Depth Uncertainty in Neural Networks

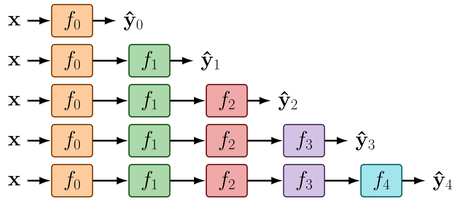
Javier Antorán*, James Urquhart Allingham*, José Miguel Hernández-Lobato
{ja666, jua23, jmh233}@cam.ac.uk

UNIVERSITY OF CAMBRIDGE

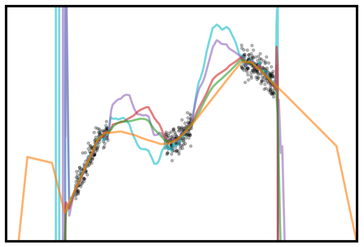
GitHub

Uncertainty over Depth

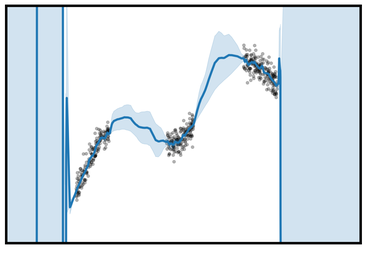
- How deep should our Neural Networks be?



- Deeper NNs express faster varying functions.



- Using probabilistic reasoning, we can translate depth uncertainty into function uncertainty.



Depth Uncertainty Networks (DUNs)

We treat depth d as a random variable with prior $p(d)$ and model weights θ as hyperparameters (Figure 1).

Tractable Exact Marginal Likelihood (MLL):

$$\log p(\mathcal{D};\boldsymbol{\theta}) = \log \Sigma_{i=0}^{D}\, p(d{=}i)p(\mathcal{D}|d{=}i;\boldsymbol{\theta})$$

In practise we target an ELBO:

$$\log p(\mathcal{D};\boldsymbol{\theta}) \geq \mathcal{L}(\boldsymbol{\alpha},\boldsymbol{\theta}) = -\mathrm{KL}(q_{\boldsymbol{\alpha}}(d)\,\|\,p(d)) + \sum_{n=1}^{N} \mathbb{E}_{q_{\boldsymbol{\alpha}}(d)}\left[\log p(\mathbf{y}^{(n)}|\mathbf{x}^{(n)},d;\boldsymbol{\theta})\right]$$

Single Forward Pass Inference:
The sequential nature of NNs allows to efficiently compute our objective and predictive posteriors: shallower nets pre-compute activations for deeper ones.

Probabilistic Model & Efficient Computational Model

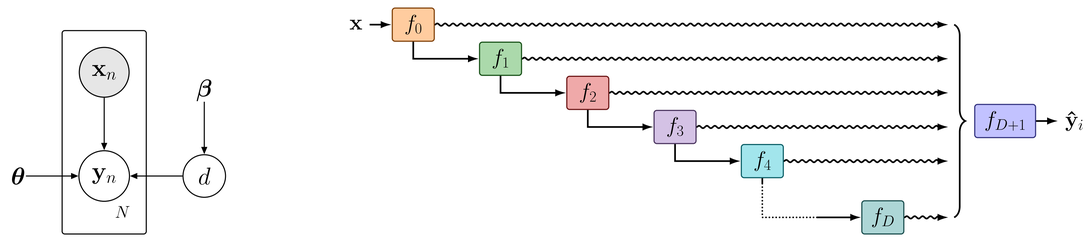


Figure 1: Left: graphical model. Right: computational model. Each block's activation is passed through the output block.

Model Uncertainty in Toy Regression

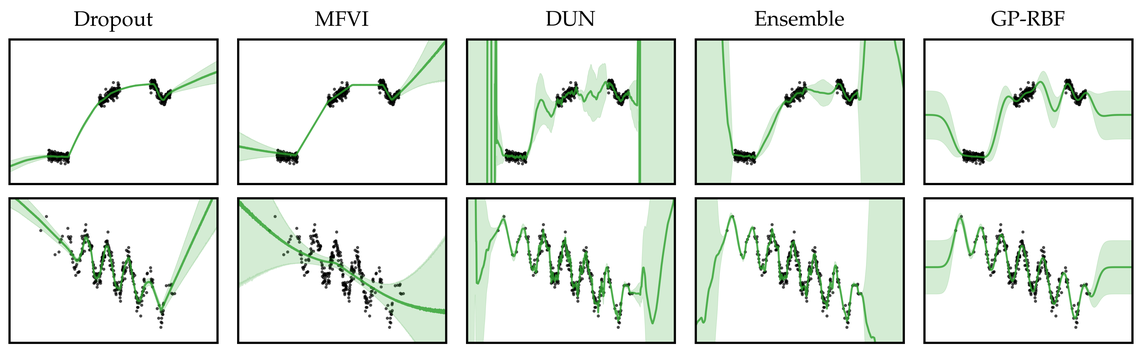


Figure 2: We train 3 hidden layer, 100 hidden unit, fully connected networks with residual connections for our baselines. DUNs use the same architecture but with 15 hidden layers. GPs use the RBF kernel. Error bars represent model uncertainty (standard deviation). DUNs and Ensembles fit the data well while providing rich uncertainty estimates in between clusters of data (black dots).

Image Classification under Distribution Shift (ResNet-50)

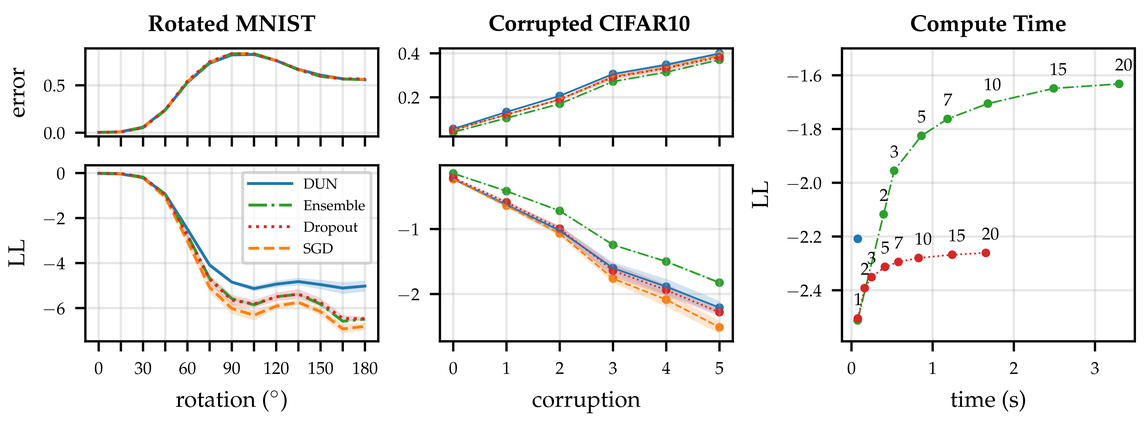


Figure 3: Left: error and LL for MNIST at varying degrees of rotation. Center: error and LL for CIFAR10 at varying corruption severities. Right: Pareto frontiers showing LL for corrupted CIFAR10 (severity 5) vs batch prediction time. DUNs are Pareto superior at all severities. Annotations show ensemble elements and dropout samples. We report results across 5 independent training runs.

DUN Neural Architecture Search

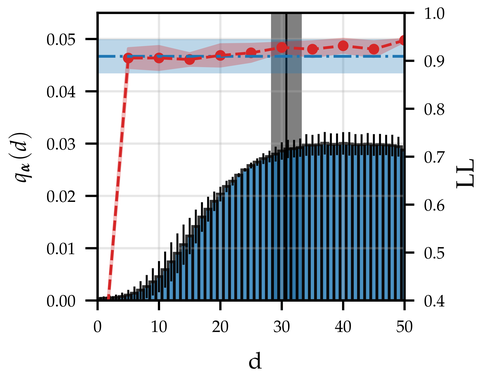


Figure 4: Approximate posterior over depth (navy) for a depth-50 DUN trained on SVHN. Test LLs obtained for standard NNs (red) of various depths are overlaid with those from a pruned DUN (blue), with the chosen depth (black).

Why VI Instead of MLE?

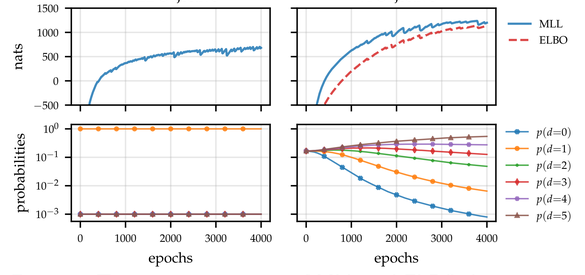


Figure 5: Top row: progression of MLL and ELBO during training. Bottom: progression of all depth posterior probabilities. The left column corresponds to optimising the MLL directly and the right to VI. For the latter, variational posterior probabilities $q(d)$ are shown. Optimising the ELBO results in higher MLL values than targeting the MLL directly.

Discussion

We have re-cast NN depth as a random variable, as opposed to a fixed parameter. NN weights are shared across depths and optimised as hyperparameters. These learn to capture diverse fits in a single NN. Both the model evidence and predictive posterior can be evaluated with a single forward pass. DUNs produce well calibrated uncertainty estimates, performing well relative to their computational budget on uncertainty-aware tasks. They scale to modern architectures and large datasets.