Multivariate Deep Evidential Regression

Nis Meinert
German Aerospace Center (DLR)
nis.meinert@dlr.de

Alexander Lavin
Institute for Simulation Intelligence
lavin@simulation.science

Abstract

There is significant need for principled uncertainty reasoning in machine learning systems as they are increasingly deployed in safety-critical domains. A new approach with uncertainty-aware neural networks (NNs), based on learning evidential distributions for aleatoric and epistemic uncertainties, shows promise over traditional deterministic methods and typical Bayesian NNs, yet several important gaps in the theory and implementation of these networks remain. We discuss three issues with a proposed solution to extract aleatoric and epistemic uncertainties from regression-based neural networks. The approach derives a technique by placing evidential priors over the original Gaussian likelihood function and training the NN to infer the hyperparameters of the evidential distribution. Doing so allows for the simultaneous extraction of both uncertainties without sampling or utilization of out-of-distribution data for univariate regression tasks. We describe the outstanding issues in detail, provide a possible solution, and generalize the deep evidential regression technique for multivariate cases.

1 Introduction

Using neural networks (NNs) for regression tasks is one of the main applications of modern machine learning. Given a dataset of $(\vec{x}_i, \vec{y}_i)$ pairs, the typical objective is to train a NN $f(\vec{x}_i|\boldsymbol{w})$ w.r.t. $\boldsymbol{w}$ such that a given loss $\mathcal{L}(\vec{x}_i, \vec{y}_i)$ becomes minimal for each $(\vec{x}_i, \vec{y}_i)$ pair. Traditional regression-based NNs are designed to output the regression target, a.k.a., the prediction for $\vec{y}_i$, directly which allows a subsequent minimization, for example of the sum of squares:

$$\min_{\boldsymbol{w}} \sum_i \mathcal{L}(\vec{x}_i, \vec{y}_i) = \min_{\boldsymbol{w}} \sum_i \underbrace{\left(\vec{y}_i - f(\vec{x}_i|\boldsymbol{w})\right)^2}_{\mathcal{L}_i(\boldsymbol{w})} . \tag{1}$$

Technically, this is nothing but a fit of a model f, parameterized with $\boldsymbol{w}$, w.r.t. $\sum_i \mathcal{L}_i$ to data. As with any fit, the model has to find a balance between being too specific (over-fitting) and being too general (under-fitting). In machine learning this balance is typically evaluated by analyzing the trained model on a separated part of the given data which was not seen during training. In practice, no model will be able to describe this evaluation sample perfectly and deviations can be categorized into two groups: *aleatoric* and *epistemic* uncertainties [1]. The former quantifies system stochasticity such as observation and process noise, and the latter is model-based or subjective uncertainty due to limited data.

In the following we will describe and analyze an approach to reliably estimate these kinds of uncertainties for NNs by modifying the architecture and introducing an appropriate loss function. The structure of this paper is as follows: First, we will briefly discuss aleatoric and epistemic uncertainties using a pseudo example. We then give an overview of the proposed solution of Amini et al. [2]. In Sec. 2 we describe several issues with the prior work, and follow with a possible solution in Sec. 3. Finally, in Sec. 4 we summarize our multivariate generalization approach extending the prior work, which we use throughout the text.

Preprint. Under review.

1.1 Aleatoric and epistemic uncertainty

In Fig. 1 we show data located at $x = \{0, 1, 3\}$ where for each value of x_i multiple measurements, $\vec{y}_i$, were taken. We generated these data by sampling from a normal distribution centered at the dashed line referred to as the ground truth (GT). The model (solid line) represents the prediction for different values of x_i. The uncertainty in data is low for $x = \{0, 1\}$ and large for $x = 3$, leading to a low aleatoric uncertainty at the former points and a high aleatoric uncertainty at the latter where there is high variance in the observed data. Similarly, the epistemic uncertainty is low at $x = \{0, 3\}$ where predictions are close to the observed data, and large at $x = 1$ where the model poorly fits the observed data. In general, aleatoric uncertainty is related to the noise level in the data and, typically, does not depend on the sample size – only the shape of this uncertainty becomes sharper with increasing sample size. In contrast, epistemic uncertainty does scale with the sample size, and either allows the model to be pulled towards the observed distribution at $x = 1$ if only the sample size in this region is increased, or allows the fit of a more complex model and thus decreasing under-fitting in general.

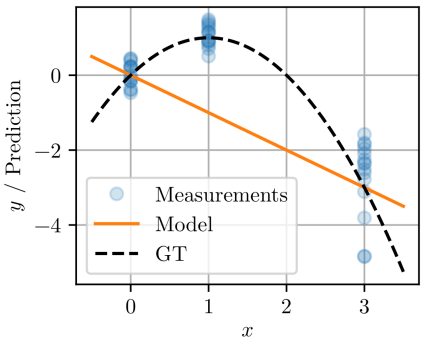


Figure 1: A fit of a model to toy data. The model has low aleatoric uncertainty at $x = \{0, 1, 2\}$ and large aleatoric uncertainty at $x = 3$, whereas the epistemic uncertainty is low at $x = \{0, 3\}$ and large elsewhere.

Pivotal for this work is the point $x = 2$ (and, technically, all other points $\mathbb{Z} \setminus \{0, 1, 3\}$) since no data were observed here. Having no data also corresponds to an epistemic uncertainty since it decreases, in theory, if more data are drawn, assuming a conclusive data sample. In contrast to the large epistemic uncertainty at $x = 1$ this uncertainty is hard to detect by evaluating a trained model, but at the same time it can be crucial for models to communicate this type of uncertainty in real-world applications such as autonomous driving [3, 4, 5], where models can easily be confronted with out-of-distribution data that was underrepresented during training, leading to dangerous and expensive failures.

1.2 Deep Evidential Regression

Different approaches have been developed to enable models to estimate either aleatoric or epistemic uncertainty, where the latter often requires out-of-distribution data or compute-intense sampling, limiting the application of such approaches [6, 7, 8, 9]. Recently, Amini et al. adopted a technique from the classification realm and attacked this problem by changing the interpretation of the parameters of the NN [2, 10]: The number of output neurons of a NN for a univariate regression task has to be increased from one to four. The output of these neurons are interpreted as α, β, μ_0, and $\kappa \in \mathbb{R}$.[1] These are the parameters of a normal-inverse-gamma function $\texttt{NIG}(\mu_0, \kappa;\, \alpha, \beta)$, and used to estimate the prediction and both uncertainties as:

$$\underbrace{\mathbb{E}[\mu] = \mu_0}_{\text{prediction}} \qquad \underbrace{\mathbb{E}[\sigma^2] = \beta/(\alpha - 1)}_{\text{aleatoric}} \qquad \underbrace{\text{var}[\mu] = \mathbb{E}[\sigma^2]/\kappa}_{\text{epistemic}} \tag{2}$$

The authors derive these relations by taking the normal-inverse-gamma distribution (NIG) as the conjugated prior of a normal distribution with unknown mean μ and standard deviation σ. Further, the authors show that by using Bayesian inference the loss function for these four parameters becomes a scaled Student's t-distribution with 2α degrees of freedom (DoF), parametrized as:

$$\mathcal{L}_i^{\text{NLL}} = \text{St}_{2\alpha}\left(y_i \middle| \mu_0, \frac{\beta(1+\kappa)}{\kappa\alpha}\right). \tag{3}$$

For reasons we will discuss later, they combine it with a second loss function, referred to as the *evidence regularizer*, using the *total evidence* Φ, yielding the total loss:

$$\mathcal{L}_i(\boldsymbol{w}) = \mathcal{L}_i^{\text{NLL}}(\boldsymbol{w}) + \lambda \times |y_i - \mu_0|\Phi\,, \tag{4}$$

[1] In [2] the authors refer to them as α, β, γ and ν, respectively.

where the coupling, λ, is a hyperparameter of the model. Note that, following the notation of [2], we have dropped indices for all parameters for the sake of brevity – see Appendix A for a more elaborated discussion.

2 Addressing issues in the prior art

In this section we discuss three issues with the prior work on Deep Evidential Regresstion [2]. We also describe new multivariate formulations, which will be detailed later in Sec. 4.

2.1 Definition of total evidence

In Bayesian inference a normal-inverse-Wishart distribution (NIW) is a conjugate prior for i.i.d. drawn events from a multivariate normal distribution with unknown mean $\vec{\mu} \in \mathbb{R}^n$ and unknown variance $\mathbf{\Sigma} \in \mathbb{R}^{n\times n}$ [11]. In the univariate case, $n = 1$, a NIW distribution becomes a NIG distribution and we slightly change our notation and use $\mu \in \mathbb{R}$ and $\sigma^2 \in \mathbb{R}$ for the (unknown) mean and the (unknown) variance, respectively. These prior densities,

$$p(\mu, \sigma^2) = \texttt{NIG}(\mu_0, \kappa;\, \alpha, \beta) \qquad\qquad p(\vec{\mu}, \mathbf{\Sigma}) = \texttt{NIW}(\vec{\mu}_0, \kappa;\, \mathbf{\Psi}, \nu) \tag{5}$$

with[2] $\mu_0, \kappa, \alpha, \beta, \nu \in \mathbb{R}$, $\vec{\mu}_0 \in \mathbb{R}^n$ and $\mathbf{\Psi} \in \mathbb{R}^{n\times n}$, correspond to the assumption that each pair (μ, σ^2) or $(\vec{\mu}, \mathbf{\Sigma})$ is sampled from a normal distribution, $\mathcal{N}$, and an inverse gamma distribution, Γ^{-1}, or inverse Wishart distribution, $\mathcal{W}^{-1}$,

$$\sigma^2 \sim \Gamma^{-1}(\alpha, \beta) \qquad\qquad \mathbf{\Sigma} \sim \mathcal{W}^{-1}(\mathbf{\Psi}, \nu) \equiv \mathcal{W}^{-1}(\nu\mathbf{\Sigma}_0, \nu) \tag{6a}$$

$$\mu|\sigma^2 \sim \mathcal{N}(\mu_0, \sigma^2/\kappa) \qquad\qquad \vec{\mu}|\mathbf{\Sigma} \sim \mathcal{N}(\vec{\mu}_0, \mathbf{\Sigma}/\kappa) \tag{6b}$$

where only the sampling of the variance is i.i.d. since it enters via the scaling parameter κ in the likelihood of the mean. Using $\mathbf{\Sigma}_0$ rather than $\mathbf{\Psi}$ corresponds to parametrizing the distribution of $\mathbf{\Sigma}$ with an inverse χ^2- rather than a Γ^{-1}-distribution in the univariate case, which has the advantage of a clearer interpretation of $\mathbf{\Sigma}_0$.

In Appendix B we derive that taking a NIG (NIW) distribution as a conjugated prior corresponds to assuming prior knowledge about the mean and the variance extracted from κ virtual measurements of the former and 2α virtual measurements (ν virtual measurements) for the latter. Therefore, it appears natural to define the total evidence of the prior as the sum of the number of virtual measurements:

$$\Phi' := \kappa + 2\alpha \qquad\qquad \Phi' := \kappa + \nu \tag{7}$$

where the former (latter) refers to the univariate (multivariate) case. In [2] the authors define the total evidence of the univariate case as[3]

$$\Phi := 2\kappa + \alpha\,. \tag{8}$$

For consistency we therefore propose to change this and to adopt our definition of Φ'. We will revisit this definition in the next section and discuss it in the context of the evidence regularizer.

2.2 Ambiguity of shape parameters

We follow the approach in [2] and use the posterior predictive or model evidence of a NIW distribution for finding a proper loss function $\mathcal{L}_i^{\text{NLL}}$. From Bayesian probability theory the model evidence is a marginal likelihood and, as such, defined as the likelihood of an observation, $\vec{y}_i \in \mathbb{R}^n$, given the evidential distribution parameters, $\mathfrak{m} = (\vec{\mu}_0, \mathbf{\Psi}, \kappa, \nu)$, and is computed by marginalizing over the likelihood parameter (a.k.a. the nuisance parameter), $\boldsymbol{\theta} = (\vec{\mu}, \mathbf{\Sigma})$, where $\kappa, \nu \in \mathbb{R}$, $\vec{\mu}, \vec{\mu}_0 \in \mathbb{R}^n$, and $\mathbf{\Sigma}, \mathbf{\Psi}$ are positive definite $\mathbb{R}^{n\times n}$ matrices. In our case of placing a NIW evidential prior on a multivariate Gaussian likelihood function an analytical solution exists and can be parametrized with a multivariate t-distribution with $\nu - n + 1$ DoF (see Appendix C.2 and C.3 for more details):

$$p(\vec{y}_i|\mathfrak{m}) = \int \mathrm{d}\boldsymbol{\theta}\, \mathcal{N}(\vec{y}_i|\boldsymbol{\theta})\ \texttt{NIW}(\boldsymbol{\theta}|\mathfrak{m}) = t_{\nu-n+1}\left(\vec{y}_i \middle| \vec{\mu}_0, \frac{1}{\nu - n + 1}\frac{1+\kappa}{\kappa}\mathbf{\Psi}\right). \tag{9}$$

[2] As eluded previously we suppress indices for the sake of brevity.

[3] In the notation of [2] κ becomes ν and the total evidence reads $\Phi = 2\nu + \alpha$.

Using this result we can compute the negative log-likelihood loss $\mathcal{L}_i^{\text{NLL}}$ for sample i as:

$$\begin{aligned}\mathcal{L}_i^{\text{NLL}} = -\log p(\vec{y}_i|\mathfrak{m}) = \log\Gamma\left(\frac{\nu-n+1}{2}\right) - \log\Gamma\left(\frac{\nu+1}{2}\right) \\ + \frac{n}{2}\log\left(\pi\,\frac{1+\kappa}{\kappa}\right) - \frac{\nu}{2}\log|\mathbf{\Psi}| \\ + \frac{\nu+1}{2}\log\left|\mathbf{\Psi} + \frac{\kappa}{1+\kappa}(\vec{y}_i-\vec{\mu}_0)(\vec{y}_i-\vec{\mu}_0)^\top\right| . \end{aligned} \tag{10}$$

From the compact notation in Eq. (9) it is obvious that $p(\vec{y}_i|\mathfrak{m})$ on its own is not capable of defining $\mathfrak{m}$ unambiguously. In particular, a fitting approach could be used to find ν, $\vec{\mu}_0$ and the product $(1+\kappa)/\kappa\,\mathbf{\Psi}$ from data. However, in order to disentangle the latter additional constraints have to be set, e.g., via an additional regularization of κ.

The higher-order evidential distribution is projected by integrating out the nuisance parameters $\vec{\mu}$ and $\mathbf{\Sigma}$ and, in the univariate case, the four DoF of $\mathfrak{m}$ collapse into three DoF of a scaled Student's t-distribution. Fitting this reduced set of DoF is not sufficient to recover all DoF of the evidential distribution. The impact of this observation is that fitting the width of the t-distribution will not help to unfold κ and $\mathbf{\Psi}$ and it is possible to find manifolds with different values of κ and β but with the same value for the loss function $\mathcal{L}_i^{\text{NLL}}$. In fact, κ can be tuned such that for any given value of $\mathcal{L}_i^{\text{NLL}}$ a value for $\mathbf{\Psi}$ or β can be find. Therefore, $\mathcal{L}_i^{\text{NLL}}$ on its own is not sufficient to learn the parameters $\mathfrak{m}$. (See Appendix D for more details.)

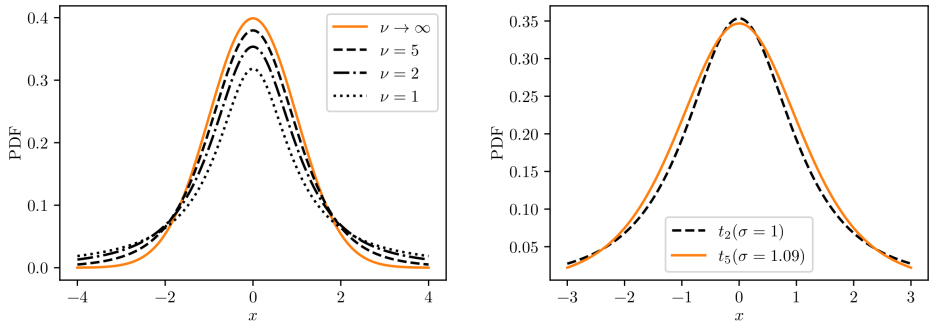
In [2] this degeneration of the loss function is broken by introducing the evidence regularizer,

$$\mathcal{L}_i^{\text{R}} = |y_i - \mu_0|\Phi\,, \tag{11}$$

however, it is unclear how the NN could learn the parameters $\mathfrak{m}$ when the evidence regularizer is disabled by setting $\lambda = 0$. More importantly, although $\mathcal{L}_i^{\text{R}}$ breaks the degeneration, using $\Phi = 2\kappa+\alpha$, the total loss $\mathcal{L}_i = \mathcal{L}_i^{\text{NLL}} + \lambda\mathcal{L}_i^{\text{R}}$ can easily be minimized (in theory) for κ by simply sending $\kappa \to 0$ since any impact on $\mathcal{L}_i^{\text{NLL}}$ can be compensated by adjusting β without changing the value of $\mathcal{L}_i^{\text{NLL}}$. Sending κ to zero drives the ratio of aleatoric and epistemic uncertainty to zero as well, cf. Eqs. (2), thus making their values useless. (In practice, $\kappa \to 0$ is numerically unstable and minimizer we fail to converge towards this point.)

In summary: The loss $\mathcal{L}_i^{\text{NLL}}$ is degenerated and requires regularization of either κ or $\mathbf{\Psi}$. The proposed regularizer of [2] does not lead to correct uncertainty estimations and has a numerical unstable minimum.

2.3 Challenging extraction of shape parameters



(a) Student's t-distributions.

(b) Fit of a t_2 distribution using a t_5 function.

Figure 2: (Left) Student's t-distribution for different values of ν and (right) fit of a t-distribution with $\nu = 5$ to a t-distribution with $\nu = 2$.

We argued previously that minimizing Eq. (10) is nothing but the fit of a t-distribution to data. In Appendix A we further elaborate that technically only a single data point is fitted per distribution,

although correlations between neighboring points will enter in practice. It is therefore difficult to estimate the number of data points used per fit, still, we argue here that a large statistic is necessary to extract the parameter ν reliably which plays a crucial role in estimating the epistemic uncertainty.

In general, small values for ν will raise the tails of the distribution but only slightly affect the shape of the core of the function where most measurements will be find, assuming that these indeed follow a t-distribution. Extraction of ν using a fit thus needs a large statistic which properly describes the tails. In reality, however, the assumption of normal distributed events often collapses especially in the tails of a distribution which makes a fit of ν even more ambitious. Assuming one only has a decent statistic near the core, the parameters ν and σ of a scaled Student's t-distribution,[4] $\mathrm{St}_\nu(y_i|\mu,\sigma)$, are highly correlated as shown in Fig. 2b where we fit σ of a t-distribution with $\nu = 5$ to a t-distribution with $\nu = 2$ on the interval $x \in [-3, 3]$.

To study possible biases of the fitted values of the parameters ν and κ we conduct a pseudo experiment where we generate data by drawing them i.i.d. from Student's t-distributions with fixed shape parameters and fit them on different sample sizes. (See Appendix E for details.) For each sample size we evaluate 200 fits and find a bias for ν and σ for small sample sizes which decreases if the sample size increases.

In summary: Extracting the parameter ν of a t-distribution requires a sufficiently large data sample. Due to correlations the fit result is biased which can be significant if the sample size is too small. We showed this for the univariate case (where ν corresponds to 2α) but the same holds for the multivariate case as well where an even larger data set is needed.

3 A possible solution

In this section we describe a possible solution for two of the aforementioned issues. The issue regarding the correlation of the shape parameters of a t-distribution is not affected by our solution proposal and biases have to be studied using data.

In [2] the authors did not introduce the term $\mathcal{L}_i^{\mathrm{R}}$ as a way to lift the degeneration of $\mathcal{L}_i^{\mathrm{NLL}}$ but motivate it as an *evidential regularizer*, similar to [10]. The idea of combining the ℓ_1-norm of the prediction error with the total evidence Φ is to enforce the NN to learn large errors in the prediction are acceptable, as long as this is reflected in a small total evidence and vice-versa. In other words, in the absence of data which would have the potential to squeeze the prediction error, the prior information should approach an uninformed prior and $\mathcal{L}_i^{\mathrm{R}}$, as proposed by the authors, is one possible metric to measure the distance to it but does not lead to a meaningful minimum as described previously. Finding a suitable metric which breaks the degeneration of $\mathcal{L}_i^{\mathrm{NLL}}$ but also pushes the distribution towards an uninformed prior is not straight-forward. For example, instances of the f-divergence family, differential entropy or taking the peak height of the function as a measure to meet the second requirement do not break the degeneration. This is because any metric of $t_\nu(\vec{\mu}_t, \mathbf{\Sigma}_t)$ is, by construction, ignorant of internal dependencies of the shape parameters but in order to break the degeneration κ and $\mathbf{\Sigma}_0$ have to be unfolded from $\mathbf{\Sigma}_t = \mathbf{\Sigma}_t(\nu, \kappa, \mathbf{\Sigma}_0)$.

We therefore propose to acknowledge the loss of one DoF and couple the parameters κ and ν with a constant hyperparmeter r, i.e., using again the index notation:

$$\nu_i = r\kappa_i \,. \tag{12}$$

Including an evidential regularizer as $|\vec{y}_i - \vec{\mu}_{0,i}|\nu_i$ to the loss function is therefore no longer necessary and minimizing $\mathcal{L}_i = \mathcal{L}_i^{\mathtt{NLL}}$ is sufficient. We motivate this ansatz by considering it unnatural to have prior information from κ virtual measurements for the mean and ν virtual measurements for the variance where the ratio κ/ν significantly fluctuates throughout the data sample or even differs largely in scale. Another way of seeing the implicit coupling of both variables is the case of vanishing epistemic uncertainty, i.e., $\kappa \to \infty$. The model should then become a normal distribution as described in Appendix A which corresponds to $\nu \to \infty$. (This is $\alpha \to \infty$ in the univariate case.) Coupling κ and ν, as proposed in Eq. (12), enforces this behavior.

In summary: For the univariate case we propose to couple the parameters κ and α via a hyperparameter r that is kept constant for all instances. Similarly, in the multivariate case one should couple κ and ν with r. In the next section we summarize how this changes the loss function.

[4] For the sake of brevity we overload here the notations for μ and σ.

4 Multivariate generalization

In this section we summarize our multivariate generalization, combine it with our proposed solution from Sec. 3 and benchmark it with a multivariate experiment. In general, using our proposed multivariate generalization it is possible to not just learn uncertainties of each regression target of a multivariate dataset individually, but also to learn their correlations. Whereas using chained univariate regressions it is possible by sampling to extract the correlation of $(\vec{y}_i, \vec{y}_j) \in \mathbb{R}^{n\times n}$ pairs at (x_i, x_j) with $i \neq j$, it is impossible by the very nature of the univariate distributions to get the correlation of $(y_{ij}, y_{ik}) \in \mathbb{R}^{1\times 1}$ at (x_i, x_i). The latter, i.e., the feature correlation at each point x_i, can only be extracted with a multivariate approach.

Taking a `NIW` distribution as the conjugated prior we found that minimizing the loss

$$\begin{aligned}\mathcal{L}_i \equiv \mathcal{L}_i^{\texttt{NLL}} = &\log\Gamma\left(\frac{\nu_i - n + 1}{2}\right) - \log\Gamma\left(\frac{\nu_i + 1}{2}\right) \\ &+ \frac{n}{2}\log(r+\nu_i) - \nu_i \sum_j \ell_j^{(i)} \\ &+ \frac{\nu_i + 1}{2}\log\left|\boldsymbol{L}_i\boldsymbol{L}_i^\top + \frac{1}{r+\nu_i}(\vec{y}_i - \vec{\mu}_{0,i})(\vec{y}_i - \vec{\mu}_{0,i})^\top\right| + \text{const.}\end{aligned} \tag{13}$$

allows the estimation of the prediction and both types of uncertainties as:[5]

$$\underbrace{\mathbb{E}[\mu] = \vec{\mu}_{0,i}}_{\text{prediction}} \qquad \underbrace{\mathbb{E}[\boldsymbol{\Sigma}] \propto \frac{\nu_i}{\nu_i - n - 1}\boldsymbol{L}_i\boldsymbol{L}_i^\top}_{\text{aleatoric}} \qquad \underbrace{\text{var}[\vec{\mu}] \propto \mathbb{E}[\boldsymbol{\Sigma}]/\nu_i}_{\text{epistemic}} \tag{14}$$

where we rewrote the positive (semi-)definite matrix $\boldsymbol{\Psi}_i = \nu_i\boldsymbol{\Sigma}_{0,i} = \nu\boldsymbol{L}_i\boldsymbol{L}_i^\top \in \mathbb{R}^{n\times n}$ with the lower triangular matrix $\boldsymbol{L}_i$ and enforce positive diagonal elements by parametrizing with $\vec{\ell}^{(i)} \in \mathbb{R}^{n(n+1)/2}$:

$$(\boldsymbol{L}_i)_{jk} = \begin{cases} \exp\left\{\ell_j^{(i)}\right\} & \text{if } j = k, \\ \ell_{jk}^{(i)} & \text{if } j > k, \\ 0 & \text{else.} \end{cases} \tag{15}$$

We note that regardless of this rewriting the limit $\nu_i \to \infty$ is numerically unstable and cut-offs have to be placed in practice.

In order to learn the parameters $\mathfrak{m}_i = (\vec{\mu}_{0,i}, \vec{\ell}^{(i)}, \nu_i)$ a NN has to have $n(n+3)/2+1$ output neurons and one hyperparameters r. By using Eq. (12) we acknowledge the loss of one DoF due to the projection of the higher-order `NIW` distribution. This reduction comes with the cost that we loose the predictive power on a global scale of the aleatoric and the epistemic uncertainties which we assume to be constant by taking r as a hyperparameter. In practice this means that if a global scale is of interest, one has to rescale the predicted uncertainties (which is viable with either Bayesian or Frequentist techniques).

Finally, we conduct a simple experiment with $n = 2$ and $r = 1$ to benchmark our multivariate generalization. Our experiment is built upon the univariate experiment described by Amini et al. [2] with a critical difference: Rather than training and evaluating the NN on partially disjunct data samples[6] where the NN has no chance to identify an increasing epistemic uncertainty, we generate data in the xy-plane with varying density. We overload our notation of x and y now being the features of our data sample given input t,

$$x = (1+\epsilon)\cos t \qquad\qquad y = (1+\epsilon)\sin t\,, \tag{16a}$$

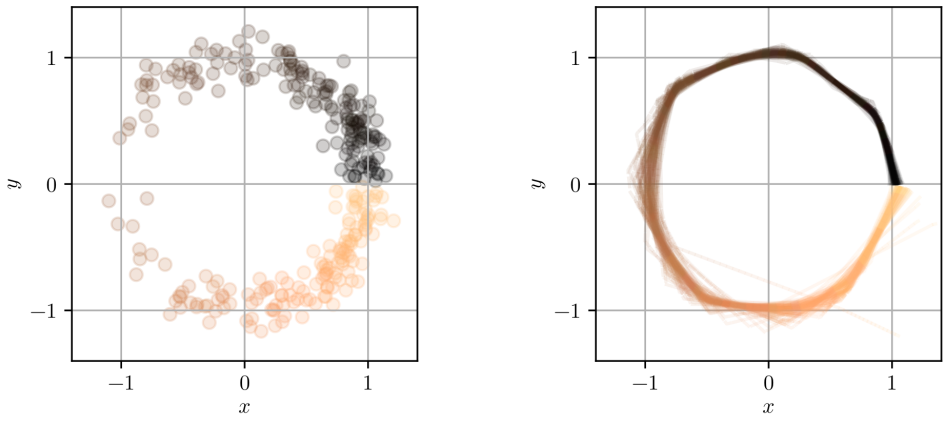
where the distribution of t is not flat but has a $\vee$ shape,

$$t \sim \begin{cases} 1 - \frac{\zeta}{\pi} & \text{if } \zeta \in [0, \pi]\,, \\ \frac{\zeta}{\pi} - 1 & \text{if } \zeta \in (\pi, 2\pi]\,, \\ 0 & \text{else,} \end{cases} \tag{16b}$$

[5] See Appendix C.1 for a derivation.

[6] In [2] the NN is trained for $t \in [-4, +4]$ but evaluated on $t \in [-7, +7]$.

with uniformly distributed $\zeta \in [0, 2\pi]$ and ϵ is drawn from a normal distribution, $\epsilon \sim \mathcal{N}(0, 0.1)$. We draw 300 data points in total (see Fig. 3a) and fit the distribution with a small, fully connected NN with a single input neuron, two hidden layers of 32 neurons using Rectified Linear Unit activation functions, and six output neurons. The output of the last layer, $\vec{p} \in \mathbb{R}^6$, is transformed and subsequently taken



(a) Synthetic data

(b) Predictions of NNs

Figure 3: (Left) The distribution of our test data in the xy-plane. The value of t is color coded (see Fig. 8 for a 3d representation). (Right) the overlayed prediction of 100 trained NNs for given values of t.

as the parameters of the evidential distribution:

$$\vec{\mu} = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \quad \boldsymbol{L} = \begin{pmatrix} \exp\{p_3\} & 0 \\ p_4 & \exp\{p_5\} \end{pmatrix}, \quad \nu = 8 + 5 \tanh p_6 \,, \tag{17}$$

where an exponential function is used to constrain the diagonal elements of $\boldsymbol{L}$ to be strictly positive and the transformation of ν corresponds to the required lower bound[7] $\nu > n + 1 = 3$ and a cut-off $\nu < 13$ – empirically, t-distributions with more than 13 DoF are almost indistinguishable from genuine normal distributions.

Constraining ν onto the interval $\nu \in (3, 13)$ makes this parameter effectively a gate. Being closed, $\nu \approx 3$, corresponds to the situation that the additional DoF of a t-distribution helps to better fit the data, whereas being open, $\nu \gg 3$, indicates that the genuine distribution function actually yields the better fit result. That is, even though the data are drawn from a normal distribution, a fit with a more flexible function will more likely find a better minimum for sparsely sampled data. This extra flexibility of the t-distribution w.r.t. a normal distribution, coming from the extra DoF, becomes less important when the data distribution becomes denser and, on average, better resembles its genuine distribution function.

In total we train 100 NNs from scratch on the synthetic data sample and overlay their predictions for $x = \mu_1$ and $y = \mu_2$ in Fig. 3b. More importantly for this work is the behavior of the parameter ν which we show in Fig. 4. We find a statistical significant drop in ν towards the center of t which corresponds to an enhancement of the epistemic uncertainty. It is exactly this area where the data distribution becomes sparse and therefore nicely meets our expectations. However, not all NNs converged to this solution and fluctuations are present. In fact, we find the shown behavior being strongly correlated with the sample size of the synthetic data sample: reducing the sample size causes serious over-fitting of the model, whereas increasing quickly opens the ν gate for all values of t. We find that in presence of a sufficient amount of data not just the ν gate is open for all values of t, but the predicted values of x and y resemble the genuine distribution well, also in the regions $0.5\pi \lesssim t \lesssim 1.5\pi$ (we already see this behavior indicated in Fig. 3b) which is again in agreement with our expectations.

The outlined technique therefore helps to detect variations of the epistemic uncertainty throughout the data landscape. See Appendix F for more details.

[7] See Appendix C.1 for details.

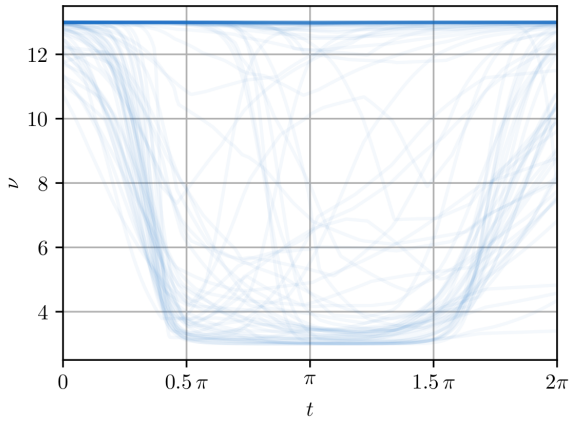


Figure 4: Overlayed parameter ν of 100 trained NNs. Low (high) values correspond to a large (low) epistemic uncertainty.

5 Related Work

Our work builds specifically on the prior art [2] for uncertainty estimation with evidential neural networks, and more generally on the advancing area of uncertainty reasoning in deep learning.

The probabilistic perspective in machine learning (ML) frames learning as inferring plausible models to explain observed data. Observed data can be consistent with many models, and therefore which model is appropriate given the data is uncertain [12]. Probabilistic (or Bayesian) ML methods are rooted in probability theory and thus provide a framework for modeling uncertainties. Traditional probabilistic methods include Gaussian processes [13], latent variable models [14], and probabilistic graphical models [15]. In recent years there have been many explorations into Bayesian approaches to deep learning [1, 16, 17, 18, 19, 20, 21, 22]. The key observation is that neural networks are typically underspecified by the data, thus different settings of the parameters correspond to a diverse variety of compelling explanations for the data – i.e., a deep learning posterior consists of high performing models which make meaningfully different predictions on test data, as demonstrated in [21, 23, 24]. This underspecification by NNs makes Bayesian inference, and by corollary uncertainty estimation, particularly compelling for deep learning. Bayesian deep learning aims to compute a distribution over the model parameters during training in order to quantify uncertainties, such that the posterior is available for uncertainty estimation and model calibration [17]. With Bayesian NNs that have thousands and millions of parameters this posterior is intractable, so implementations largely focus on several approximate methods for Bayesian inference: First, Markov Chain Monte Carlo (MCMC) methods iteratively draw samples from the unknown posterior distribution, and efficient MCMC methods make use of gradient information rather than performing random walks. In particular stochastic gradient MCMC for Bayesian NNs [25, 26, 27, 28], with a main drawback being the inability to capture complex distributions in the parameter space without increasing the computational overhead. Second, variational inference (VI) performs Bayesian inference by using a computationally tractable *variational* distribution to approximate the posterior. One approach by Graves et al. [29] is to use a Gaussian variational posterior to approximate the distribution of the weights in a network, but the capacity of the uncertainty representation is limited by the variational distribution. In general we see that MCMC has a higher variance and lower bias in the estimate, while VI has a higher bias but lower variance [30]. The preeminent Bayesian deep learning approach by Gal and Ghahramani [7] showed that variational inference can be approximated without modifying the network. This is achieved through a method of approximate variational inference called Monte Carlo Dropout (MCD), whereby dropout is performed during inference, using multiple dropout masks.

Better understanding of the integration of deep learning with probabilistic ML such as Gaussian processes (GP) is also a fruitful direction, namely with Deep Kernel Learning [18, 31] and deep GP [32, 33].

Alternative to the prior-over-weights approach of Bayesian NN, one can view deep learning as an evidence acquisition process – different from the Bayesian modeling nomenclature, evidence here is a measure of the amount of support collected from data in favor of a sample to be classified into a certain class, and uncertainty is inversely proportional to the total evidence [10]. Samples during training each add support to a learned higher-order, evidential distribution, which yields epistemic and aleatoric uncertainties without the need for sampling. Several recent works develop this approach to deep learning and uncertainty estimation which put this in practice with *prior networks* that place priors directly over the likelihood function [2, 6]. These approaches largely struggle with regularization [10], generalization (particularly without using out-of-distribution training data) [6, 19], capturing aleatoric uncertainty [34], and the issues we have addressed above with the prior art Deep Evidential Regression [2].

There are also the frequentist approaches of bootstrapping and ensembling, which can be used to estimate NN uncertainty without the Bayesian computational overhead as well as being easily parallelizable – for instance Deep Ensembles, where multiple randomly initialized NNs are trained and at test time the output variance from the ensemble of models is used as an estimate of uncertainty [8].

6 Conclusion

We discussed the recent developments towards uncertainty-aware neural networks, *Deep Evidential Regression* [2], identifying several outstanding issues with the approach and providing detailed solutions grounded in theory and experimental results. In addition to correcting the prior art, we extend it for multivariate scenarios. The solutions and new approach we presented here would benefit from future studies towards empirical validation.

Broader Impact

Neural networks have already had significant impacts in many applications – from medical imaging to dialogue systems to autonomous vehicles – and will continue to do so for years to come. Yet there are important shortcomings in our understanding and confidence in this class of machine learning, notably in the ability to estimate uncertainties and calibrate models. This problem becomes more complex, and potentially dangerous, when NNs are built within larger systems that combine data, software, hardware, and people in dynamic, complex ways. Reliable and systematic methods of uncertainty quantification with NNs are needed, especially considering deployments in safety critical domains such as medicine and autonomous vehicles. In addition to the practical utility of reliably quantifying uncertainty in NNs, there is a significant need to build confidence in the models and establish trust with the end-users. Work towards quantifying uncertainties in machine learning is essential such that these models and systems "know when they don't know", and are thus more trusted and usable in real-world scenarios.

Reproducibility

The source code for reproducing our experiments – implementation of the NN and multivariate methods, and algorithms to generate the data – is available on `github.com/avitase/mder/` (MIT License). The model and experiments are lightweight, running locally on a 4-core MacBook Pro in under an hour. We use the `Python` programming language as well as various libraries, most notably `PyTorch`, `NumPy`, `Jupyter` and `matplotlib`.

References

[1] A. Kendall & Y. Gal, *What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?*, arXiv:1703.04977 [cs.CV] (2017).

[2] A. Amini et al., *Deep Evidential Regression*, arXiv:1910.02600 [cs.LG] (2020).

[3] A. Geiger et al., *Are we ready for autonomous driving? The KITTI vision benchmark suite*, IEEE Conference on Computer Vision and Pattern Recognition, 10.1109/CVPR.2012.6248074 (2012).

[4] M. Bojarski et al., *End to End Learning for Self-Driving Cars*, arXiv:1604.07316 (2016).

[5] C. Godard et al., *Unsupervised Monocular Depth Estimation with Left-Right Consistency*, arXiv:1609.03677 [cs.CV] (2017).

[6] A. Malinin & M. Gales, *Predictive Uncertainty Estimation via Prior Networks*, arXiv:1802.10501 [stat.ML] (2018).

[7] Y. Gal & Z. Ghahramani, *Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning*, arXiv:1506.02142 [stat.ML] (2016).

[8] B. Lakshminarayanan et al., *Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles*, arXiv:1612.01474 [stat.ML] (2017).

[9] M. Jain, *DEUP: Direct Epistemic Uncertainty Prediction*, arXiv:2102.08501 [cs.LG] (2021).

[10] M. Sensoy et al., *Evidential Deep Learning to Quantify Classification Uncertainty*, arXiv:1806.01768 [cs.LG] (2018).

[11] A. Gelman et al., *Bayesian Data Analysis*, Third Edition, ISBN:978-1439840955.

[12] Z. Ghahramani, *Probabilistic machine learning and artificial intelligence*, Nature 521, 452–459 (2015).

[13] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, ISBN:0-262-18253-X (2006).

[14] C. Bishop, *Latent Variable Models*, ISBN:978-94-011-5014-9 (1998).

[15] D. Koller & N. Friedman, *Probabilistic Graphical Models* ISBN:978-0-262-01319-2 (2009).

[16] R.M. Neal, *Bayesian Learning for Neural Networks*, ISBN:978-1-4612-0745-0 (1996).

[17] C. Guo et al., *On Calibration of Modern Neural Networks*, arXiv:1706.04599 [cs.LG] (2017).

[18] A.G. Wilson et al., *Deep Kernel Learning*, arXiv:1511.02222 [cs.LG] (2015).

[19] D. Hafner et al., *Noise Contrastive Priors for Functional Uncertainty*, arXiv:1807.09289 [stat.ML] (2018).

[20] Y. Ovadia et al., *Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift*, arXiv:1906.02530 [stat.ML] (2019).

[21] P. Izmailov et al., *Subspace inference for Bayesian deep learning*, arXiv:1907.07504 [cs.LG] (2019).

[22] N. Seedat & C. Kanan, *Towards calibrated and scalable uncertainty representations for neural networks*, arXiv:1911.00104 [cs.LG] (2019).

[23] T. Garipov et al., *Loss Surfaces, Mode Connectivity, and Fast Ensembling of DNNs*, arXiv:1802.10026 [stat.ML] (2018).

[24] K. Zołna et al., *Classifier-agnostic saliency map extraction*, arXiv:1805.08249 [cs.LG] (2020).

[25] M. Welling & Y.W. Teh, *Bayesian learning via Stochastic Gradient Langevin Dynamics*, ICML, 681-688 (2011).

[26] C. Li et al., *Learning Weight Uncertainty with Stochastic Gradient MCMC for Shape Classification*, CVPR, 5666-5675 (2016).

[27] C. Park et al., *Sampling-based Bayesian Inference with gradient uncertainty*, arXiv:1812.03285 [cs.LG] (2018).

[28] W. Maddox, *A Simple Baseline for Bayesian Uncertainty in Deep Learning*, arXiv:1902.02476 [cs.LG] (2019).

[29] A. Graves et al., *Speech Recognition with Deep Recurrent Neural Networks*, arXiv:1303.5778 [cs.NE] (2013).

[30] P.-A. Mattei, *A Parsimonious Tour of Bayesian Model Uncertainty*, arXiv:1902.05539 [stat.ME] (2019).

[31] A. Lavin, *Neuro-symbolic Neurodegenerative Disease Modeling as Probabilistic Programmed Deep Kernels*, arXiv:2009.07738 [cs.LG] (2021).

[32] D. Duvenaud et al., *Avoiding pathologies in very deep networks*, arXiv:1402.5836 [stat.ML] (2014).

[33] V. Dutordoir et al., *Deep Neural Networks as Point Estimates for Deep Gaussian Processes*, arXiv:2105.04504 [stat.ML] (2021).

[34] P. Gurevich & H. Stuke, *Gradient conjugate priors and multi-layer neural networks*, arXiv:1802.02643 [math.ST] (2019).

A Maximum likelihood estimation using Gaussians

A simple approach to estimate uncertainties of a regression-based NN is to assume that data are drawn i.i.d. from normal distributions, i.e.,

$$y_i \sim \mathcal{N}\big(\mu_i, \sigma_i^2\big) \equiv \mathcal{N}\big(\mu(x_i), \sigma^2(x_i)\big) \tag{18}$$

for each (x_i, y_i) pair of the data sample at hand. Instead of using Bayesian inference one could simple seek for the maximum of the combined likelihood

$$\max_{\boldsymbol{w}} L(\boldsymbol{w}) = \prod_i \mathcal{N}\big(y_i|\mu_i, \sigma_i^2\big)\,, \tag{19}$$

or, alternatively, for the minimum of the negative log-likelihood

$$\min_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}) = \sum_i \underbrace{\left(\frac{1}{2}\log(2\pi\sigma_i^2) + \frac{(y_i-\mu_i)^2}{2\sigma_i^2}\right)}_{\mathcal{L}_i(\boldsymbol{w})} \tag{20}$$

and let the NN itself estimate μ_i and σ_i^2 by adding one extra neuron to the output layer and interpret the output values of these two neurons as $\mu_i \equiv \mu(x_i, \boldsymbol{w})$ and $\sigma_i^2 \equiv \sigma^2(x_i, \boldsymbol{w})$.

We note that this corresponds to fitting Gaussian functions to single data points and it is the objective of the supervisor of the training process of the NN to ensure that $\mu(x_i, \boldsymbol{w})$ and $\sigma^2(x_i, \boldsymbol{w})$ do not over-fit the data as shown in the right side of Fig. 5. Here, the model is obviously unable to differentiate

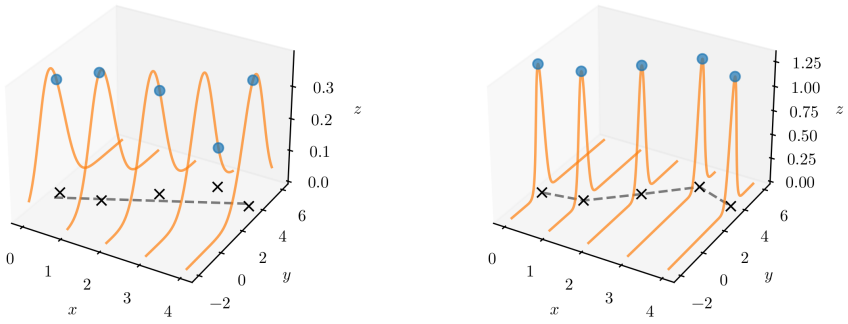


Figure 5: Fitting a data distribution in x and y using Eq. (20). The prediction of the fitted model is shown as a dashed line in the x-y plane and the value of $\mathcal{L}_i$ as a solid line in the y-z plane. The fit on the right over-fits the data but has the smaller total loss $\mathcal{L}$.

between aleatoric and epsitemic uncertainty and will merge both components into σ_i if the model is under-fitting. In case of missing data the model will likely interpolate between regions where data are available and thus underestimate the epistemic uncertainty.

For the sake of brevity we drop the index notation of the parameters of the likelihood in the main text but stress the tight coupling of each (x_i, y_i) pair to its individual (μ, σ^2) pair. In reality however, given a sufficiently large data sample, a well behaving NN will likely produce similar values for (μ, σ^2) if two pairs (x_i, y_i) and (x_j, y_j) are close. Technically, this approach is equivalent to fitting independent functions to each data point but the NN will correlate adjacent points.

B Interpretation of the shape parameters of NIG and NIW distributions

An interpretation of the shape parameters of a NIG or NIW distribution can be find by analyzing the joint posterior density after taking m measurements,

$$\vec{y} \in \mathbb{R}^m \qquad \text{or} \qquad \boldsymbol{Y} = (y_{ij}) = \begin{pmatrix} \vec{y}_1^{\,T} \\ \vdots \\ \vec{y}_n^{\,T} \end{pmatrix} \in \mathbb{R}^{n\times m}\,, \tag{21}$$

i.e., multiplying the prior density by the normal likelihood yields the posterior density

$$p(\mu, \sigma^2|\vec{y}) = \texttt{NIG}(\mu_0', \kappa';\, \alpha', \beta') \qquad p(\vec{\mu}, \boldsymbol{\Sigma}|\boldsymbol{Y}) = \texttt{NIW}(\vec{\mu}_0', \kappa';\, \boldsymbol{\Psi}', \nu') \tag{22}$$

with

$$\kappa' = \kappa + m \qquad \kappa' = \kappa + m \tag{23a}$$

$$2\alpha' = 2\alpha + m \qquad \nu' = \nu + m \tag{23b}$$

$$\mu_0' = \frac{1}{\kappa+m}\begin{pmatrix}\kappa \\ m\end{pmatrix}\begin{pmatrix}\mu_0 \\ \langle\vec{y}\,\rangle\end{pmatrix} \qquad \vec{\mu}_0' = \frac{1}{\kappa+m}\begin{pmatrix}\kappa \\ m\end{pmatrix}\begin{pmatrix}\vec{\mu}_0 \\ \langle\boldsymbol{Y}\,\rangle\end{pmatrix} \tag{23c}$$

$$2\beta' = 2\beta + \tilde{s} + m\frac{\kappa}{\kappa'}\,(\mu_0 - \langle\vec{y}\,\rangle)^2 \qquad \boldsymbol{\Psi}' = \boldsymbol{\Psi} + \tilde{\boldsymbol{S}} + m\frac{\kappa}{\kappa'}\,(\vec{\mu}_0 - \langle\boldsymbol{Y}\rangle)(\vec{\mu}_0 - \langle\boldsymbol{Y}\rangle)^\top \tag{23d}$$

where we introduced the squared sum of residuals (up to a scaling constant $(n-1)$ these are estimators of the sample variance)

$$\tilde{s} = \sum_{i=1}^{m}(y_i - \langle\vec{y}\,\rangle)^2 \qquad \tilde{\boldsymbol{S}} = \sum_{i=1}^{m}(\vec{y}_i - \langle\boldsymbol{Y}\rangle)\,(\vec{y}_i - \langle\boldsymbol{Y}\rangle)^\top \tag{24}$$

and the expectation value

$$\langle\vec{y}\,\rangle = \frac{1}{n}\sum_{i}^{n} y_i \in \mathbb{R} \qquad \langle\boldsymbol{Y}\rangle = \frac{1}{n}\sum_{i,j=1}^{m,n} y_{ij}\,\vec{e}_i \in \mathbb{R}^m\,. \tag{25}$$

These relations can easily be interpreted as the combination of prior information and the information contained in the data. In particular, Eq. (23c) reads as the weighted sum of two measurement outcomes of the mean, where the weights correspond to κ (virtual) measurements encoded in the prior and the m (actual) measurements, i.e., the prior distribution $\mathcal{N}(\vec{\mu}_0, \boldsymbol{\Sigma}/\kappa)$ can be thought of as providing the information equivalent to κ observations with sample mean $\vec{\mu}_0$. Similarly, using Eq. (23d), the prior distribution $\mathcal{W}^{-1}(\nu\boldsymbol{\Sigma}_0, \nu)$ can be thought of as providing the information equivalent to ν observations with average squared deviation $\boldsymbol{\Sigma}_0$.

C Derivation of multivariate generalization in detail

C.1 Moments

We assume our data was drawn from a multivariate Gaussian with unknown mean and variance $(\vec{\mu}, \mathbf{\Sigma})$. We probabilistically model these parameters $\boldsymbol{\theta}$ according to:

$$\begin{aligned}\mathbf{\Sigma} &\sim \mathcal{W}^{-1}(\mathbf{\Psi}, \nu) \equiv \mathcal{W}^{-1}(\nu\mathbf{\Sigma}_0, \nu)\,,\\ \vec{\mu}|\mathbf{\Sigma} &\sim \mathcal{N}(\vec{\mu}_0, \mathbf{\Sigma}/\kappa)\,,\end{aligned}$$

where $\mathcal{N}$ is a multivariate normal distribution

$$\mathcal{N}(\vec{x}\,|\vec{\mu}, \mathbf{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n\,|\mathbf{\Sigma}|}}\exp\left\{-\frac{1}{2}\,(\vec{x}-\vec{\mu})^\top\mathbf{\Sigma}^{-1}(\vec{x}-\vec{\mu})\right\} \tag{26a}$$

$$= \frac{1}{\sqrt{(2\pi)^n\,|\mathbf{\Sigma}|}}\exp\left\{-\frac{1}{2}\,\mathrm{tr}\big((\vec{x}-\vec{\mu})(\vec{x}-\vec{\mu})^\top\mathbf{\Sigma}^{-1}\big)\right\} \tag{26b}$$

and $\mathcal{W}^{-1}$ is an Inverse-Wishart distribution

$$\mathcal{W}^{-1}(\mathbf{\Sigma}|\mathbf{\Psi}, \nu) = \frac{1}{\Gamma_n(\nu/2)}\sqrt{\frac{1}{2^{\nu n}}\,\frac{|\mathbf{\Psi}|^\nu}{|\mathbf{\Sigma}|^{\nu+n+1}}}\exp\left\{-\frac{1}{2}\,\mathrm{tr}\big(\mathbf{\Psi}\mathbf{\Sigma}^{-1}\big)\right\}\,.$$

Using $\mathbf{\Sigma}_0$ rather than $\mathbf{\Psi}$ corresponds to parametrizing the distribution of $\mathbf{\Sigma}$ with an inverse χ^2- rather than a Γ^{-1}-distribution in the univariate case. This has the advantage of a clearer interpretation of $\mathbf{\Sigma}_0$, i.e., the prior distribution $\mathcal{W}^{-1}(\nu\mathbf{\Sigma}_0, \nu)$ can be thought of as providing the information equivalent to ν observations with average squared deviation $\mathbf{\Sigma}_0$. Similarly, the prior distribution $\mathcal{N}(\vec{\mu}_0, \mathbf{\Sigma}/\kappa)$ can be though of as providing the information equivalent to κ observations with sample mean $\vec{\mu}_0$, cf. Eqs. (23) and [11].

The prior joint distribution (a `NIW` distribution) factorizes and can be written as:

$$\begin{aligned}p(\vec{\mu}, \mathbf{\Sigma}) &= p(\vec{\mu}) \times p(\mathbf{\Sigma})\\ &= \underbrace{\mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \mathbf{\Sigma}/\kappa) \times \mathcal{W}^{-1}(\mathbf{\Sigma}\,|\mathbf{\Psi}, \nu)}_{\texttt{NIW}}\,.\end{aligned}$$

The first order moments are then given by

$$\langle\vec{\mu}\,\rangle_{\texttt{NIW}} = \langle\vec{\mu}\,\rangle_{\mathcal{N}} = \vec{\mu}_0\,, \tag{27a}$$

$$\begin{aligned}\langle\mathbf{\Sigma}\rangle_{\texttt{NIW}} &= \langle\mathbf{\Sigma}\rangle_{\mathcal{W}^{-1}}\\ &= \frac{1}{\nu-n-1}\,\mathbf{\Psi}\\ &= \frac{\nu}{\nu-n-1}\,\mathbf{\Sigma}_0 \quad (\text{for } \nu > n+1)\,.\end{aligned} \tag{27b}$$

Using these and

$$\langle\mu_i\,\mu_j\rangle_{\texttt{NIW}} = \langle\mu_i\,\mu_j\rangle_{\mathcal{N}} = \Sigma_{ij}/\kappa + \mu_{0,i}\,\mu_{0,j}$$

we find the variance of $\vec{\mu}$ being:

$$\begin{aligned}\mathrm{var}(\vec{\mu}\,)_{\texttt{NIW}} &= \begin{pmatrix}\vdots & \\ \ldots & \langle\mu_i\,\mu_j\rangle_{\texttt{NIW}} \quad \ldots\\ \vdots & \end{pmatrix} - \begin{pmatrix}\vdots & \\ \ldots & \langle\mu_i\rangle_{\texttt{NIW}}\langle\mu_j\rangle_{\texttt{NIW}} \quad \ldots\\ \vdots & \end{pmatrix}\\ &= \frac{1}{\kappa(\nu-n-1)}\mathbf{\Psi}\\ &= \frac{\nu}{\kappa(\nu-n-1)}\mathbf{\Sigma}_0 \quad (\text{for } \nu > n+1)\,.\end{aligned} \tag{27c}$$

We note that in the univariate case, $n = 1$, these relations become

$$\langle\mu\rangle = \mu_0\,, \tag{28a}$$

$$\langle\sigma^2\rangle = \beta/(\alpha-1)\,, \tag{28b}$$

$$\mathrm{var}(\mu) = \beta/(\kappa(\alpha-1)) \tag{28c}$$

for

$$\sigma^2 \sim \Gamma^{-1}(\alpha, \beta)\,,$$
$$\mu|\sigma^2 \sim \mathcal{N}(\mu_0, \sigma^2/\kappa)\,,$$

as expected.

C.2 Model evidence

Here we derive the posterior predictive or model evidence of a NIW distribution,

$$\texttt{NIW}(\vec{\mu}, \mathbf{\Sigma}|\vec{\mu}_0, \mathbf{\Psi}, \kappa, \nu) = \mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \mathbf{\Sigma}/\kappa) \times \mathcal{W}^{-1}(\mathbf{\Sigma}|\mathbf{\Psi}, \nu)\,. \tag{29}$$

where $\mathcal{N}(\vec{x})$ as well as $\mathcal{W}^{-1}(\mathbf{\Sigma})$ are proper normalized in $\vec{x}$ and $\mathbf{\Sigma}$, respectively, such that

$$\int \mathrm{d}\vec{x}\,\mathcal{N}(\vec{x}|\vec{\mu}, \mathbf{\Sigma}) = \int \mathrm{d}\vec{\mu}\,\mathcal{N}(\vec{x}|\vec{\mu}, \mathbf{\Sigma}) = \int \mathrm{d}\vec{\mu}\,\mathcal{N}(\vec{\mu}|\vec{x}, \mathbf{\Sigma}) = 1\,, \tag{30a}$$

$$\int \mathrm{d}\mathbf{\Sigma}\,\mathcal{W}^{-1}(\mathbf{\Sigma}|\mathbf{\Psi}, \nu) = 1\,. \tag{30b}$$

From Bayesian probability theory the model evidence is a marginal likelihood and, as such, defined as the likelihood of an observation $\vec{y}_i \in \mathbb{R}^n$ given the evidential distribution parameters $\mathfrak{m} = (\vec{\mu}_0, \mathbf{\Psi}, \kappa, \nu)$ and is computed by marginalizing over the likelihood parameter $\boldsymbol{\theta} = (\vec{\mu}, \mathbf{\Sigma})$, where $(\kappa, \nu) \in \mathbb{R}$, $(\vec{\mu}, \vec{\mu}_0) \in \mathbb{R}^n$ and $(\mathbf{\Sigma}, \mathbf{\Psi})$ are positive definite $\mathbb{R}^{n\times n}$ matrices:

$$p(\vec{y}_i|\mathfrak{m}) = \frac{p(\vec{y}_i|\boldsymbol{\theta}, \mathfrak{m})\,p(\boldsymbol{\theta}|\mathfrak{m})}{p(\boldsymbol{\theta}|\vec{y}_i, \mathfrak{m})} = \int \mathrm{d}\boldsymbol{\theta}\,p(\vec{y}_i|\boldsymbol{\theta})\,p(\boldsymbol{\theta}|\mathfrak{m})\,. \tag{31}$$

In our case of placing a NIW evidential prior on a multivariate Gaussian likelihood function, i.e.,

$$p(\vec{y}_i|\mathfrak{m}) = \int \mathrm{d}\boldsymbol{\theta}\,\mathcal{N}(\vec{y}_i|\boldsymbol{\theta})\ \texttt{NIW}(\boldsymbol{\theta}|\mathfrak{m})\,,$$

an analytical solution exists and can be parametrized with a multivariate t-distribution with $\nu - n + 1$ DoF, cf. Sec. C.3:

$$p(\vec{y}_i|\mathfrak{m}) = t_{\nu-n+1}\left(\vec{y}_i\middle|\vec{\mu}_0, \frac{1}{\nu-n+1}\frac{1+\kappa}{\kappa}\mathbf{\Psi}\right) \tag{32a}$$

$$= \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu-n+1}{2}\right)}\sqrt{\frac{\kappa^n}{(1+\kappa)^n}\frac{1}{\pi^n|\mathbf{\Psi}|}} \times \left(1 + \frac{\kappa}{1+\kappa}(\vec{y}_i - \vec{\mu_0})^\top \mathbf{\Psi}^{-1}(\vec{y}_i - \vec{\mu}_0)\right)^{-\frac{\nu+1}{2}}$$

$$= \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu-n+1}{2}\right)}\sqrt{\frac{\kappa^n}{(1+\kappa)^n}\frac{1}{\pi^n|\mathbf{\Psi}|}} \times \left(\frac{\left|\mathbf{\Psi} + \frac{\kappa}{1+\kappa}(\vec{y}_i - \vec{\mu_0})(\vec{y}_i - \vec{\mu}_0)^\top\right|}{|\mathbf{\Psi}|}\right)^{-\frac{\nu+1}{2}}, \tag{32b}$$

where we used Sylvester's determinant theorem,

$$|\mathbf{\Psi} + c(\vec{\mu} - \vec{\mu}_0)(\vec{\mu} - \vec{\mu}_0)^\top| = |\mathbf{\Psi}|\left(1 + c(\vec{\mu} - \vec{\mu}_0)^\top \mathbf{\Psi}^{-1}(\vec{\mu} - \vec{\mu}_0)\right) \tag{33}$$

with $c \in \mathbb{R}$, to derive Eq. (32b). Obviously, $p(\vec{y}_i|\mathfrak{m})$ on its own is not capable of defining $\mathfrak{m}$ unambiguously. In particular, a fitting approach could be used to find ν, $\vec{\mu}_0$ and the product $(1+\kappa)/\kappa\,\mathbf{\Psi}$ from data. However, in order to disentangle the latter additional constraints have to be set, e.g., via an additional regularization of κ.

Using this result we can compute the negative log-likelihood loss $\mathcal{L}_i^{\text{NLL}}$ for sample i as:

$$\begin{aligned}\mathcal{L}_i^{\text{NLL}} &= -\log p(\vec{y}_i|\mathfrak{m}) \\ &= \log\Gamma\left(\frac{\nu-n+1}{2}\right) - \log\Gamma\left(\frac{\nu+1}{2}\right) \\ &\quad + \frac{n}{2}\log\left(\pi\,\frac{1+\kappa}{\kappa}\right) - \frac{\nu}{2}\log|\mathbf{\Psi}| \\ &\quad + \frac{\nu+1}{2}\log\left|\mathbf{\Psi} + \frac{\kappa}{1+\kappa}(\vec{y}_i - \vec{\mu}_0)(\vec{y}_i - \vec{\mu}_0)^\top\right|\,,\end{aligned} \tag{34a}$$

or, alternatively, using a slightly different parametrization of $\mathfrak{m}$ with $\mathbf{\Psi} \equiv \nu\mathbf{\Sigma}_0$:

$$\begin{aligned}\mathcal{L}_i^{\texttt{NLL}} &= \log\Gamma\left(\frac{\nu-n+1}{2}\right) - \log\Gamma\left(\frac{\nu+1}{2}\right) \\ &+ \frac{n}{2}\log\left(\nu\pi\,\frac{1+\kappa}{\kappa}\right) - \frac{\nu}{2}\log|\mathbf{\Sigma}_0| \\ &+ \frac{\nu+1}{2}\log\left|\mathbf{\Sigma}_0 + \frac{\kappa}{1+\kappa}\frac{1}{\nu}(\vec{y}_i-\vec{\mu}_0)(\vec{y}_i-\vec{\mu}_0)^\top\right| . \end{aligned} \tag{34b}$$

We note that in the univariate case, $n = 1$, Eq. (32a) becomes a non-standardized Student's t-distribution with ν DoF:

$$\begin{aligned} p(y_i|\mathfrak{m}) &= \mathrm{St}_\nu\left(y_i\middle|\mu_0, \frac{1+\kappa}{\kappa}\sigma_0^2\right) \\ &= \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)}\sqrt{\frac{\kappa}{1+\kappa}\frac{1}{\pi\nu\sigma_0^2}}\left(1+\frac{\kappa}{1+\kappa}\frac{(y-\mu_0)^2}{\nu\sigma_0^2}\right)^{-\frac{\nu+1}{2}} \end{aligned} \tag{35a}$$

and Eq. (34a) reduces to

$$\begin{aligned}\mathcal{L}_i^{\texttt{NLL}} &= -\log p(\vec{y}_i|\mathfrak{m}) \\ &= \log\Gamma\left(\frac{\nu}{2}\right) - \log\Gamma\left(\frac{\nu+1}{2}\right) \\ &+ \frac{1}{2}\log(\pi/\kappa) - \frac{\nu}{2}\log(\nu\sigma_0^2(1+\kappa)) \\ &+ \frac{\nu+1}{2}\log(\nu\sigma_0^2(1+\kappa)+\kappa(y_i-\mu_0)^2) \end{aligned} \tag{35b}$$

and thus reproduces the findings of [2] with $\nu = 2\alpha$, $\nu\sigma_0^2 = 2\beta$, $\kappa = v$ and $\mu_0 = \gamma$. Similar to the multivariate case, $p(y, \mathfrak{m})$ on its own is not sufficient to define $\mathfrak{m}$ unambiguously.

C.3 Analytical derivation of the model evidence

One way to derive Eq. (32a) is, first, to use the fact that arguments can be shifted within the product of the two multivariate normal distributions $\mathcal{N}(\vec{y}_i|\vec{\mu}, \mathbf{\Sigma})$ and $\mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \mathbf{\Sigma}/\kappa)$,

$$\begin{aligned}\mathcal{N}(\vec{y}_i|\vec{\mu}, \mathbf{\Sigma}) \times \mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \mathbf{\Sigma}/\kappa) &= \mathcal{N}(\vec{\mu}\,|\vec{y}_i, \mathbf{\Sigma}) \times \mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \mathbf{\Sigma}/\kappa) \\ &= \mathcal{N}\left(\vec{y}_i\middle|\,\vec{\mu}_0, \frac{1+\kappa}{\kappa}\mathbf{\Sigma}\right) \times \mathcal{N}\left(\vec{\mu}\,\middle|\,\frac{\vec{y}_i+\kappa\vec{\mu}_0}{1+\kappa}, \frac{1}{1+\kappa}\mathbf{\Sigma}\right) , \end{aligned} \tag{36}$$

which we use to separate the integration parameter $\vec{\mu}_0$. Secondly, the normal distribution (26b) of a Normal-Inverse-Wishart distribution can be partially absorbed by the Inverse-Wishart distribution using a similar trick and Eq. (33),

$$\begin{aligned}\texttt{NIW}(\vec{\mu}, \mathbf{\Sigma}|\vec{\mu}_0, \mathbf{\Psi}, \kappa, \nu) &= \mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \mathbf{\Sigma}/\kappa) \times \mathcal{W}^{-1}(\mathbf{\Sigma}|\mathbf{\Psi}, \nu) \\ &= t_{\nu-n+1}\left(\vec{\mu}\,\middle|\,\vec{\mu}_0, \frac{1}{\nu-n+1}\mathbf{\Psi}/\kappa\right) \\ &\times \mathcal{W}^{-1}\left(\mathbf{\Sigma}\middle|\mathbf{\Psi} + \kappa(\vec{\mu}-\vec{\mu}_0)(\vec{\mu}-\vec{\mu}_0)^\top, \nu+1\right) , \end{aligned} \tag{37}$$

allowing for a straightforward integration of $\boldsymbol{\Sigma}$. Using both we can evaluate integral (31):

$$
\begin{aligned}
\int \mathrm{d}\boldsymbol{\theta}\, p(\vec{y}_i|\boldsymbol{\theta})\, p(\boldsymbol{\theta}|\mathfrak{m}) &= \iint \mathrm{d}\vec{\mu}_0\, \mathrm{d}\boldsymbol{\Sigma}\, \mathcal{N}(\vec{y}_i|\vec{\mu}, \boldsymbol{\Sigma}) \times \mathcal{N}(\vec{\mu}\,|\vec{\mu}_0, \boldsymbol{\Sigma}/\kappa) \times \mathcal{W}^{-1}(\boldsymbol{\Sigma}|\boldsymbol{\Psi}, \nu) \\
&\overset{(36)}{=} \int \mathrm{d}\boldsymbol{\Sigma} \left[\int \mathrm{d}\vec{\mu}\, \mathcal{N}\left(\vec{\mu} \,\middle|\, \frac{\vec{y}_i + \kappa\vec{\mu}_0}{1+\kappa}, \frac{1}{1+\kappa}\boldsymbol{\Sigma} \right) \right] \\
&\quad \times \mathcal{N}\left(\vec{y}_i \,\middle|\, \vec{\mu}_0, \frac{1+\kappa}{\kappa}\boldsymbol{\Sigma} \right) \times \mathcal{W}^{-1}(\boldsymbol{\Sigma}|\boldsymbol{\Psi}, \nu) \\
&\overset{(30a)}{=} \int \mathrm{d}\boldsymbol{\Sigma}\, \mathcal{N}\left(\vec{y}_i \,\middle|\, \vec{\mu}_0, \frac{1+\kappa}{\kappa}\boldsymbol{\Sigma} \right)\, \mathcal{W}^{-1}(\boldsymbol{\Sigma}|\boldsymbol{\Psi}, \nu) \\
&\overset{(29)}{=} \int \mathrm{d}\boldsymbol{\Sigma}\, \texttt{NIW}\left(\vec{y}_i, \boldsymbol{\Sigma} \middle| \vec{\mu}_0, \boldsymbol{\Psi}, \frac{\kappa}{1+\kappa}, \nu \right) \\
&\overset{(37)}{=} t_{\nu-n+1}\left(\vec{y}_i \,\middle|\, \vec{\mu}_0, \frac{1+\kappa}{\kappa(\nu-n+1)}\boldsymbol{\Psi} \right) \\
&\quad \times \int \mathrm{d}\boldsymbol{\Sigma}\, \mathcal{W}^{-1}\left(\boldsymbol{\Sigma} \middle| \boldsymbol{\Psi} + \frac{\kappa}{1+\kappa}(\vec{y}_i - \vec{\mu}_0)(\vec{y}_i - \vec{\mu}_0)^\top, \nu+1 \right) \\
&\overset{(30b)}{=} t_{\nu-n+1}\left(\vec{y}_i \,\middle|\, \vec{\mu}_0, \frac{1+\kappa}{\kappa(\nu-n+1)}\boldsymbol{\Psi} \right) .
\end{aligned}
$$

D Degeneration

To understand the degeneration of the model evidence due to the ambiguity of κ and $\mathbf{\Psi}$ it is helpful to look at the compact notation in Eq. (9): The higher-order evidential distribution is projected by integrating out the nuisance parameters $\vec{\mu}$ and $\mathbf{\Sigma}$ and, in the univariate case, the four DoF of $\mathfrak{m}$ collapse into three DoF of a scaled Student's t-distribution. Fitting this reduced set of DoF is not sufficient to recover all DoF of the evidential distribution.[8] The impact of this observation is that fitting the width of the t-distribution will not help to unfold κ and $\mathbf{\Psi}$ and it is possible to find manifolds with different values of κ and β but with the same value for the loss function $\mathcal{L}_i^{\text{NLL}}$ as proposed in [2]. In fact, κ can be tuned such that for any given value of $\mathcal{L}_i^{\text{NLL}}$ a value for $\mathbf{\Psi}$ or β can be find. Since the inference of the parameters of NNs is achieved by minimizing a loss function, $\mathcal{L}_i^{\text{NLL}}$, which in our case solely depends on the product $\beta(1+\kappa)/\kappa$ but not on β and κ individually, the contour lines in Fig. 6 correspond to configurations with constant value for $\mathcal{L}_i^{\text{NLL}}$ but different values for β and κ which we refer to as the *degeneration*. In tangible terms, each value of $\mathcal{L}_i^{\text{NLL}}$ maps to infinitely many aleatoric and epistemic uncertainties according to Eqs. (2). Hence, $\mathcal{L}_i^{\text{NLL}}$ on its own as proposed in [2], is not sufficient to learn the parameters $\mathfrak{m}$.

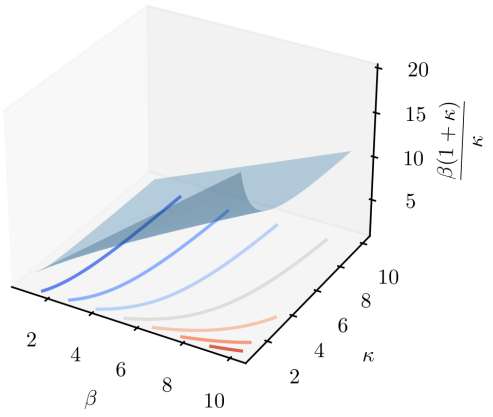


Figure 6: Parameters β and κ, as well as their combination as used in the univariate loss function. Contour lines in the β-κ plane indicate manifolds where the value of the loss function is constant.

[8] This is similar to the example of fitting the distribution of the sum of two i.i.d. drawn samples from normal distributions with common mean μ but different variance σ_1^2 and σ_2^2. Here, the result will follow a normal distribution with mean μ and variance $\sigma_1^2 + \sigma_2^2$ where it is impossible to unfold σ_1^2 and σ_2^2 just by using the distribution of the sum – the sum is a projection and one DoF is lost.

E Bias of fit parameters

To study possible biases of fits with Student's t-distributions in the parameters ν and κ we conduct a pseudo experiment where we generate data by drawing them i.i.d. from Student's t-distributions with fixed shape parameters (GT) and fit them on different sample sizes. For each sample size we evaluate 200 fits and indicate the values above (below) the 16 % (84 %) quantile in Fig. 7 as error bars after subtracting the GT value. This study shows a bias for ν and σ which decreases if the sample size increases.

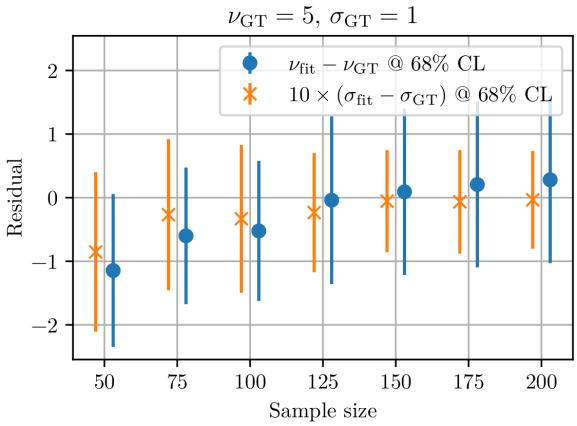


Figure 7: Residuals of the fitted parameter and the GT values used for generating the data. Here, a residual of zero corresponds to an unbiased estimate.

F Multivariate data sample

The data distribution used for the multivariate experiments outlined in Sec. 4 is shown in Fig. 8. The NNs are trained to predict $(x, y) \in \mathbb{R}^2$ given $t \in \mathbb{R}$. In total, 100 NNs of the same architecture are trained on the same data distribution. The density of the data in t varies and is maximal (minimal) at $t = \{0, 2\pi\}$ ($t = \pi$). On top, Gaussian noise is added to r but not φ where we defined:

$$\begin{pmatrix} r \\ \varphi \end{pmatrix} \overset{f}{\mapsto} \begin{pmatrix} x \\ y \end{pmatrix} = r \begin{pmatrix} \cos\varphi \\ \sin\varphi \end{pmatrix}. \tag{38}$$

Using

$$\mathrm{cov}(r, \varphi) = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & 0 \end{pmatrix} \tag{39}$$

the covariance of x and y can thus be estimated by using the Jacobian $\boldsymbol{J}_f$ of the mapping function f,

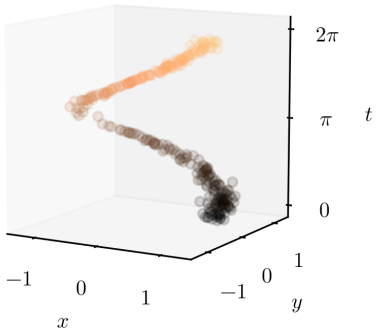


Figure 8: The distribution of our test data. The value of t is color coded (see Fig. 3a for a projection of the data into the xy-plane).

$$\boldsymbol{J}_f = \begin{pmatrix} \cos\varphi & -r\sin\varphi \\ \sin\varphi & r\cos\varphi \end{pmatrix}, \tag{40}$$

yielding

$$\mathrm{cov}(x, y) = \boldsymbol{J}_f \,\mathrm{cov}(r, \varphi)\, \boldsymbol{J}_f^T = \sigma_r^2 \begin{pmatrix} \cos^2\varphi & \sin\varphi\cos\varphi \\ \sin\varphi\cos\varphi & \sin^2\varphi \end{pmatrix}. \tag{41}$$

Note that this corresponds to a maximal correlation of x and y,

$$\mathrm{corr}(x, y) = \begin{cases} +1 & \text{for } 0 < t < \frac{\pi}{2}\,, \\ -1 & \text{for } \frac{\pi}{2} < t < \pi\,, \\ +1 & \text{for } \pi < t < \frac{3\pi}{2}\,, \\ -1 & \text{for } \frac{3\pi}{2} < t < 2\pi\,. \end{cases} \tag{42}$$

The covariance for x and y as well as their correlation are also learned by NNs and respective predictions are shown in Fig. 9 (up to a scaling constant). For regions with large statistic and constant correlation the expected behavior is well reproduced. During transitions due to sign flips of the correlation at $t_{\mathrm{sf}} = \{0.5\pi, \pi, 1.5\pi\}$ and in regions of large epistemic uncertainty the estimation of the covariance becomes unstable. We find that if the sample size is increased (and thus the epistemic uncertainty is sufficiently suppressed) the prediction of both values becomes stable for all values of t except for small finite regions centered at t_{sf}.

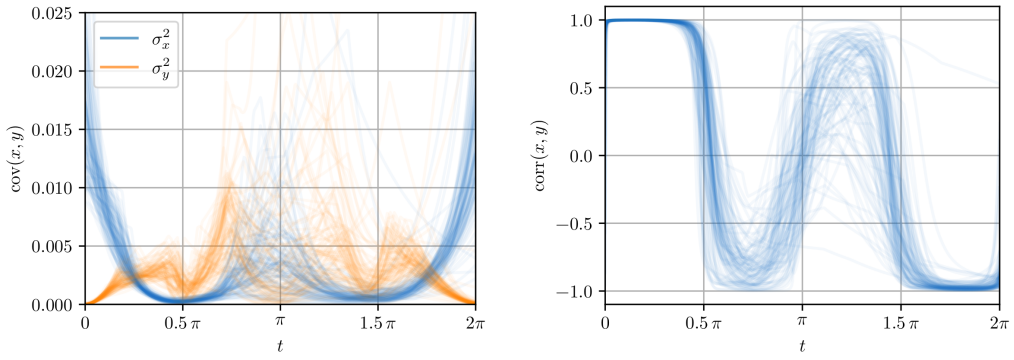


Figure 9: (Left) Overlayed covariance (up to a scaling constant) and (right) correlation of x and y as predicted by 100 NNs.