration dataset have been removed. We evaluate the active-learning scheme for three different acquisition functions: BALD (maximum mutual information), max entropy (maximum predictive entropy) and our proposed lowest expected accuracy function, LEA, described above.

Figure 3 shows the results of this procedure. The left panel shows the evolution of the accuracy on the validation set for the dense and convolution architectures, using the three different acquisition functions. LEA consistently outperforms BALD, as well as active learning based solely on the maximum predictive uncertainty. To visualize the acquisition process for a particular active-learning iteration, we also show (right panel) the accuracy on the calibration dataset (background tiles) for an example active-learning iteration, together with the pool of unlabelled data (red crosses) and the inputs acquired by the acquisition function (green circles). The panel shows that the acquired inputs are from regions in the (H, I) plane that neither maximise epistemic uncertainty nor predictive uncertainty, thus acquiring different inputs than using BALD or max entropy as acquisition function. This explains why LEA outperforms the other two acquisition functions: neither H nor I alone suffice to estimate the expected accuracy.

6.3. Uncertainty

To understand how the uncertainty distribution in figure 2 correlates with noise level, we now turn to the uncertainty density distributions at a fixed noise level. Figure 4 shows, for different neural-network architectures, the joint distribution of the predictive uncertainty defined by equation (4) and epistemic uncertainty as defined by equation (6) in the (H, I) plane. The uncertainty measures are calculated using approximations to the Bayesian posterior provided by the ensemble method as explained in section 4.2. The joint distributions in figure 4 are evaluated on the test set of CIFAR10G shifted by impulse noise with noise parameter $\alpha = 0.09$, corresponding to 9% of pixels set to the maximum grayscale intensity, see figure 1. The joint distributions in figure 4 allow us to separate the origin of uncertainty: a sample with large value of I on the vertical axis has higher epistemic uncertainty, whereas a sample on the horizontal axis has no epistemic uncertainty. The dashed red diagonal line indicates the bound on epistemic uncertainty imposed by equation (8). In particular this means that a sample close to the epistemic bound is dominated by epistemic uncertainty.

The first panel in the top row of figure 4 shows the joint distribution for the dense model. The joint distribution is shifted towards higher predictive uncertainty and lower epistemic uncertainty. The top-center panel in figure 4 shows the joint uncertainty distribution for the convolutional neural network. Here, the perceived epistemic uncertainty is larger compared to the other models. The right-most panel in figure 4 shows that the attention-based model exhibits similar epistemic uncertainty to the convolutional neural network in the center panel. The attention-based model does however perceive a lower aleatoric uncertainty, although this is not accompanied by a significant increase in accuracy.

Turning to MNIST, figure 4 shows the corresponding joint distributions for the three architectures at noise level $\alpha = 0.09$. The first panel shows joint distributions of predictive and epistemic uncertainty for the
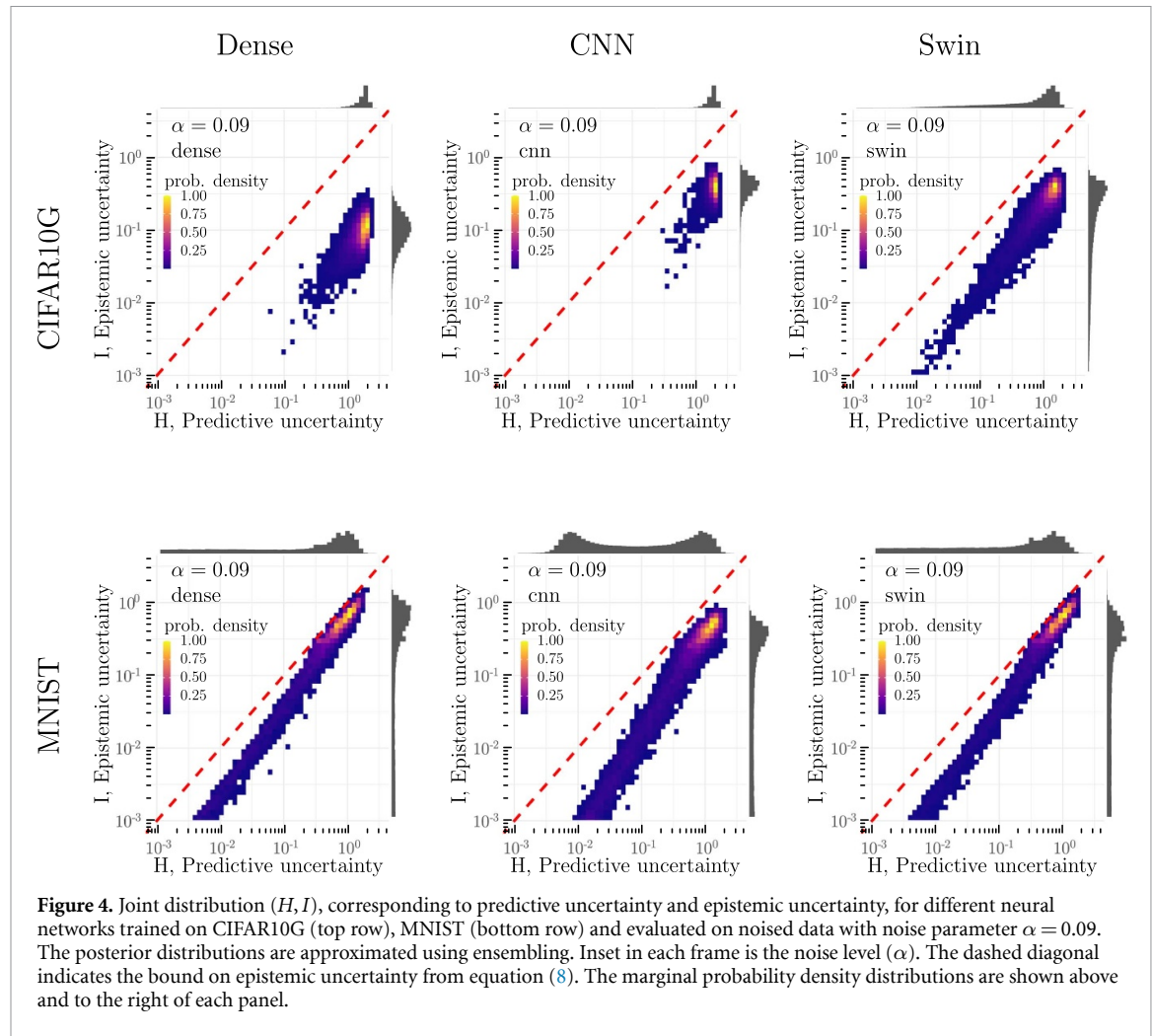
Figure 4. Joint distribution (H, I), corresponding to predictive uncertainty and epistemic uncertainty, for different neural networks trained on CIFAR10G (top row), MNIST (bottom row) and evaluated on noised data with noise parameter $\alpha = 0.09$. The posterior distributions are approximated using ensembling. Inset in each frame is the noise level (α). The dashed diagonal indicates the bound on epistemic uncertainty from equation (8). The marginal probability density distributions are shown above and to the right of each panel.

Table 5. Change in mean predictive and epistemic uncertainty, $\Delta\bar{H}$ and $\Delta\bar{I}$, for fully connected (Dense), convolutional (CNN) and attention-based (Swin) models from $\alpha = 0$ to $\alpha = 0.09$. Posterior approximation by ensembling.

Architecture	Dataset	α	$\Delta\bar{H}$	$\Delta\bar{I}$
Dense	MNIST	0.09	0.49	0.33
Dense	CIFAR	0.09	0.08	0.03
CNN	MNIST	0.09	0.65	0.24
CNN	CIFAR	0.09	0.85	0.25
Swin	MNIST	0.09	0.63	0.41
Swin	CIFAR	0.09	0.22	0.20

dense model. In contrast to the dense model on CIFAR10G, the epistemic uncertainty is larger than the other models and the predictive uncertainty is smaller. The center panel shows the joint distribution for the convolutional neural network, and the right-most panel the attention-based Swin model.

To evaluate the dependence of the uncertainty quantification on the training dataset we calculate the induced shift in joint distribution of predictive and epistemic uncertainty when we apply the same data-distributional shift to neural networks trained on MNIST and CIFAR10G. The shift from $\alpha = 0$ to $\alpha = 0.09$ in mean predictive and epistemic uncertainty is quantified in table 5. For the dense and Swin model we observe a significantly smaller shift in both H and I on CIFAR10G compared to MNIST, whereas the convolutional neural network perceives a larger shift on CIFAR10G compared to MNIST.

The dense and Swin model both contain more parameters for CIFAR10G compared to MNIST stemming from the four extra pixels in both spatial directions. One possible method to keep the parameter count constant would be to resample the resolution of the images, however this would also introduce some dependence on the resampling method. Here we keep the original resolution for simplicity and leave the possibility of resampling for future work.

7. Discussion

7.1. Accuracy

Our results allow us to draw three conclusions regarding the connection between the joint distribution of epistemic and predictive uncertainty, and prediction accuracy.

First, looking at the joint distribution helps to identify samples where a given model is more accurate. One important goal of uncertainty quantification is to assess when the predictions of a model can be trusted. It is natural to expect that a prediction is more accurate when the predictive uncertainty is low. We show that the joint distribution of epistemic and predictive uncertainty can identify accurate predictions while predictive uncertainty or epistemic uncertainty alone cannot. This can be seen in figure 2, where the joint distribution clearly resolves where the models are accurate, and where they are not. Figure 2 show that the projection of the joint distribution on either the axis of predictive uncertainty or the axis of epistemic uncertainty mixes samples with high and low accuracy. As a consequence, selecting for predictive uncertainty or epistemic uncertainty alone, is not as effective in identifying where a given model is more accurate. This explains the findings in e.g. [20, 52], regarding the clustering of incorrect predictions at high uncertainty, and also explains why it is difficult to achieve higher precision using the marginalized measures of epistemic and predictive uncertainty, as observed in [22]: samples with small values for the marginalized distributions can still have accurate model predictions.

Second, for the attention-based model it is difficult to find a threshold of the projected distribution on either predictive uncertainty or epistemic uncertainty that results in high accuracy. The top-right panels in figure 2 show a distinct structure for the accuracy conditioned on the joint distribution where the marginalized distribution on either axis mixes samples of high and low accuracy. This has implications for active learning. In the present example, choosing samples based on e.g. posterior predictive uncertainty alone is inefficient, because it results in training on data regions where the model is already accurate.

Third, by training on data regions where the accuracy is low, there is potential to increase overall prediction accuracy. The joint distribution of epistemic and predictive uncertainty can be used to identify such data regions where the model accuracy can be improved, also for data that is not yet labeled. Figure 2 show that the regions are specific to each network architecture, and to the dataset used for training. The mixing of high and low accuracy prediction when projecting the distributions in figure 2 also explain the difficulty in achieving better sample efficiency by using the decomposition into aleatoric and epistemic uncertainty observed in [13]: the projected distributions will contain regions of high prediction accuracy for small uncertainties. An active learning scheme that makes use of uncertainty quantification could therefore benefit from selection using the joint distribution.

7.2. Active learning

Since prediction accuracy depends on both epistemic and predictive uncertainty, we proposed an acquisition function for active learning that uses a parameterization of expected accuracy in terms of both conditional mutual information and predictive entropy. Our proposed LEA (lowest expected accuracy) acquisition function, defined in section 5.3, outperforms acquisition functions using marginal uncertainty distributions. The left panel in figure 3 shows that our calibrated accuracy acquisition produce samples that enable the neural networks achieve higher validation accuracy with less training samples. The right panel in figure 3 shows that LEA picks inputs from regions that would have been missed by BALD (acquired inputs do not maximise mutual information) and max entropy (acquired inputs do not maximise predictive entropy). We stress that the proposed acquisition function only considers single sample uncertainty, we expect that e.g accuracy gain per iteration can be further improved by incorporating expected accuracy in state of the art acquisition functions such as BatchBALD [40] or EPIG [53]. Also note that the comparison with BALD and max entropy in figure 3 can be argued to be unfair in the sense that LEA uses information about the targets from the calibration dataset. This could be amended by calibrating on the current training dataset instead of a held-out calibration dataset.

7.3. Uncertainty

We can draw two conclusions from our results regarding how uncertainty quantification depends on model architecture.

First, the origin of uncertainty is not objective. Table 4 shows that there are significant differences in the perceived origin of uncertainty between the different model architectures for in-domain data. For CIFAR10G, the fully connected neural network perceives a higher degree of aleatoric uncertainty, and low epistemic uncertainty compared to the convolutional and attention-based models. This shows that the origin of uncertainty depends on the model architecture: the fully connected neural network struggles to express the relationship between inputs and classes of CIFAR10G, and thus perceives a higher degree of aleatoric

uncertainty. Even though the model has low accuracy, the ensemble posterior approximation show a low epistemic uncertainty. Thus we do not expect to be able to salvage the accuracy of the fully connected neural network by increasing the size of the dataset. For CIFAR10G, the top row of figure 4 and table 2 show that for a moderate data-distributional shift, the fully connected neural network perceives a higher degree of aleatoric uncertainty and low epistemic uncertainty compared to the convolutional and attention-based models. Here the dense architecture is already perceiving the validation data as aleatoric noise and thus continues to do so for further shifts.

Second, the different model posteriors agree about the origin of uncertainty when the dataset complexity is low. In the case of MNIST, the bottom row of figure 4 and table 3 show that all neural-network architectures considered here have similar joint distributions of predictive and epistemic uncertainty. In other words, the MNIST data has relatively low complexity, as evident by the higher average accuracies in table 1, and therefore all three architectures succeed in capturing the important features. As a consequence, all three architectures agree about the origin of uncertainty.

In addition, there are two conclusions we can draw from our results regarding how predictive and epistemic uncertainty depend on the dataset. First, the relative change in perceived uncertainty under a fixed data-distributional shift depends on the dataset, and varies with model architecture. Comparing the change in the joint distribution under a fixed data-distributional shift in table 5, we observe that a given model architecture understands the same data-distributional shift in different ways, depending on the underlying training dataset. There is an asymmetry in this shift sensitivity: only the convolutional model shows a stronger sensitivity of the predictive and epistemic uncertainty when evaluated on CIFAR10G compared to MNIST. Furthermore, the fully connected model perceives the impulse noise on CIFAR10G very differently from impulse noise on MNIST. The distributional shift for MNIST induces a large change in both epistemic and predictive uncertainty, whereas for CIFAR10G the induced shift in uncertainty is significantly smaller. This can be explained by CIFAR10G containing realistic digital images, hence we expect impulse noise to be more in-domain than for MNIST. The converse is true for the convolutional model, where the induced shift is larger for CIFAR10G. In summary, the sensitivity to a particular data-distributional shift depends on the data domain and on neural-network architecture. Even though this difference in perceived relative change of uncertainty is present for widely different data, it also has implications for different domains in the same training dataset, something that would be interesting to quantify in more detail.

Second, robustness of prediction accuracy under data-distributional shifts for a given model depends on the dataset. For CIFAR10G, the higher accuracy of the fully connected neural network on noised data in table 2 shows that this architecture is more robust against this particular distributional shift, even though the model is less accurate close to the training domain as seen in table 4. For MNIST, the convolutional model is instead more robust than both the dense and attention-based architectures as seen in table 3.

8. Conclusions

Posterior predictive entropy and mutual information are used extensively as measures of predictive uncertainty and epistemic uncertainty to assess the uncertainty and performance of neural networks and their predictions [2]. We introduced the joint distribution of predictive uncertainty and epistemic uncertainty and demonstrated how it is related to model accuracy.

Previous work have shown that in both active learning [16, 26] and predictive uncertainty estimation [20, 21], it is difficult to formulate a general strategy making use of a decomposition of uncertainty into predictive and epistemic parts. Our results explain why it is difficult to use predictive or epistemic uncertainty separately as a measure of model efficacy by showing how the joint distribution resolves information that is lost in projections. We showed that the joint distribution contains information about when a neural network is accurate, and that the distribution is specific to the particular choice of neural-network architecture and dataset.

To test these insight, we proposed a novel acquisition function using expected accuracy parameterized in terms of epistemic and predictive uncertainty. The proposed acquisition function outperforms two common acquisition functions based on marginal uncertainty distributions.

In addition, we also demonstrated that the origin of uncertainty is not objective: different model architectures will disagree about the aleatoric and epistemic uncertainty. Furthermore, for a given model, the sensitivity of the uncertainty quantification to a specific type of data-distributional shift depends on the underlying training dataset.

We conclude by mentioning the most important open questions. First, is it possible to explain how uncertainty quantification depends on architecture from the mathematical theory of neural networks, and how to use this to build architectures that target robust uncertainty quantification. Second, given the recent rise of attention-based architectures across multiple domains such as natural language processing and

computer vision, it is of utmost importance to properly understand their posteriors and related uncertainty measures. Third, in practical application, uncertainty quantification using the Bayesian posterior depends on accurate posterior approximations. Here we used ensembles as a robust baseline, but there is a growing need for accurate and computationally effective posterior approximation methods. Finally, entropy and mutual information are two measures of uncertainty derived from the high-dimensional model posterior and the posterior predictive distribution. Whether there might be other, complementary or more informative, derived quantities that can capture the uncertainty of artificial neural networks better also remains an interesting question.

In summary, the joint distribution of predictive and epistemic uncertainty can inform on neural-network efficacy when calibrated for a given dataset and architecture.

Data availability statement

No new data were created or analysed in this study.

Acknowledgments

Appendix A. Ensemble posterior

The Bayesian mean in equation (1) can be approximated by Monte-Carlo sampling of the integration over model parameters. Given N parameter samples $\theta^{(i)}$ of equal posterior probability, the Bayesian mean can be approximated as in equation (A.1).

$$p(y|x,\mathcal{D}) = \int_\theta f(y,x;\theta)\, p(\theta|\mathcal{D})\, \mathrm{d}\theta \approx \frac{1}{N}\sum_{i=0}^{N} f\left(y,x;\theta^{(i)}\right). \tag{A.1}$$

A frequently employed method to sample from the model space is to train an ensemble [34, 38] of N identical neural networks using different initial parameter values $\theta^{(i)}_{\mathrm{initial}}$ and regularized by quadratic loss on parameter norm corresponding to the Gaussian prior. Training these neural networks to maximize the likelihood of the training data gives a set of parameters $\theta^{(i)}$, that then provides an approximation to the Bayesian mean by equation (A.2), where $\theta^{(i)}$ are the ensemble member parameters.

$$p(y=c|x,\mathcal{D}) \approx \frac{1}{N}\sum_{i=0}^{N} f_c\left(x;\theta^{(i)}\right). \tag{A.2}$$

Here it is assumed that the likelihood of the minima attained by $\theta^{(i)}$ are equally probable. Note that it is not clear *a priori* whether the minima $\theta^{(i)}$ are degenerate, but for the regime of neural networks for visual perception it is typically the case that they are not [54].

Appendix B. Toy model posterior

The relation between mutual information of the parameter posterior and mutual information of the posterior predictive in equation (6) provides a way of calculating the expected change in entropy of the high-dimensional parameter posterior $p(\theta|\mathcal{D})$ in terms of the typically lower-dimensional posterior predictive distribution $p(y|x,\mathcal{D})$ and the likelihood $p(y|x,\theta)$. To illuminate this relation, and evaluate the involved quantities in closed-form, we present a detailed verification for a simple toy model. This also serves as an illustration of the computational complexity involved in computing the Bayesian posterior directly.

The toy problem consists of classifying points on the real line into two classes c_1, c_2 and we use a simple two-parameter linear model

$$p(c_1|x,\theta_1,\theta_2) = \begin{cases} 1 & x-\theta_1 < -\theta_2, \\ \frac{(x-\theta_1+\theta_2)}{2\theta_2} & |x-\theta_1| < \theta_2, \\ 0 & x-\theta_1 > \theta_2, \end{cases} \tag{B.1}$$
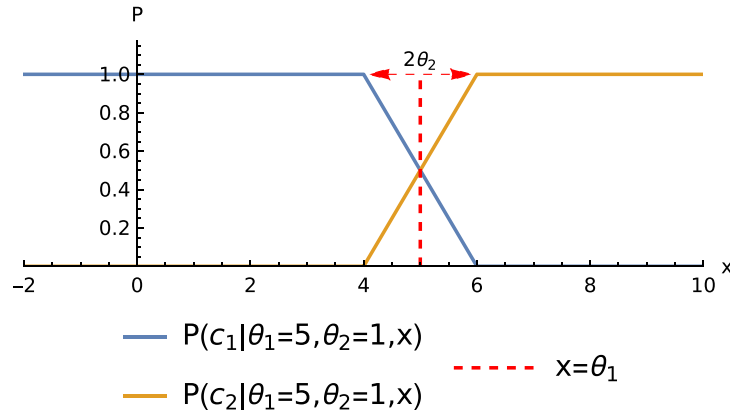
Figure B1. Example of the likelihood $p(c|\theta_1,\theta_2,x)$ in equations (B.1) and (B.2) over the two classes $c \in \{c_1, c_2\}$ for different values of x with model parameters $\theta_1 = 5$ and $\theta_2 = 1$ corresponding to the decision boundary.
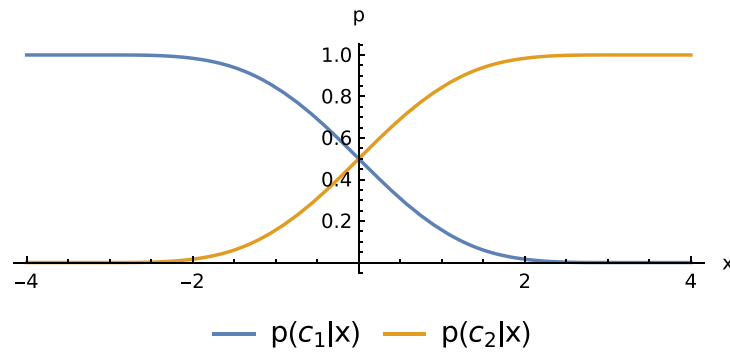

Figure B2. Prior class distribution given the uniform prior $p(\theta_1, \theta_2)$ on the model parameters.

$$p(c_2|x,\theta_1,\theta_2) = 1.0 - p(c_1|x,\theta_1,\theta_2). \tag{B.2}$$

By construction, this model has a strong prior for samples of class 1 being located to the left of a decision boundary at θ_1 and class 2 to the right. See figure B1 for the resulting probability distributions for a particular choice of the model parameters θ_1 and θ_2.

Let θ_1 be uniformly distributed on $[-1, 1]$ and θ_2 on $[\frac{1}{2}, 2]$. With this prior on the parameters, we can calculate the prior predictive distribution over the two classes by

$$p(c|x) = \int_{\theta_1}\int_{\theta_2} p(c|x,\theta_1,\theta_2) p(\theta_1,\theta_2) \, d\theta_1 d\theta_2, \tag{B.3}$$

visualized in figure B2, where we see that the prior parameter distribution results in a smooth prior predictive distribution.
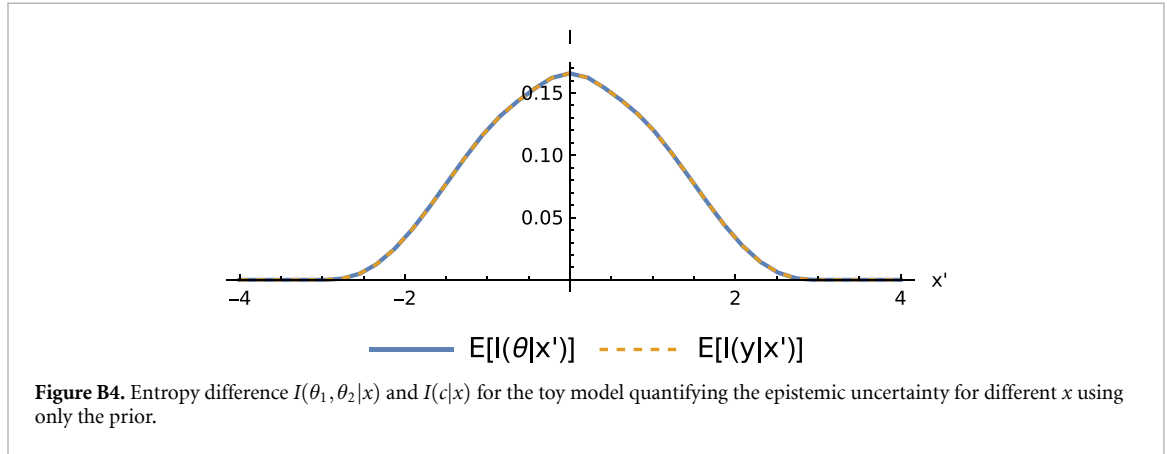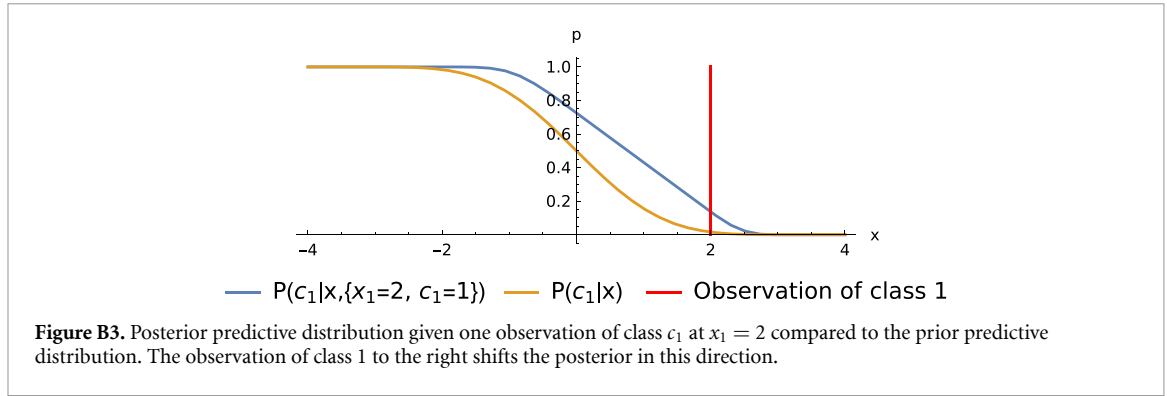
Suppose we observe class c_1 at $x_1 = 2$, we can then calculate a posterior distribution for the model parameters

$$p(\theta_1,\theta_2 | \{x_1, c_1\}) = \frac{p(c_1|x_1,\theta_1,\theta_2) p(\theta_1,\theta_2)}{p(c_1|x_1)} \tag{B.4}$$

where we have used Bayes theorem to express the parameter posterior in terms of conditional probabilities that can be calculated explicitly.

With this posterior we calculate the posterior predictive distribution in equation (1) of the introduction, resulting in a slightly shifted distribution for class 1 in figure B3, compared to the class prior in figure B2.

Using the toy model we can now explicitly verify the relation between equations (5) and (6). The expected entropy difference in equation (5) is given by

Figure B3. Posterior predictive distribution given one observation of class c_1 at $x_1 = 2$ compared to the prior predictive distribution. The observation of class 1 to the right shifts the posterior in this direction.

Figure B4. Entropy difference $I(\theta_1, \theta_2 | x)$ and $I(c|x)$ for the toy model quantifying the epistemic uncertainty for different x using only the prior.

$$I(\theta_1, \theta_2 | x) = \int_{\theta_1} \int_{\theta_2} p(\theta_1, \theta_2) \log(p(\theta_1, \theta_2)) \, d\theta_1 d\theta_2$$
$$- \sum_{i=1,2} p(c_i | x) \int_{\theta_1} \int_{\theta_2} p(\theta_1, \theta_2 | \{x_1, c_i\}) \log(p(\theta_1, \theta_2 | \{x_1, c_i\})) \, d\theta_1 d\theta_2 \qquad (B.5)$$

and the posterior predictive entropy difference in equation (6) is given by

$$I(c|x) = \sum_{i=1,2} p(c_i | x) \log(p(c_i | x))$$
$$- \int_{\theta_1} \int_{\theta_2} \left(\sum_{i=1,2} p(c_i | x, \theta_1, \theta_2) \log(p(c_i | x, \theta_1, \theta_2)) \right) p(\theta_1, \theta_2) \, d\theta_1 d\theta_2. \qquad (B.6)$$

Note that in this case, where we compute the entropy difference when adding a single observation equation (B.6) only uses the prior distribution.

Numerically evaluating these expressions gives figure B4 where the two curves are indistinguishable, as expected.

Figure B4 shows that the epistemic uncertainty is largest close to the decision boundary of the prior. This can be understood intuitively by the fact that the model and parameter priors are such that adding observations of class 1 far to the left (or class 2 far to the right) does not add new information.

Continuing, we can perform the same calculation but instead add a new observation on top of the first one. Assuming independent observations the posterior now becomes

$$p(\theta_1, \theta_2 | (x_1, c_1), (x_2, c_2)) = \frac{p(c_1 | x_1, \theta_1, \theta_2) \, p(c_2 | x_2, \theta_1, \theta_2) \, p(\theta_1, \theta_2)}{p((x_1, c_1), (x_2, c_2))} \qquad (B.7)$$

and carefully calculating the entropy differences now instead results in the epistemic uncertainty in figure B5. Note first that the two expressions are still in excellent agreement. The observation of class 1 at $x = 2$ is in tension with the prior which can be seen by the bi-modal epistemic uncertainty.

The epistemic uncertainty can be compared to the entropy of the posterior predictive distribution in figure B6 which peaks in the region between the prior decision boundary and the observed class 1 sample.
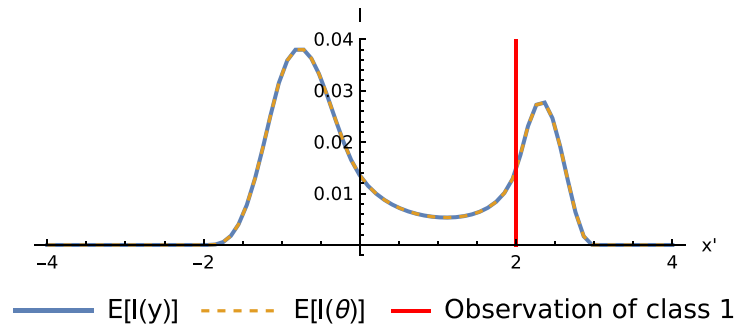
Figure B5. Entropy difference $I(\theta_1, \theta_2 | x, \{x_1 = 2, c = 1\})$ and $I(c|x, \{x_1 = 2, c = 1\})$ for the toy model quantifying the epistemic uncertainty for different x after a single observation $\{x = 2, c = 1\}$.
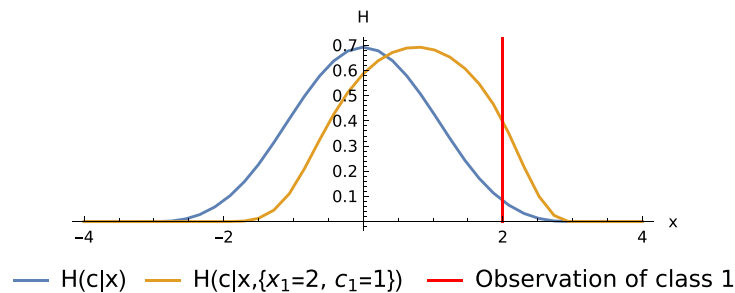

Figure B6. Entropy of the posterior predictive distribution using the prior $H(c|x)$ and after one observation $H(c|x, \{x_1 = 2, c = 1\})$ for the toy model quantifying aleatoric and epistemic uncertainty for different x.

ORCID iDs

Hampus Linander ⬤ https://orcid.org/0000-0002-7865-8414
Oleksandr Balabanov ⬤ https://orcid.org/0000-0003-0417-9856
Bernhard Mehlig ⬤ https://orcid.org/0000-0002-3672-6538

References

[1] Gawlikowski J *et al* 2023 A survey of uncertainty in deep neural networks *Artif. Intell. Rev.* **56** 1513–89
[2] Abdar M *et al* 2021 A review of uncertainty quantification in deep learning: techniques , applications and challenges *Inf. Fusion* **76** 243–97
[3] Gustafsson F K, Danelljan M and Schon T B 2020 Evaluating scalable Bayesian deep learning methods for robust computer vision *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops* pp 318–9
[4] Xin Li and Guo Y 2013 Adaptive active learning for image classification *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 859–66
[5] Lee J-G, Jun S, Cho Y-W, Lee H, Bae Kim G, Beom Seo J and Kim N 2017 Deep learning in medical imaging: general overview *Korean J. Radiol.* **18** 570–84
[6] Mervin L H, Johansson S, Semenova E, Giblin K A and Engkvist O 2021 Uncertainty quantification in drug design *Drug Discovery Today* **26** 474–89
[7] Lye K O, Mishra S and Ray D 2020 Deep learning observables in computational fluid dynamics *J. Comput. Phys.* **410** 109339
[8] Liu Y, Qin H, Zhang Z, Pei S, Jiang Z, Feng Z and Zhou J 2020 Probabilistic spatiotemporal wind speed forecasting based on a variational Bayesian deep learning model *Appl. Energy* **260** 114259
[9] Butler K T, Manh Duc L, Thiyagalingam J and Perring T G 2021 Interpretable, calibrated neural networks for analysis and understanding of inelastic neutron scattering data *J. Phys.: Condens. Matter* **33** 194006
[10] Neal R M 2012 *Bayesian Learning for Neural Networks* vol 118 (Springer)
[11] David J C M 1992 Information-based objective functions for active data selection *Neural Comput.* **4** 590–604
[12] Der Kiureghian A and Ditlevsen O 2009 Aleatory or epistemic? Does it matter? *Struct. Safety* **31** 105–12
[13] Beluch W H, Genewein T, Nürnberger A and Köhler J M 2018 The power of ensembles for active learning in image classification *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 9368–77
[14] Nguyen V-L, Destercke S and Hüllermeier E 2019 Epistemic uncertainty sampling *Int. Conf. on Discovery Science* (Springer) pp 72–86
[15] Gal Y, Islam R and Ghahramani Z 2017 Deep bayesian active learning with image data *Int. Conf. on Machine Learning* (PMLR) pp 1183–92
[16] Chitta K, Alvarez J M and Lesnikowski A 2018 Large-scale visual active learning with deep probabilistic ensembles (arXiv:1811.03575)

[17] Ruhe D, Cina G, Tonutti M, de Bruin D and Elbers P 2019 Bayesian modelling in practice: Using uncertainty to improve trustworthiness in medical applications (arXiv:1906.08619)

[18] Shi H, Worrall D, Knegt S, Veeling B, Huisman H and Welling M 2019 Supervised uncertainty quantification for segmentation with multiple annotations *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention* (Springer) pp 137–45

[19] Reinhold J C, Yufan H, Han S, Chen Y, Gao D, Lee J, Prince J L and Carass A 2020 Validating uncertainty in medical image translation *2020 IEEE 17th Int. Symp. on Biomedical Imaging (ISBI)* (IEEE) pp 95–98

[20] Ghoshal B, Tucker A, Sanghera B and Lup Wong W 2021 Estimating uncertainty in deep learning for reporting confidence to clinicians in medical image segmentation and diseases detection *Comput. Intell.* **37** 701–34

[21] Ghoshal B and Tucker A 2020 Estimating uncertainty and interpretability in deep learning for coronavirus (covid-19) detection (arXiv:2003.10769)

[22] Nair T, Precup D, Arnold D L and Arbel T 2020 Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation *Medical Image Anal.* **59** 101557

[23] Awad A 2019 Denoising images corrupted with impulse, Gaussian, or a mixture of impulse and Gaussian noise *Eng. Sci. Technol. Int. J.* **22** 746–53

[24] Lecun Y, Bottou L, Bengio Y and Haffner P 1998 Gradient-based learning applied to document recognition *Proc. IEEE* **86** 2278–324

[25] Krizhevsky A 2009 Learning multiple layers of features from tiny images *Tech. Report* (University of Toronto)

[26] Depeweg S, Hernandez-Lobato J-M, Doshi-Velez F and Udluft S 2018 Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning *Proc. 35th Int. Conf. on Machine Learning* (*Proc. of Machine Learning Research* vol 80) ed J Dy and A Krause (PMLR) pp 1184–93

[27] Ovadia Y, Fertig E, Ren J, Zachary Nado D S, Nowozin S, Dillon J V, Lakshminarayanan B and Snoek J 2019 *Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift* (Curran Associates Inc.)

[28] Hendrycks D and Dietterich T 2019 Benchmarking neural network robustness to common corruptions and perturbations (arXiv:1903.12261)

[29] Mehlig B 2021 *Machine Learning With Neural Networks* (Cambridge University Press)

[30] Sivia D and Skilling J 2006 *Data Analysis: A Bayesian Tutorial* (OUP Oxford)

[31] Izmailov P, Vikram S, Hoffman M D and Gordon Gordon Wilson A 2021 What are Bayesian neural network posteriors really like? *Int. Conf. on Machine Learning* (PMLR) pp 4629–40

[32] Kendall A and Gal Y 2017 What uncertainties do we need in Bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems* p 30

[33] Balabanov O, Mehlig B and Linander H 2023 Bayesian posterior approximation with stochastic ensembles *2023 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (IEEE) pp 13701–11

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

[35] Duane S, Kennedy A D, Pendleton B J and Roweth D 1987 Hybrid Monte Carlo *Phys. Lett.* B **195** 216–22

[36] Mehlig B, Heermann D W and Forrest B M 1992 Hybrid monte carlo method for condensed-matter systems *Phys. Rev.* B **45** 679

[37] Radford M N 1996 *Monte Carlo Implementation* (Springer) pp 55–98

[38] Hoffmann L and Elster C 2021 Deep ensembles from a Bayesian perspective (arXiv:2105.13283)

[39] Shannon C E 1948 A mathematical theory of communication *Bell Syst. Tech. J.* **27** 379–423

[40] Kirsch A, Van Amersfoort J and Gal Y 2019 Batchbald: efficient and diverse batch acquisition for deep Bayesian active learning *Advances in Neural Information Processing Systems* vol 32 pp 7026–37

[41] Wimmer L, Sale Y, Hofman P, Bischl B and Hüllermeier E 2023 Quantifying aleatoric and epistemic uncertainty in machine learning: Are conditional entropy and mutual information appropriate measures? *Uncertainty in Artificial Intelligence* (PMLR) pp 2282–92

[42] Tobias Springenberg J, Dosovitskiy A, Brox T and Riedmiller M 2014 Striving for simplicity: the all convolutional net (arXiv:1412.6806)

[43] Liu Z, Lin Y, Cao Y, Han H, Wei Y, Zhang Z, Lin S and Guo B 2021 Swin transformer: Hierarchical vision transformer using shifted windows *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)* pp 10012–22

[44] Dagli R 2021 Image classification with swin-transformers (available at: https://github.com/keras-team/keras-io/blob/afcfff54d01ead91b57ea82b0cc2d99b1f046ee4/examples/vision/swin_transformers.py)

[45] Angluin D 1988 Queries and concept learning *Mach. Learn.* **2** 319–42

[46] Atlas L, Cohn D and Ladner R 1989 Training connectionist networks with queries and selective sampling *Advances in Neural Information Processing Systems* vol 2

[47] Liu P, Wang L, Ranjan R, Guojin H and Zhao L 2022 A survey on active deep learning: from model driven to data driven *ACM Comput. Surv.* **54** 1–34

[48] Settles B 2012 *Active Learning Synthesis Lectures on Artificial Intelligence and Machine Learning* (Morgan & Claypool Publishers)

[49] Martinez-Cantin R 2014 Bayesopt: a bayesian optimization library for nonlinear optimization, experimental design and bandits *J. Mach. Learn. Res.* **15** 3735–9

[50] Houlsby N, Huszár F, Ghahramani Z, and Lengyel M'e 2011 Bayesian active learning for classification and preference learning (arXiv:1112.5745)

[51] Lewis D D 1995 A sequential algorithm for training text classifiers: corrigendum and additional data *Acm Sigir Forum* vol 29 (ACM) pp 13–19

[52] Krishnan R and Tickoo O 2020 Improving model calibration with accuracy versus uncertainty optimization *Advances in Neural Information Processing Systems* vol 33 pp 18237–48

[53] Bickford Smith F, Kirsch A, Farquhar S, Gal Y, Foster A and Rainforth T 2023 Prediction-oriented bayesian active learning *Int. Conf. on Artificial Intelligence and Statistics* (PMLR) pp 7331–48

[54] Fort S and Jastrzebski S 2019 Large scale structure of neural network loss landscapes *Advances in Neural Information Processing Systems* vol 32