Dendritic normalisation improves learning in sparsely connected artificial neural networks

A D Bird[1,2*], H Cuntz[1,2]

[1] Frankfurt Institute for Advanced Studies, Frankfurt-am-Main, 60438, Germany

[2] Ernst Strüngmann Institute (ESI) for Neuroscience in cooperation with the Max Planck Society, Frankfurt-am-Main, 60528, Germany

*bird@fias.uni-frankfurt.de

Abstract

Inspired by the physiology of neuronal systems in the brain, artificial neural networks have become an invaluable tool for machine learning applications. However, their biological realism and theoretical tractability are limited, resulting in sometimes slow and awkward parameter fitting to training data. We have recently shown that biological neuronal firing rates in response to distributed inputs are largely independent of size, meaning that neurons are typically responsive to the proportion, not the absolute number, of their inputs that are active. Here we introduce such a normalisation, where the strength of a neuron's afferents is divided by their number, to various sparsely-connected artificial networks. The learning performance is dramatically increased. The resulting machine learning tools are universally applicable and biologically inspired, rendering them more stable and better understood.

Keywords

Artificial neural network; Sparse networks; Normalisation; Dendrites; Dendritic constancy

Introduction

Artificial neural networks have had a huge impact over the last couple of decades, enabling substantial advances in machine learning for fields such as image recognition (Krizhevsky et al, 2012), translation (Sutskever et al, 2014), and medical diagnosis (Ardila et al, 2019). The inspiration for these tools comes from biological neuronal networks, where learning arises from changes in the strength of synaptic connections between neurons through a number of different plasticity mechanisms (McCulloch & Pitts, 1943; Hebb, 1949; Hopfield et al, 1983). The growth of neural networks away from the limitations of biological networks, for example the use of global backpropagation algorithms that have access to information unavailable to real synaptic connections (Rumelhart et al, 1986), has meant that state-of-the-art artificial intelligence algorithms differ fundamentally from the biological function of the brain. Nevertheless, a number of biophysical principles have been successfully reintroduced, using salient features of real neuronal networks to make advances in the field of artificial neural networks (LeCun et al, 1990; Maass, 1997; Gütig & Sompolinsky, 2006; Gütig, 2016; Kim & Chow, 2018; Krotov &

Hopfield, 2019). Here we show how the dendritic morphology of a neuron, which influences both its connectivity and excitability, produces a weight normalisation that improves learning in such networks.

Real neurons receive synaptic contacts across an extensively branched dendritic tree. Dendrites are leaky core conductors, where afferent currents propagate along dendritic cables whilst leaking across the cell membrane (Rall, 1962). Larger dendrites increase the number of potential connections a cell can receive, meaning that more afferent currents can contribute to depolarisation (Chklovskii, 2004). Conversely, larger cells typically have lower input resistances, due to the increased spatial extent and membrane surface area, meaning that larger synaptic currents are necessary to induce the same voltage response and so bring a neuron to threshold (Rall, 1957; Mainen & Sejnowski, 1996). It has recently been shown by Cuntz et al (2019) that these two phenomena cancel each other exactly: the excitability of neurons receiving distributed excitatory synaptic inputs is largely invariant to changes in size and morphology. In addition, neurons possess several active mechanisms to help maintain firing-rate homeostasis through both synaptic plasticity regulating inputs (Abbott & Nelson, 2000; Royer & Paré, 2003) and changes in membrane conductance regulating responses (Gorur-Shandilya et al, 2019). These results imply a consistent biophysical mechanism that contributes to stability in neuronal activity despite changes in scale and connectivity.

Changing connectivity has traditionally not played much of a role in artificial neural networks, which typically used fully-connected layers where each neuron can receive input from all cells in the preceding layer. Sparsely-connected layers have however long been used as alternatives in networks with a variety of different architectures (LeCun et al, 1990). Sparse connectivity more closely resembles the structure of real neuronal networks and a number of recent studies have demonstrated that larger, but sparsely-connected, layers can be more efficient than smaller fully-connected layers both in terms of total parameter numbers and training times (Louizos et al, 2017; Mocanu et al, 2018). The advantage in efficiency comes from the ability to entirely neglect synaptic connections that do not meaningfully contribute to the function of the network. Sparse networks are also less likely to be overfitted to their training data as sparse representations of inputs are forced to focus on essential features of the signal instead of less-informative noise.

To produce an appropriate sparse connectivity a number of regularisation techniques have been suggested; L^1- and L^0-regularisations (Tibshirani, 1996; Louizos et al, 2017) both penalise (the latter more explicitly) the number of connections between neurons during training. Mocanu et al (2018), building on previous work (McDonnell & Waagen, 1993; Stanley & Miikkulainen, 2002; Whiteson & Stone, 2006), have recently introduced an evolutionary algorithm to reshape sparse connectivity, with weak connections being successively excised and randomly replaced. This procedure causes sparse artificial networks to develop small-world and scale-free topologies similar to biological neuronal circuits (Bassett & Bullmore, 2017) and has comparable performance to fully-connected layers despite having many fewer parameters to optimise.

Normalisation is another feature that has previously been shown to enhance learning in neural networks. In particular Ioffe & Szegedy (2015) introduced batch normalisation, where the inputs over a given set of training data are normalised, and Salimans & Kingma (2016) introduced L^2-normalisation, where afferent synaptic weights are normalised by their total magnitude. The latter is reminiscent of heterosynaptic plasticity, where afferent synapses across a neuron depress in response to potentiation at one contact in order to maintain homeostasis (Royer & Paré, 2003). Both techniques have been applied in fully-connected networks and both work to keep neuronal activities in the region where they are most sensitive to changes in inputs. The normalisation that arises from the relationship between a real neuron's morphology and connectivity sits in this context and provides a particularly powerful, and biologically realistic, way to normalise sparse inputs. Dividing the magnitude of individual synaptic weights by their number distributes activity across neurons whilst keeping each cell sensitive to changes in inputs; neurons therefore encode signals from the proportion of presynaptic partners that are active, providing a simple and broadly applicable technique to ensure faster convergence to optimal solutions.

Methods and Models

Network architectures

We first consider a simple ANN with one hidden layer to demonstrate the utility of our approach. The size of the input layer for both the MNIST and MNIST-Fashion datasets is 784, as each image is a 28×28 pixel greyscale picture. The hidden layer consists of M neurons, each neuron i receiving a number n_i contacts from the previous layer. In **Figure 1**, $M = 30$, 100, and 300. In **Figures 2 and 3**, $M = 100$. Neuronal activation in the input and hidden layers as a function of input z_i is controlled by a sigmoid function $\sigma(z_i)$

$$\sigma(z_i) = \frac{1}{1 + e^{-z_i}} \tag{1}$$

where z_i is the weighted input to neuron i, given by

$$z_i = b_i + \sum_{n_i} w_{k,i} a_k \tag{2}$$

Here b_i is the bias of each neuron i, $w_{k,i}$ is the synaptic weight from neuron k in the previous layer to neuron i, and $a_k = \sigma(z_k)$ is the activation of presynaptic neuron k. The set of all $w_{k,i}$ for a given postsynaptic neuron i form an afferent weight vector $\mathbf{w}_i$.

Both datasets have ten classes and the output of the ANN is a probability distribution assigning confidence to each possible classification. Neurons in the output layer are represented by softmax neurons where the activation function $\sigma_s(z_i)$ is given by

$$\sigma_s(z_i) = \frac{e^{z_i}}{\sum_{i=1}^{10} e^{z_i}} \tag{3}$$

The cost function C is taken to be the log-likelihood

$$C = -\log(a_{\text{Correct}}) \tag{4}$$

where a_{Correct} is the activation of the output neuron corresponding to the correct input.

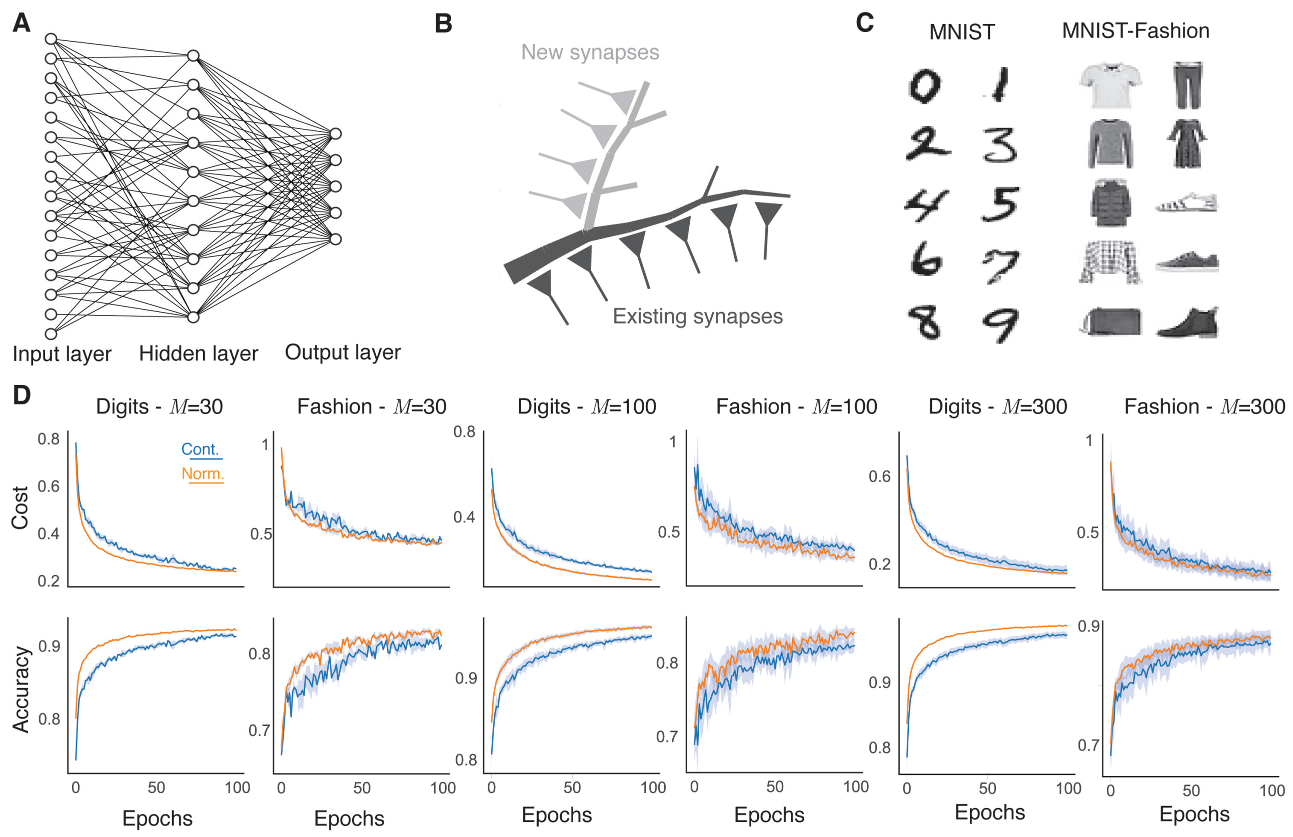

Fig 1. Dendritic normalisation improves learning in sparse artificial neural networks A, Schematic of a sparsely-connected artificial neural network. Input units (left) correspond to pixels from the input. Hidden units (centre) receive connections from some, but not necessarily all, input units. Output units (right) produce a classification probability (**Eq 4**). **B**, Schematic of dendritic normalisation. A neuron receives inputs across its dendritic tree (dark grey). In order to receive new inputs, the dendritic tree must expand (light grey), lowering the intrinsic excitability of the cell through increased membrane leak and spatial extent. **C**, Example 28×28 pixel greyscale images from the MNIST (left) and MNIST-Fashion (right) datasets. The MNIST images are handwritten digits from 0 to 9 and the MNIST-Fashion images have ten classes, respectively: T-shirt/top, trousers, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot. **D**, Learning improvement with dendritic normalisation (orange) compared to the unnormalised case (blue). Top row: Log-likelihood cost on training data (**Eq 4**). Bottom row: Classification accuracy on test data. From left to right: digits with $M = 30$ hidden neurons, fashion with $M = 30$, digits with $M = 100$, fashion with $M = 100$, digits with $M = 300$, fashion with $M = 300$. Solid lines show the mean over 10 trials and shaded areas the mean ± one standard deviation. SET hyperparameters are $\varepsilon = 0.2$ and $\zeta = 0.15$.

For **Figure 3**, we generalise our results to deeper architectures and threshold-linear neuronal activations. In **Figures 3a and 3c** we expand the above to include 2 and 3 sparse hidden hidden layers, each with $M = 100$ sigmoid neurons. In **Figures 3b and c** we consider a simple convolutional neural network (LeCun et al, 2009) with 20 5×5 features and 2×2 maxpooling. In **Figure 3d** we return to the original architecture with $M = 100$, but replace the sigmoid activation function $\sigma(z)$ for the hidden neurons with a non-saturating threshold-linear activation function $\tau(z)$ defined by

$$\tau(z_i) = \max(0, z) \tag{5}$$

In all cases traditional stochastic gradient descent (Robbins & Monro, 1951; LeCun et al, 1998) is used with a minibatch size of 10 and a learning rate η of 0.05.

Sparse evolutionary training (SET)

Connections between the input and hidden layers are established sparsely (**Figure 1a**) using the sparse evolutionary training algorithm introduced by Mocanu et al (2018). Briefly, connections are initiated uniformly randomly with probability ε to form an Erdős-Rényi random graph (Erdős & Rényi, 1959). After each training epoch, a fraction ζ of the weakest contacts are excised and an equal number of new random connections are formed. New connections are distributed normally with mean 0 and standard deviation 1.

MNIST and MNIST-Fashion datasets

The ANN is trained to classify 28×28 pixel greyscale images into one of ten classes. Two distinct datasets are used. The MNIST, introduced by LeCun et al (1998), consists of handwritten digits which must be sorted into the classes 0 to 9 (**Figure 1b**, left). The MNIST-Fashion dataset was introduced by Xiao et al (2017) as a direct alternative to the original MNIST and consists of images of clothing. The classes here are defined as T-shirt/top, trousers, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot (**Figure 1b**, right). Each dataset contains $60,000$ training images and $10,000$ test images. State-of-the-art classification accuracy for the original MNIST dataset is as high as 99.77% (Cireşan et al, 2012), which likely exceeds human-level performance due to ambiguity in some of the images. For the newer MNIST-Fashion dataset state-of-the art networks can achieve classification accuracies of 96%. Such performance is achieved with deep network architectures, which we do not reproduce here, rather showing an improvement in training between comparable, and comparatively simple, artificial neural networks.

Code and data availability

All code is written in Python 3.6 and is freely available for download (see **Supplementary File 1**) alongside the MNIST and MNIST-Fashion datasets. These can also be downloaded from various places, including at the time of writing `yann.lecun.com/exdb/mnist/` and `github.com/zalandoresearch/fashion – mnist` respectively. The convolutional network in **Figures 3b and d** makes use of Keras with a TensorFlow backend (`keras.io`).

In addition, we applied dendritic normalisation in Keras with TensorFlow for immediate inclusion in Keras-based deep learning models. The normalisation requires a custom layer, constraint, and optimiser. This is included as **Supplementary File 2**.

Results

Dendritic normalisation and stochastic gradient descent

Let $\mathbf{w}_i$ be the input weight vector to a given neuron i. Then the dendritic normalisation can be written as

$$\mathbf{w_i} = \frac{\mathbf{s}}{\|\mathbf{v_i}\|_\mathbf{0}}\mathbf{v}_i \tag{6}$$

where $\mathbf{v}_i$ is a vector of the same size as $\mathbf{w_i}$, $\|\mathbf{v}\|_\mathbf{0}$ is the L^0-norm of $\mathbf{v}_i$ (the number of non-zero elements), and s is a scalar that determines the magnitude of the vector $\mathbf{w}_i$. This normalisation arises from the fact that, for real neurons, increased synaptic connectivity is balanced by the additional leak conductance and spatial extent of the new dendritic cables required to collect the inputs (**Figure 1b**). The parametrisation here differs from that introduced by Salimans & Kingma (2016) for fully-connected networks with Euclidean normalisation in two fundamental ways. Firstly, the magnitude parameter s is the same across all neurons as it reflects a conserved relationship between connectivity and excitability. If a network were to include distinct classes of artificial neurons with distinct synaptic integration properties, different values of s may be appropriate for each class, but should not differ between neurons of the same class. Secondly, the L^0-norm $\|\mathbf{v_i}\|_\mathbf{0}$ is distinct from the Euclidean L^2-norm in that it is almost surely constant with respect to $\mathbf{v}_i$: Connections are not created or destroyed by stochastic gradient descent.

The gradients of the cost function C with respect to $\mathbf{v}_i$ and s can be written as

$$\nabla_{\mathbf{v}_i} C = \frac{s}{\|\mathbf{v}_i\|_0} \nabla_{\mathbf{w}_i} C \qquad , \qquad \frac{\partial C}{\partial s} = \sum_{\mathbf{w}_i} \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|_0} \nabla_{\mathbf{w}_i} C \tag{7}$$

where $\nabla_{\mathbf{w}_i} C$ is the usual gradient of C with respect to the full weight vector $\mathbf{w}_i$ and the sum in the second equation is over all weight vectors in the network. An interesting consequence of these equations is that neurons with more afferent connections will have smaller weight updates, and so be more stable, than those with fewer afferent connections.

Dendritic normalisation improves learning in neural networks

We consider the performance of sparse neural networks with and without dendritic normalisation on the MNIST and MNIST-Fashion datasets (**Figure 1**). For sparse networks using sparse evolutionary training (SET) with one hidden layer consisting of 30, 100, and 300 neurons, with connection probability $\varepsilon = 0.2$ and SET excision rate $\zeta = 0.15$, the normalised network consistently learns faster than the unnormalised control network (orange lines against blue lines). This result holds across both the cost on the training sets and the accuracy on the test sets for both datasets and across all network sizes, indicating a robust improvement in performance. In addition, the variability between different independent training regimes (shaded areas in **Figure 1d** show mean plus or minus one standard deviation over 10 independently initiated training regimes) is reduced, dendritic normalisation therefore also makes training more robust and reliable.

Evolution of connectivity

It is possible to visualise the connection structure that results from training the control and normalised networks on the MNIST data (**Figure 2**). **Figure 2a** shows how the number of efferent connections from each input neuron (organised as pixels) changes with the number of training epochs. Initially, the connections are randomly distributed and have no spatial structure, but the SET algorithm gradually imposes a heavier weighting on the central input neurons as training progresses. This feature was shown before by Mocanu et al (2018) as central pixels are likely to be more informative over the relevant datasets. Comparing the control (left, blue) and normalised (right, orange) networks, it is interesting to note that the bias towards central pixels is less strong in the normalised case: Input neurons over a relatively broad area are strongly connected to the hidden layer.

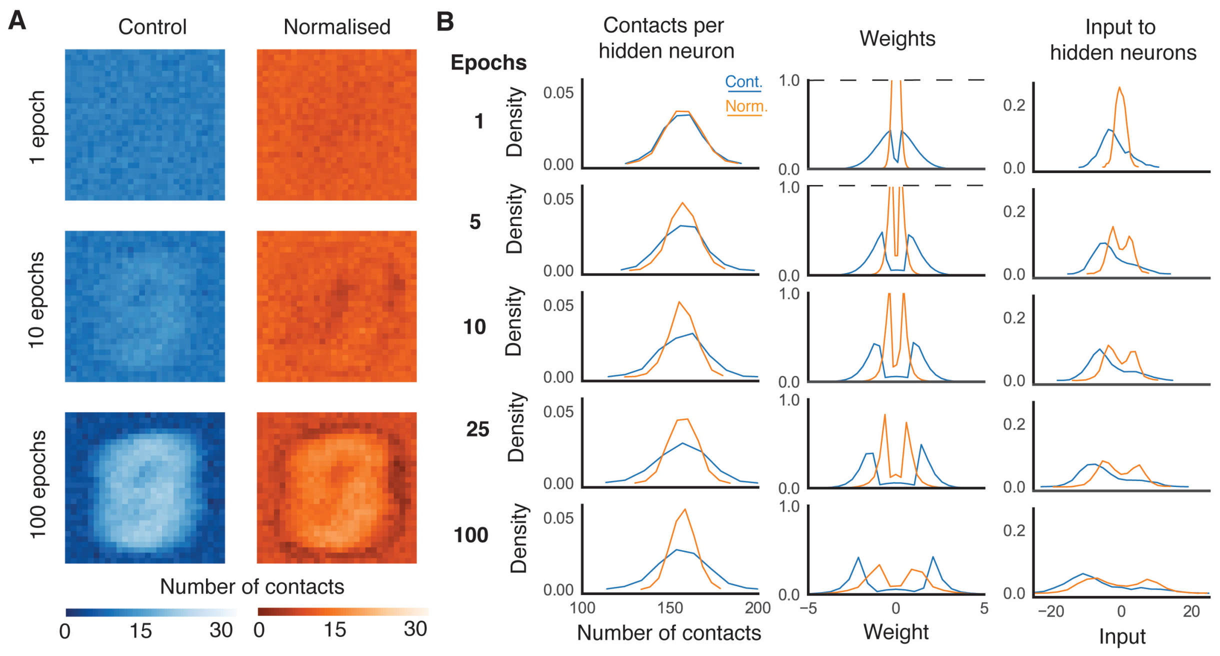

Fig 2. Evolution of synaptic weights. A, Number of efferent contacts from each input neuron (pixel) to neurons in the hidden layer as the weights evolve. The left panels (blue) show the unnormalised case and the right (orange) the normalised case. **B**, Afferent contacts for the unnormalised (blue) and normalised (orange) cases. From left to right: Distribution of the number of afferent contacts arriving at each hidden neuron, weights, and mean weighted input to each hiddden neuron over the test set. All panels show the average over 10 trials on the original MNIST dataset with hyperparameters $M = 100$, $\varepsilon = 0.2$, and $\zeta = 0.15$. Dashed lines show where the vertical axis has been truncated to preserve the scale.

Postsynaptically, the number of contacts received by each hidden neuron is less variable in the normalised case (**Figure 2b**, left column), the weights of these contacts are typically smaller in absolute value and less dispersed (**Figure 2b**, central column); the resultant weighted inputs over the test data to hidden neurons are therefore more consistent (**Figure 2b**, right column). The normalisation appears to make better use of neural resources by distributing connections and weights more evenly across the available cells, whilst keeping expected inputs closer to 0, the steepest part of the activation function, where responses are most sensitive to inputs. In addition, the smaller connection weights suggest that normalised networks may be even more robust to overfitting than the equivalent unnormalised sparse network. This is supported by the increased improvement in the case of more complex networks, both in terms of more hidden units and more layers, as well as the greater improvement in evaluation accuracy compared to training cost (**Figures 1 and 3**).

Improved training in deeper networks

The improvement in learning performance generalises to deeper networks with multiple hidden layers (**Figure 3a**, as well as convolutional networks with a sparsely connected layer following the max pooling layer (**Figure**

3b). In all cases, learning is improved by dendritic normalisation (orange versus blue lines in **Figures 3c and d** for the MNIST-Fashion dataset) in a similar manner to that for the single-layered networks described above. Indeed, the improvement seems to increase with the complexity of the architecture. Dendritic normalisation is therefore applicable, and beneficial, as a universal technique for sparse layers in deep networks.

The improvement in performance is not limited to artificial neurons with a sigmoid activation function. When neurons instead have non-saturating threshold linear activations (Eq 5) the dendritic normalisation again improves learning (**Figure 3e**).

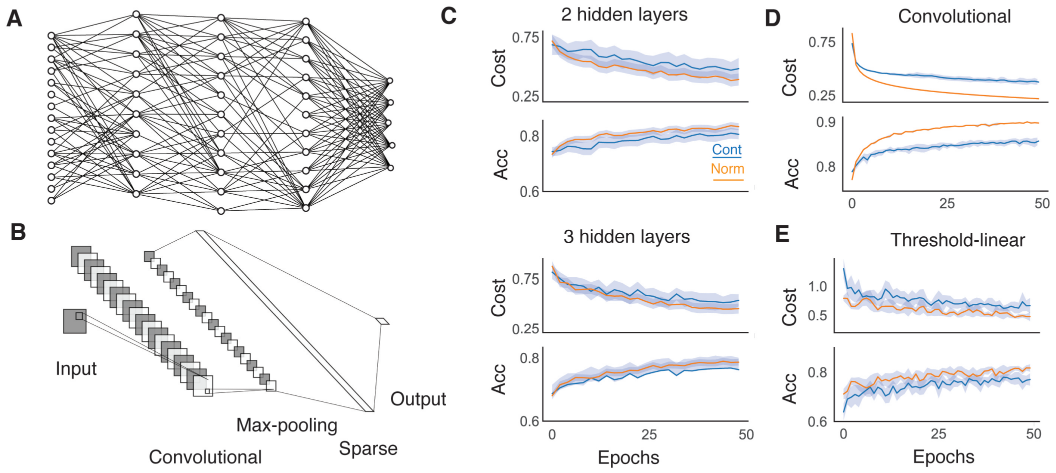

Fig 3. Improved training in deeper networks. A Schematic of a sparsely connected network with 3 hidden layers. The output layer is fully connected to the final hidden layer, but all other connections are sparse. **B** Schematic of a convolutional neural network with 20 5×5 features and 2×2 maxpooling, followed by a sparsely connected layer with $M = 100$ neurons. **C** Learning improvement with dendritic normalisation (orange) compared to the unnormalised control case (blue) for networks with 2 (top) and 3 (bottom, see panel **A**) sparsely-connected hidden layers, each with $M = 100$ neurons. Top of each: Log-likelihood cost on training data (Eq 4). Bottom of each: Classification accuracy on test data. **D** Improved learning in the convolutional network described in **B** for an unnormalised (blue) and normalised (orange) sparsely-connected layer. Top: Log-likelihood cost on training data (Eq 4). Bottom: Classification accuracy on test data. **E** Improved learning in a network with one hidden layer with $M = 100$ threshold-linear neurons (Eq 5) for unnormalised (blue) and normalised (orange) sparsely-connected layers. Top: Log-likelihood cost on training data (Eq 4). Bottom: Classification accuracy on test data. Solid lines show the mean over 10 trials and shaded areas the mean $\pm$ one standard deviation. All results are on the MNIST-Fashion dataset. Hyperparameters are $\varepsilon = 0.2$ and $\zeta = 0.15$.

Discussion

We have shown that normalising afferent synaptic weights by their number (L^0-normalisation), in a manner similar to real neurons (Cuntz et al, 2019), improves the learning performance of sparse artificial neural networks with a variety of different structures. Such dendritic normalisation constrains the weights and expected inputs to be within relatively tight bands, potentially making better use of available neuronal resources. Neurons respond more to the proportion of their inputs that are active and highly-connected neurons are relatively less excitable.

We believe that such a normalisation procedure is robust and should be applied to improve the performance of feedforward sparse networks.

Other results on normalisation (Ioffe & Szegedy, 2015; Salimans & Kingma, 2016) have also demonstrated improvements in training performance in fully-connected feedforward networks. Such approaches work by keeping neurons relatively sensitive to changes in inputs and our results here can be seen as the sparse, and biologically justified, analogue, with similarly simple and broad applicability to the L^2-normalisation introduced by Salimans & Kingma (2016). The comparison between the heterosynaptic plasticity-like L^2-normalisation and our dendritic L^0-normalisation is particularly interesting. In real neurons the former relies on actively re-weighting afferent contacts (Royer & Paré, 2003) whereas the latter arises purely from the passive properties of dendritic trees. Neurons often display complementary functionality between passive structure and active processes, for example in the equalisation of somatic responses to synaptic inputs at different locations both dendritic diameter taper (Bird & Cuntz, 2016) and active signal enhancement (Rumsey & Abbott, 2006) play a role. Synaptic normalisation is in a similar vein. The effects are, however, distinct in some ways: while both normalisations keep neurons sensitive to inputs, the responses to learning differ. Heterosynaptic plasticity enhances changes in the relative weighting of contacts, whereas dendritic normalisation increases the stability of well-connected neurons while allowing faster learning in poorly connected cells (Eq 7). This makes dendritic normalisation particularly suited to situations with evolving connectivity.

In the context of biological realism, the normalisation here has room for development. We sought a straightforwardly demonstrable and quantifiable computational role for the dendritic normalisation described in Cuntz et al (2019) and so used the most well-developed theories within artificial neural networks (Richards et al, 2019). Such networks have a number of features that are impossible to implement in the brain, so more investigation into the benefits of size-invariant excitability in living systems is necessary. Firstly, a single artificial neuron can form connections that both excite and inhibit efferent cells, a phenomenon which is not seen in the connectivity of real neurons (Dale, 1935). It is possible to regard the mix of excitatory and inhibitory connections as a functional abstraction of real connections mediated by intermediate inhibitory interneurons (Silberberg & Markram, 2007), but a more satisfying picture may emerge by considering distinct inhibitory populations of cells. Secondly, we train our networks using supervised backpropagation which employs global information unavailable to real synaptic connections. A large number of studies have developed more plausible local algorithms for training networks (Hopfield et al, 1983; Gütig & Sompolinsky, 2006; Kim & Chow, 2018; Krotov & Hopfield, 2019). Such algorithms are a natural fit for our normalisation as it too is implemented biophysically through the morphology of the dendritic tree; an immediate goal is to incorporate our result into algorithms on unsupervised and reinforcement learning. Thirdly, our networks are strictly feedforward whereas typical neuronal networks possess recurrent connections. Many studies of such networks employ sparsity in recurrent excitatory connections (Lazar et al, 2009; Kim & Chow, 2018) and this is another natural arena for our normalisation; here the two streams of input would be in competition as more strongly recurrently connected cells would receive relatively weak feedforward inputs and vice versa. This is a particularly interesting avenue to explore. Fourthly, the neurons here are rate-based with either saturating or non-saturating outputs. Spiking networks can have different properties properties (Maass, 1997) and spikes could be incorporated into any of the approaches described above.

Dendrites in general have much more to offer in terms of artificial neural computations. Synaptic connections are distributed spatially over branched dendritic trees, allowing for a number of putative computational operations to occur within a single cell (London & Häusser, 2005). Dendrites are able to passively amplify signals selectively based on their timing and direction (Rall, 1962) and actively perform hierarchical computations over branches (Poirazi et al, 2003). Cuntz et al (2019) noted that while mean neuronal firing rates are size-independent, the timing of individual spikes is not necessarily unaffected by morphology. This means that signals encoded by rates are normalised whereas those encoded by spike timing may not be, implying that the two streams of information across the same circuit pathways are differentially affected by changing connectivity. Dendrites additionally hold continuous variables through their membrane potential and conductances that shape ongoing signal integration (Gidon & Segev, 2012). Such properties have potential computational roles that, while sometimes intuitive, have yet to be systematically studied at the level of neuronal circuits.

Overall, this study has two major consequences. The first is a practical normalisation procedure that drastically improves learning in sparse artificial neural networks trained using backpropagation. The procedure can be used effectively for any sparsely-connected layers in shallow or deep networks of any architecture. Given that sparse layers can display better scalability than fully-connected layers (Mocanu et al, 2018), we believe that this procedure could become standard in deep learning. Furthermore, the biological plausibility of the procedure means that it is highly appropriate as a component of more physiologically realistic learning rules. Secondly, we have taken the insights from Cuntz et al (2019) and demonstrated a previously unappreciated way that the structure of dendrites, in particular their properties as leaky core conductors receiving distributed inputs, contributes to the computational function of neuronal circuits.

Acknowledgments

We would like to thank P Jedlicka and K Shapcott for helpful discussion. We acknowledge funding through BMBF grant 01GQ1406 (Bernstein Award 2013).

Supporting information

S1 File. Python 3.6 scripts to produce and train normalised artificial neural networks, as well as generate all figures in the paper.

S2 File. Keras and TensorFlow code for a normalised sparse layer with gradients given by Eq 7.

References

Abbott, LF, & Nelson, SB. 2000. Synaptic plasticity: Taming the beast. *Nat Neurosci*, **3**, 1178–1183.

Ardila, D, Kiraly, AP, Bharadwaj, S, Choi, B, Reicher, JJ, Peng, L, Tse, D, Etemadi, M, Ye, W, Corrado, DP, Naidich, SP, & Shetty, S. 2019. End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. *Nat Med*, **25**(6), 954–961.

Bassett, DS, & Bullmore, ET. 2017. Small-world brain networks revisited. *Neuroscientist*, **23**(5), 499–516.

Bird, AD, & Cuntz, H. 2016. Optimal current transfer in dendrites. *PLOS Comput Biol*, **12**(5), e1004897.

Chklovskii, D. 2004. Synaptic connectivity and neuronal morphology: Two sides of the same coin. *Neuron*, **43**(5), 609–617.

Cireşan, D, Meier, U, & Schmidhuber, J. 2012. Multi-column deep neural networks for image classification. *Proc IEEE Conf on Comput Vis Pat Rec* 3642-3649.

Cuntz, H, Bird, AD, Beining, M, Schneider, M, Mediavilla, L, Hoffmann, FZ, Deller, T, & Jedlicka, P. 2019. A general principle of dendritic constancy – A neuron's size and shape invariant excitability. *bioRxiv*, 787911.

Dale, H. 1935. Pharmacology and nerve-endings. *Proc R Soc Med*, **28**(3), 319–332.

Erdős, P, & Rényi, A. 1959. On random graphs. *Pub Math*, **6**, 290–297.

Gidon, A, & Segev, I. 2012. Principles governing the operation of synaptic inhibition in dendrites. *Neuron*, **75**(2), 330–341.

Gorur-Shandilya, S, Marder, E, & O'Leary, T. 2019. Homeostatic plasticity rules that compensate for cell size are susceptible to channel deletion. *bioRxiv*, 753608.

Gütig, R. 2016. Spiking neurons can discover predictive features by aggregate-label learning. *Science*, **351**(6277).

Gütig, R, & Sompolinsky, H. 2006. The tempotron: A neuron that learns spike timing–based decisions. *Nat Neurosci*, **9**(3), 420–428.
Nat Neurosci, **9**(3), 420–428.

Hebb, D. 1949. The organization of behavior: A neuropsychological theory. *Wiley*.

Hopfield, JJ, Feinstein, D, & Palmer, R. 1983. 'Unlearning' has a stabilizing effect in collective memories. *Nature*, **304**(5922), 158–159.

Ioffe, S, & Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML* 32:9.

Kim, CM, & Chow, CC. 2018. Learning recurrent dynamics in spiking networks. *eLife*, **7**, e37124.

Krizhevsky, A, Sutskever, I, & Hinton, G. 2012. ImageNet classification with deep convolutional neural networks. *NIPS* 25: 1097-1105.

Krotov, D, & Hopfield, JJ. 2019. Unsupervised learning by competing hidden units. *PNAS*, **116**(16), 67723–7731.

Lazar, A, Pipa, G, & Triesch, J. 2009. SORN: A self-organizing recurrent neural network. *Front Comput Neurosci*, **3**.

LeCun, Y, Boser, B, Denker, JS, Henderson, D, Howard, RE, Hubbard, W, & Jackel, LD. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Comput*, **1**(4): 541–551.

LeCun, Y, Denker, JS, & Solla, SA. 1990. Optimal brain damage. *NIPS* 2: 598-605.

LeCun, Y, Bottou, L, Bengio, Y, & Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proc IEEE* **86**(11): 2278–2324.

London, M, & Häusser, M. 2005. Dendritic computation. *Annu Rev Neurosci* **28**: 503–532.

Louizos, C, Welling, M, & Kingma, DP. 2017. Learning sparse neural networks through L^0 regularization. *arXiv:1712.01312*

Maass, W. 1997. Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, **10**(9), 1659–1671.

Mainen, ZF, & Sejnowski, T. 1996. Influence of dendritic structure on firing pattern in model neocortical neurons. *Nature*, **382**(6589), 363–366.

McCulloch, WS, & Pitts, W. 1943. A logical calculus of the ideas immanent in nervous activity. *Bull Math Biophys*, **5**(4), 115–133.

McDonnell, JR, & Waagen, D. 1993. Evolving neural network connectivity. *Proc IEEE Neur Net*, **2**, 863–868.

Mocanu, DC, Mocanu, E, Stone, P, Nguyen, PH, Gibescu, M, & Liotta, A. 2018. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nat Comms*, **9**(1).

Poirazi, P, Brannon, T, & Mel, B. 2003. Pyramidal neurons as two-layer neural networks. *Neuron*, **37**(6), 989–999.

Rall, W. 1957. Membrane time constant of motoneurons. *Science*, **126**(3271), 454–454.

Rall, W. 1962. Theory of physiological properties of dendrites. *Ann New York Acad Sci*, **96**, 1071–1092.

Richards, BA, Lillicrap, TP, Beaudoin, P, Bengio, Y, Bogacz, R, Christensen, A, Clopath, C, Ponte Costa, R, de Berker, A, Ganguli, S *et al*. 2019. A deep learning framework for neuroscience. *Nat Neurosci*, **22**, 1761–1770.

Robbins, H, & Monro, S. 1951. A stochastic approximation method. *Ann Math Statist*, **22**(3), 400–407.

Royer, S, & Paré, D. 2003. Conservation of total synaptic weight through balanced synaptic depression and potentiation. *Nature*, **422**, 518–522.

Rumelhart, DE, Hinton, GE, & Williams, RJ. 1986. Learning representations by back-propagating errors. *Nature*, **323**(6088), 533–536.

Rumsey, CC, & Abbott, LF. 2006. Synaptic democracy in active dendrites. *J Neurophysiol*, **96**(5), 2307–2318.

Salimans, T, & Kingma, DP. 2016. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *NIPS* 29: 901-909.

Silberberg, G, & Markram, H. 2007. Disynaptic inhibition between neocortical pyramidal cells mediated by Martinotti cells. *Neuron*, **53**(5), 735–746.

Stanley, KO, & Miikkulainen R. 2002. Evolving neural networks through augmenting topologies. *Evol Comput*, **10**, 99-127.

Sutskever, I, Vinyals, O, & Le, QV. 2014. Sequence to sequence learning with neural networks. *NIPS* 27: 3104-3112.

Tibshirani, R 1996. Regression shrinkage and selection via the lasso. *JRSS B*, **58**(1), 267–288.

Whiteson, S, & Stone, P. 2006. Evolving function approximation for reinforcement learning. *J Mach Learn Res*, **7**, 877–917.

Xiao, H, Rasul, K, & Vollgraf, R. 2017. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*.

Symbol	Interpretation
a_i	Activation of neuron i (Eqs 1 and 3)
b_i	Bias of neuron i (Eq 2)
C	Log-likelihood cost function (Eqs 4 and 7)
n_i	Number of afferent contacts to neuron i (also written $\|\mathbf{v}_i\|_0$)
s	Constant of proportionality between normalised and unnormalised inputs (Eq 6)
$\mathbf{v}_i$	Unnormalised input to neuron i (Eq 6)
$\mathbf{w}_i$	Normalised input to neuron i (Eq 6)
ε	SET connection probability
ζ	SET excision probability
η	Learning rate for stochastic gradient descent
σ	Sigmoid activation function (Eq 1)
σ_s	Softmax activation function (Eq 3)
τ	Threshold-linear activation function (Eq 5)

Table 1. Table summarising symbols and interpretations.