

# Note: Training stochastic neural networks

M. Rule

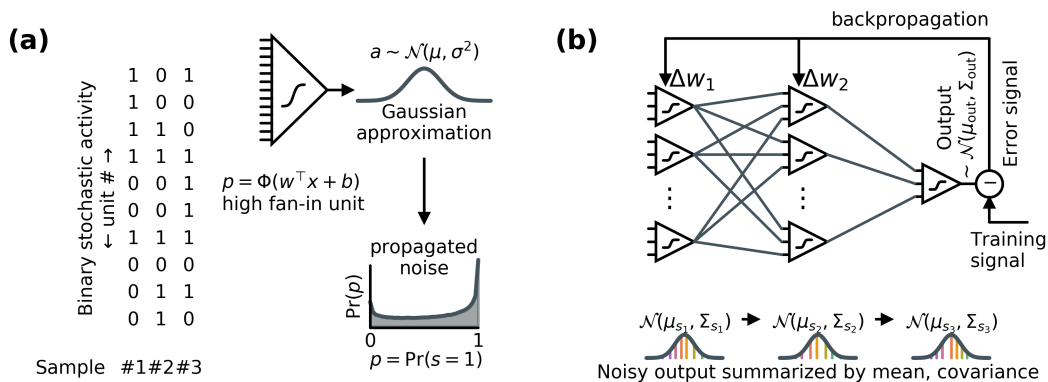
October 2, 2019

Feed-forward neural networks consist of a series of layers. In each layer, outputs from past layers are combined linearly, then passed through some nonlinear transformation. As long as all computations are differentiable, the entire network is differentiable as well. This allows artificial neural networks to be trained using gradient-based optimization techniques (backpropagation).

Methods for training stochastic networks via backpropagation are less well developed, but solutions exist and are the subject of ongoing research (c.f. Rezende et al. 2014 and the numerous papers that cite it). In the context of models of neural computation, Echeveste et al. (2019) trained stochastic neural networks with rectified-polynomial nonlinearities.

One advantage of stochastic neural networks is that they allow a non-differentiable binary spiking network to be treated as differentiable. Backpropagation with binary spiking networks ill-posed, because the derivative of the hard-threshold (Heaviside step function) diverges. Deterministic binary networks are typically trained using pseudogradient methods, in which the hard-threshold is replaced by a differentiable soft-threshold when propagating gradients backwards.

However, in stochastic networks, it is possible for the moments of network activity (e.g. mean  $\mu$  and covariance  $\Sigma$ ) to be differentiable, even if the underlying activations are not. This opens up another avenue for training binary networks using backpropagation.



**Figure 1: training stochastic neural networks** (a) Weighted sums over many inputs average together into a Gaussian variable (due to the central limit theorem), and samples from the noisy network can be described in terms of their mean and covariance. (b) Instead of mean firing rates, one can propagate both means and covariances through the layers in a neural network, using moment approximations. This provides a differentiable representation of stochastic binary networks that can be trained via backpropagation.

One can use moment approximations to obtain differentiable models of how noise propagates in a neural network. This is feasible if each neuron takes in a large number of inputs, so that the activation can be modeled as Gaussian, despite the binary nature of spiking (Fig. 1a). A good example of this is in Echeveste et al. (2019), which uses stochastic rate neurons with rectified polynomial nonlinearities.

For binary spiking networks, The dichotomized Gaussian (Macke et al 2011) provides convenient moment approximations, which can be optimized via backpropagation (Fig. 1b).

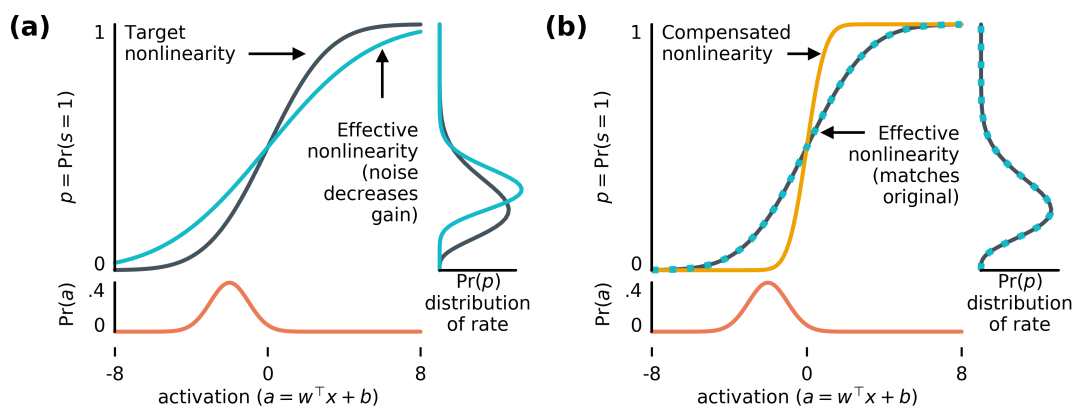
The moment-backpropagation explored here would be impossible in a biological neural network, since it requires the joint distribution of network activity. Generically, one might expect that if noise is present during learning, then neurons will also learn to suppress and ignore this noise. This predicts that networks trained in the presence of noise may also become dependent on this noise for their computational properties (Fig. 4def). In silico, this style of learning can be implemented by sampling an ensemble of frozen noise trajectories, and differentiating the evolution of each of these (now deterministic) trajectories with respect to the network parameters (Echeveste et al. 2019).

Learning in the presence of noise can allow neural networks to harness variability to perform useful computation. For example, Echeveste et al. (2019) show that networks can learn to represent uncertainty by sampling.

One caveat of this, however, is that networks trained to be robust to noise (or to harness it), may be sensitive to fluctuating noise levels. A neural network that requires access to noise for sampling may fail if that noise is removed, in much the same way that a deterministic network may fail when noise is added.

If noise is a constitutive component of neural computation, the brain must have mechanisms that stabilize noise levels, or allow computations to function robustly throughout any physiological fluctuations in noise levels.

Homeostasis in input-output statistics is observed in neurons, and is widely assumed to be necessary to stabilize neural dynamics (Marder and Prinz 2002, Zenke et al. 2013, O’Leary et al. 2013, O’Leary and Marder 2016, Zenke and Gerstner 2017, O’Leary 2018). One way to mitigate the effect of noise might be to use homeostasis to stabilize the input-output statistics of single neurons (Fig 2).



**Figure 2: Homeostatic approaches to noise-robustness (a)** Changes in noise statistics interact with nonlinearities, and affect the (average) transfer function of neurons. **(b)** If noise statistics are consistent over time, we can detect and correct for distortion in transfer functions in single neurons, using local homeostatic plasticity.