Author response: 'Stationary Activations for Uncertainty Calibration in Deep Learning' NeurIPS: #5154

We thank the anonymous reviewers for their enthusiasm and detailed comments on the manuscript. We summarise the reviews as positive, and the main concerns were related to clarifying the motivation and the experiments. We agree with the requests, and we will use the additional ninth page in the camera-ready paper for expanding the details as requested. We start by addressing R2's concerns as they had the lowest score.

R2: (1) The main motivation for this work is to establish understanding about the link between Matérn GP priors and neural network activation functions. This link is *explicit* as shown in this paper. A motive for this is to allow NN models to incorporate some of the appealing properties of GP models (*e.g.*, well-characterized uncertainties), while maintaining the flexibility and efficiency of NNs. The choice of kernel/activation function is up to the modelling task and 'expert knowledge'. We merely provide a building block. The Matérn is a widely used prior, and worth adding to the NN tool set. Stationarity encodes conservative behaviour suitable for uncertainty quantification (see R4(2)). **(2)** The references R2 provides ([1–3, 5]) are tackling a different problem, where the kernel is not used for encoding specific prior information, but inferred from data (*cf.*, 'automatic statistician'), and [4] is covered in this paper (limiting RBF case and NN kernel). **(3)** In the experiments, we originally reported only accuracy and NLPD (accounting for uncertainty). As requested, we added AUC to the results, which is in-line with the previous results (see table), with our proposed model outperforming the baselines. Comparison to SIREN is interesting (NB: the SIREN paper was put on arXiv *after* the submission DL), and to answer your question, we ran the experiments with it as well (see table). On the UCI tasks, SIREN performs comparably to our method. On the OOD image classification task, it performs clearly worse (known-class NLPD: 0.103 vs. 0.106, unknown-class NLPD: 0.896 vs. 1.62), but still better than the baselines. Note that SIREN encodes a different type of prior (infinite smoothness, like the RBF).

UCI classification tasks with Matérn-3/2 activation/kernel showing the AUC metric. Also results for SIREN activations included.

(10-fold cv)	n	d	c	SVGP AUC	GPDNN AUC	SV-DKL AUC	Matérn activ. AUC	SIREN activation NLPD	SIREN activation ACC	SIREN activation AUC
Adult	45222	14	2	.893±.004	.774±.052	.912±.003	**.913±.004**	.314±.006	.854±.005	.912±.004
Connect-4	67556	42	3	.824±.005	.675±.019	.909±.013	**.913±.004**	.449±.008	.825±.005	.909±.003
Covtype	581912	54	7	.971±.001	.943±.015	**.998±.000**	**.998±.000**	.119±.002	.957±.001	**.998±.000**
Diabetes	768	8	2	.817±.049	.769±.053	.512±.095	**.838±.051**	.487±.066	.771±.054	.835±.054

R1: (1) We are glad that the question about relation to kernel feature expansions was brought up. Fourier features (random, dense/structured, sparse) are typically leveraged for stationary kernels by projecting the GP problem on a set of harmonic basis functions. While we share the idea of using the Fourier duality, the resulting model is spanned by different basis functions; *e.g.*, sinusoidal FFs enforce (global) stationarity (approximation is based on Eq. (5)), while our approach is *locally* stationary as defined by the Gaussian weights in Eq. (2), which the approximation is based on. This discussion was left out in the interest of space, but the additional page will give us space to cover this. **(2)** We appreciate the suggestions for improving the visualizations. We did focus on real data (with three different real-world experiment setups) in our quantitative experiments, and the toy data examples were to give a general understanding. Your suggestion of varying the number of hidden units and MC samples is good and easy to do. We'll include this in the appendix to facilitate understanding of the effect of these parameters. We'll also run a GP on CIFAR-10 as suggested. **(3)** We recognize your concern with the term 'uncertainty calibration' (used here as GPs are commonly said to have representative uncertainties). We will replace it with the less loaded 'uncertainty quantification' where applicable.

R3: (1) The paper 'Deep Neural Networks as Gaussian Processes' (thank you for the reference) works in the scope of our Sec. 2 (connection from activation functions to GP kernels for principled inference). We take the opposite direction (GP prior $\rightarrow$ activation function) to encode properties of the GP prior into the NN. **(2)** Indeed, the SPDE link for constructing Gauss–Markov random fields from stationary kernels is relevant here. The deeper-level connection is related to the spectrum of the corresponding 'covariance operator' (the covariance function is formally the kernel of this operator). We agree that there is still a lot to uncover in this space. **(3)** The computational complexity of our model is on par with using other activation functions (say, a ReLU) in the NN models. In practice, we may converge slightly slower (*e.g.*, 53 s vs. 42 s for the results on the 'adult' data set). Compared to the GP models, especially with large data sets, we gain a considerable speed-up. **(4)** You are right that in this regard our approach can be thought of as a simplified Bayesian neural network—but yet as one, where the Matérn prior can be directly encoded.

R4: (1) We understand your comment on not dedicating attention to the inference methods (also raised by **R2**). This is true and mostly on purpose to retain focus. MC dropout is neither fast nor exciting, but does its job and unlikely introduces complications that would raise suspicions/complications with the model. Extending beyond it is left as future work. **(2)** In both Fig. 1–2 the number of hidden units is left small (also no ensembling or such used) to highlight the noisiness of the corresponding NN models. In Fig. 1, the uncertainty does not always increase right outside of the data range as it does for the GP models on the top. This property reflects the remaining suboptimality of the model. In Fig. 2, the trained models end up in different local optima (this could probably be tuned). Mean-reversion is characteristic for stationary models (outside data the model knows it's uncertain and reverts to the mean) and a desired property (also applies to the RBF fig). **(3)** The activation func. lengthscale parameter is fixed in all the NN experiments, because the preceding layer(s) take care of scaling the inputs, which serves the same purpose. We will discuss this in the main paper. **(4)** As discussed in Sec. 5, the only practical problems were encountered with the spiky (non-differentiable) Matérn-1/2 activation. In general, tuning the learning rate might also help prevent possible issues with convergence.