Dither is Better than Dropout for Regularising Deep Neural Networks

Andrew J.R. Simpson [#1]

[#] *Centre for Vision, Speech and Signal Processing, University of Surrey*
Surrey, UK
[1] Andrew.Simpson@Surrey.ac.uk

Abstract—**Regularisation of deep neural networks (DNN) during training is critical to performance. By far the most popular method is known as *dropout*. Here, cast through the prism of signal processing theory, we compare and contrast the regularisation effects of dropout with those of *dither*. We illustrate some serious inherent limitations of dropout and demonstrate that dither provides a far more effective regulariser which does not suffer from the same limitations.**

Index terms—**Deep learning, regularisation, dropout, dither.**

I. Introduction

In nonlinear signal processing, the use of additive noise prior to nonlinear processing (such as quantization or truncation) acts to decorrelate (or suppress) nonlinear distortion products. This process is known as *dithering* and can also be used in discrete signal processing to mitigate aliasing issues resulting from nonlinear distortion products which fall beyond the Nyquist limit.

Deep neural networks [1] may be interpreted as discrete (sampled) systems consisting of linear filters and nonlinear demodulation stages [2] and it has been suggested [3] that the inherent nonlinear distortion and aliasing contribute to problems of overfitting. Thus, in principle, if dither acts to suppress nonlinear distortion and aliasing it should also act to regularise a DNN.

At face value, *dropout* [4] appears somewhat compatible with dither and is known to be a useful regulariser. However, despite the cited motivation of 'preventing co-adaptation' [4], a coherent signal-processing-based rationale for dropout as regulariser has not emerged. Furthermore, the empirical results of dropout are typically conflated with those of stochastic gradient descent (SGD) and/or batch-averaged SGD and very little is known of possible dependencies or interactions between the two.

From a signal processing point of view, the gradients necessary for SGD constitute the output of a high-pass filter and the process of averaging these gradients over a batch constitutes a low-pass filter. In the ubiquitous batch-based SGD, this combination results in a band-pass filter. Therefore, batch size directly affects the bandwidth of the band-pass filter. This, in turn, constrains the process of SGD by preventing, to some degree, high-frequency (i.e., fine or abrupt) SGD steps from occuring.

In terms of sampling theory, although dropout acts similarly to dither in decorrelating nonlinear distortion products by perturbing the nonlinearity, it is not additive. A further critical difference between dropout and dither is that dropout discards a number of samples – a process that may be interpreted as stochastic decimation. One consequence of this is that dropout introduces distortion. A second consideration is that, according to sampling theory, decimation may only safely be performed following suitable low-pass filtering. Hence, dropout must introduce aliasing and distortion. Thus, unless this aliasing and distortion can be suppressed, dropout is likely to *cause* overfitting rather than prevent it. In other words, it is predictable that dropout applied with small or no batch averaging will result in *anti-regularisation*.

In this paper, we illustrate that dither provides regularisation which is independent of batch size, whilst the effect of dropout ranges from regularisation to anti-regularisation dependent upon the batch size.

Fig. 1. Example MNIST image. We took the 28x28 pixel images and unpacked them into a vector of length 784 to form the input at the first layer of the DNN.

II. Method

Regularisation is critical in the so-called 'small-data regime' – where the balance between parameters and data is skewed towards the parameters. For case study, we chose the well-known computer vision problem of hand-written digit classification using the MNIST dataset [5]. For the input layer we unpacked the images of 28x28 pixels into vectors of length 784. An example digit is given in Fig. 1. Pixel intensities were normalized to zero mean. Replicating Hinton's [6] architecture, but using the biased sigmoid activation function [2], we built a fully connected network of size 784x100x10

units, with a 10-unit softmax output layer, corresponding to the 10-way digit classification problem.

In order to place ourselves in the small-data regime, we used only the first 256 training examples of the MNIST dataset and tested on the full 10,000 test examples. We trained three versions of the model. The first version was trained without any regularisation. The second was trained with 50% dropout and the third version was trained with dither. For training with dither, uniform noise of unit scale and zero mean was added to the input (image only) data of each batch. The three classes of model were each independently instantiated and trained using SGD with batch sizes of 2, 4, 8, 16, 32, 64, 128 and 256 (i.e., 256 = full training set). Each separate model was trained for 100 full-sweep iterations of SGD (without momentum) and the test error computed (over the 10,000 test examples) at each iteration. For reliable comparison, each model was trained from the exact same random starting weights. A learning rate (SGD step size) of 1 was used for all training.

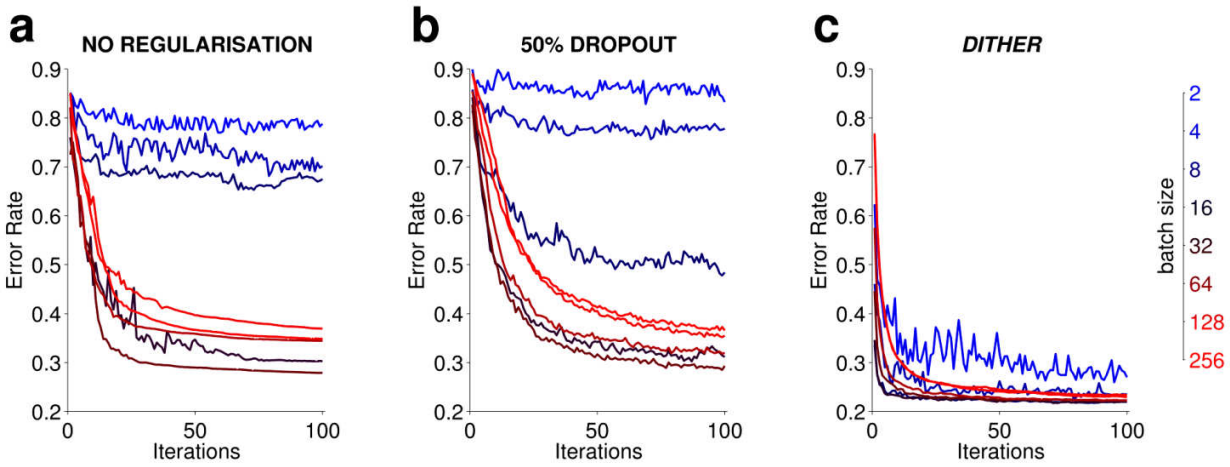

Fig. 2. Regularisation during training: Dropout Versus Dither. a plots the test error function of SGD iterations, for the various batch sizes, for the un-regularised models **b** plots the same for the models trained with 50% dropout **c** plots the same for the models trained with dither.

III. Results

Fig. 2a plots the test-error rates, as a function of full-sweep SGD iterations, for the un-regularised models of various batch sizes. Performance is dependent upon batch size; The model is essentially unable to learn anything useful when the batch size is less than 16 and peaks for batch size of 32. Fig. 2b plots the same for the models regularised using 50% dropout. As expected, performance is extremely dependent on batch size; Performance is substantially worse (than without dropout) for the batch sizes of 2 and 4, is improved (relative to no dropout) for the batch size of 8 and is similar (to performance without dropout) for the larger batch sizes. This tends to suggest that the regularisation provided by the data itself (at higher frequencies) is realised relatively well by simply averaging over larger batches (hence there is little evident advantage to dropout in this case).

Fig. 2c plots the test-error rate functions for the models trained with dither. As expected, relatively little dependence on batch size is in evidence and, in all cases, both learning rate and ultimate performance is starkly superior to dropout.

Across all the models, there is a general trend for the batchsize of 32 to perform best (more obviously in the non-dithered cases). This tends to suggest that the data itself regularises best when averaged over batches of 32 and this probably relates to the nature of the data.

In summary, without dither the models at small batch sizes failed to learn anything useful and dropout made matters worse. Thus, the prediction (derived from signal processing theory) of dropout resulting in anti-regularisation for small batch sizes appears to have been confirmed. However, with dither, the same models trained (with the same batch sizes) were able to achieve an impressive nearly 80%-correct on the test set, despite only 256 training examples.

IV. Discussion and Conclusion

In this paper, we have demonstrated that dither is a superior regulariser to dropout and that, unlike dropout, the regularisation provided by dither is more or less independent of batch size. We have argued that dither is superior to dropout as regulariser due to the fact that it is not dependent upon batch size and due to the fact that it is inherently wideband and additive. We have also documented, for the first time, paradoxical anti-regularisation effects of dropout at small batch sizes.

Acknowledgment

AJRS did this work on the weekends and was supported by his wife and children.

REFERENCES

[1] Bengio Y (2009) “Learning deep architectures for AI”, Foundations and Trends in Machine Learning 2:1–127.

[2] Simpson AJR (2015) “Abstract Learning via Demodulation in a Deep Neural Network”, arxiv.org abs/1502.04042

[3] Simpson AJR (2015) Over-Sampling in a Deep Neural Network, arxiv.org abs/1502.03648

[4] Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov R (2012) “Improving neural networks by preventing co-adaptation of feature detectors”, The Computing Research Repository (CoRR), abs/1207.0580.

[5] LeCun Y, Bottou L, Bengio Y, Haffner P (1998) “Gradient-based learning applied to document recognition”, Proc. IEEE 86: 2278–2324.

[6] Hinton GE, Osindero S, Teh Y (2006). “A fast learning algorithm for deep belief nets”, Neural Computation 18: 1527–1554.