“‘

Neural Computation no. 1200

The formation of topographic maps which maximize the average mutual information of the output responses to noiseless input signals

Marc M. Van Hulle
K.U.Leuven, Laboratorium voor Neuro- en Psychofysiologie
Campus Gasthuisberg, Herestraat 49, B-3000 Leuven, BELGIUM
E-mail: marc@neuro.kuleuven.ac.be

April 29, 1996

Abstract

This note introduces an extremely simple and local learning rule for topographic map formation. The rule, called the Maximum Entropy learning Rule (MER), maximizes the unconditional entropy of the map's output for any type of input distribution. The aim of this note is to show that MER is a viable strategy for building topographic maps that maximize the average mutual information of the output responses to noiseless input signals when input noise and noise-added input signals are available only.

1 Introduction

Ralph Linsker was among the first to introduce information-theoretic principles into the field of self-organizing neural networks. He devised a global learning rule for topographic map formation by maximizing the average mutual information (Shannon information rate) between the output and the signal part of the input, which is corrupted by noise (Linsker, 1989). He argued that mutual information maximization is an optimal way to extract statistically salient input features. Others have devised learning rules from computational principles that are different but somehow related to the former: 1) principal component analysis (Földiák, 1989; Rubner and Schulten, 1990; Sanger, 1989) –and its non-linear extension which is modeled by Kohonen's SOM algorithm (Ritter *et al.*, 1992)– as a way to account for most of the input's variance but not for the noise, and 2) smoothing and predictive filtering (Atick and Redlich, 1991) for which the limiting case approximates mutual information maximization. Linsker also introduced a two-stage, local learning rule for maximizing the average mutual information when the input distribution is multivariate Gaussian (Linsker, 1992). The aim of this note is to introduce a simple learning strategy for building topographic maps which maximize the average mutual information of the map outputs, in response to noiseless inputs, by using input noise and noise-added input signals only.

2 Maximum Entropy Learning Rule

The aim of topographic map formation is to develop, in an unsupervised way, a mapping from a higher d-dimensional space V of input signals onto an equal or lower-dimensional discrete lattice A of N formal neurons. To each formal neuron $i \in A$ corresponds a unique weight vector $\mathbf{w}_i = [w_{i1}, ..., w_{id}]$ defined in V-space. As a result of this mapping, the formal neurons quantize the input space V, with probability density function ($p.d.f.$) $p(\mathbf{v})$, into a discrete number of partition cells or quantization regions.

The definition of quantization region is, in our case, different from the one used in a Voronoi (Dirichlet) tessellation. Assume that V and A have the same dimensionality d and that A comprises a periodic and regular d-dimensional lattice with a rectangular topology. Since the topology is rectangular, the lattice consists of a number of d-dimensional quadrilaterals (Fig. 1A). For example in Fig. 1B, the quadrilateral labeled H_e is defined by the neurons i, j, k, m: the weights of neurons i, j, k, m delimit a region in V-space. Hence, each quadrilateral represents a "granular" quantization region in V-space. We also consider the "imaginary" quadrilaterals facing the outer border of the lattice and which represent the "overload" quantization regions (Fig. 1B). To each of the Q quadrilaterals of A, we associate a code membership function:

$$\mathbb{1}_{H_j}(\mathbf{v}) = \begin{cases} \frac{2^d}{n_{H_j}} & \text{if } \mathbf{v} \in H_j \\ 0 & \text{if } \mathbf{v} \notin H_j, \end{cases} \tag{1}$$

with n_{H_j}, $1 \leq n_{H_j} \leq 2^d$, the number of vertices of H_j. We assume that $p(\mathbf{v})$ is stationary and ergodic and that the probability that $\mathbf{v}$ falls on one of the links equals zero: hence, a single input will activate two or more quadrilaterals only when these quadrilaterals overlap. (Note that several quadrilaterals may be active at the same time when the lattice is tangled.) Define S_i as the set of 2^d quadrilaterals that have neuron i as a common vertex ($e.g.$ for neuron j in Fig. 1B, S_j comprises H_h, H_i, H_e, H_f). Neuron i is said to be "active" if at least one of the quadrilaterals of the set S_i is active ($i.e.$ with non-zero code membership function output). The d-dimensional Maximum Entropy learning Rule (MER) is defined as:

$$\Delta \mathbf{w}_i = \eta \sum_{H_j \in S_i} \mathbb{1}_{H_j}(v)\ Sgn(\mathbf{v} - \mathbf{w}_i), \quad \forall i \in A, \tag{2}$$

with η the learning rate (a positive constant) and $Sgn(.)$ the sign function acting componentwise. The set S_i guarantees that only the vertices of the active quadrilaterals are updated. In Appendix 1, we show that the average of eq. (2) converges since it performs stochastic gradient descent on a positive definite cost function E ($i.e.$ a Liapunov function).

We know that the unconditional entropy is maximal when each of the N neurons is active with equal probability. In Appendix 1 we show that, for the one-dimensional case, MER is guaranteed to yield a unique and equiprobable activity distribution: $P(1) = ... = P(i) = ...P(N) = \frac{1}{N}$, with $P(i)$ the probability that neuron i is active; for the general d-dimensional case we show that the output (weight) density is proportional to the input density for large N ($i.e.$ an equiprobable quantization). Furthermore, also in Appendix 1, we show that a tangled lattice cannot be stable configuration in the d-dimensional (as well as in the one-dimensional) case.

3 Mutual Information Maximization

From the previous section we know that MER will yield a topographic map which maximizes the unconditional entropy of its N binary outputs, given the input distribution $p(\mathbf{v})$. The binary outputs result from quantizing the input signal $\mathbf{v}$ ($i.e.$ the $\mathbb{1}_{H_j}$'s). Define $F(\mathbf{y}, \mathbf{x})$ as the mutual information between

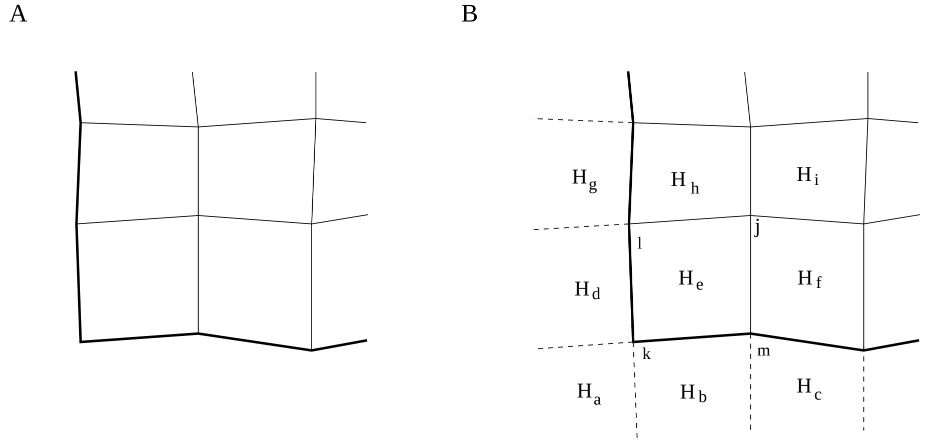

Figure 1: Definition of quantization region. (A) Portion of a two-dimensional lattice A with a rectangular topology, represented in V space. To every line crossing (vertex) corresponds a neuron weight vector. The bold line shows the outer border of the lattice. (B) The same portion of the lattice but with some of the neurons and quadrilaterals labeled. The quadrilaterals inside the lattice, labeled H_e, H_f, H_h, H_i, are the "granular" quantization regions; those that are labeled H_a, H_b, H_c, H_d, H_g are the "overload" regions and they are constructed by extending the links connecting the neuron on the border with its immediate neighbors (dashed lines).

the network's binary output $\mathbf{y}$ and the network's desired binary output $\mathbf{x}$ in response to noiseless input signals. The mutual information can be written as:

$$F(\mathbf{y}, \mathbf{x}) = I(\mathbf{y}) - I(\mathbf{y}|\mathbf{x}), \tag{3}$$

with $I(\mathbf{y})$ the unconditional entropy of $\mathbf{y}$ and $I(\mathbf{y}|\mathbf{x})$ the conditional entropy of $\mathbf{y}$, given $\mathbf{x}$. By maximizing F, we maximize the correspondence between the $\mathbf{y}$ and $\mathbf{x}$ vectors. Assume that we dispose of two types of input signals: the input signal $\mathbf{v}$ which is the sum of a clean input signal $\mathbf{s}$ and input noise $\mathbf{n}$ (additive noise), and the input noise signal $\mathbf{n}$. Define further p_s as the clean signal distribution, p_n as the noise distribution and p_{sn} as the distribution of $\mathbf{v}$, *i.e.* the input signal distribution in the presence of additive noise. Linsker (1992) suggested to maximize the mutual information between the signal portion of the noisy input (additive noise) and the output response to the noisy input, and introduced a learning strategy in which gradient ascent was performed on F directly, with a negative learning rate for $I(\mathbf{y}|\mathbf{x})$. Here we will propose a different, much simpler learning strategy for maximizing our definition of average mutual information.

We can rewrite eq. (3) as $F(\mathbf{y}, \mathbf{x}) = I(\mathbf{x}) - I(\mathbf{x}|\mathbf{y})$, with $I(\mathbf{x})$ the unconditional entropy of $\mathbf{x}$, and with $I(\mathbf{x}|\mathbf{y})$ the average uncertainty about $\mathbf{x}$ when we know $\mathbf{y}$ (equivocation). Hence, an $\mathbf{y}$ vector does not contribute to $I(\mathbf{x}|\mathbf{y})$ when its activation region coincides with that of an $\mathbf{x}$ vector, or when it is a subset thereof. Since $I(\mathbf{x}|\mathbf{y})$ is expected to decrease as the distribution of $\mathbf{y}$ vectors approaches the desired distribution of $\mathbf{x}$ vectors, the mutual information F will be maximal when $I(\mathbf{x}|\mathbf{y})$ is minimal. (Note that $max(F) = max(I(\mathbf{x})) = \log N$.) Hence, F is a function of the binary output vectors $\mathbf{y}$ only, and the problem reduces to one of determining the topographic map which maximizes the unconditional entropy of the distribution of output responses to p_s. This will be achieved in the following way.

We use MER to derive two maps: one which maximizes the unconditional entropy of the map's output corresponding to p_{sn} and another which maximizes it for p_n. The purpose of this entropy maximization is to derive non-parametric models of both input distributions: since entropy maximization leads to an equiprobable quantization and since, by the latter, the weight distribution closely approximates the input signal distribution for large N, MER can be used for reconstructing the shape of p_{sn} and p_n (*e.g.* using splines). (Note that, by the separation of density estimation from density reconstruction, such an approach is different from other adaptive non-parametric density estimation approaches, such as adaptive kernel- and nearest-neighbor density estimation, see Silverman, 1986.) Once p_{sn} and p_n are modeled in this way, we can infer the clean signal distribution: since the noise is additive, p_{sn} equals the convolution of p_s and p_n. Hence, in principle, we can obtain the desired clean signal distribution p_s by deconvolution. We can then re-apply MER to determine the weight vectors which maximize the uncondition entropy of the distribution of output responses to p_s and thus, which maximize F.

For the sake of exposition, we have used in the simulations a procedure which is even simpler than deconvolution. Assume that both input distributions are univariate Gaussians and that N is odd. Hence, at convergence, the averaged weights w_{mid}, with $mid = (N-1)/2 + 1$, represents also the means of the corresponding distributions. The weights for p_s are inferred from those for p_{sn} and p_n in the following way: $w_j^s = \sqrt{1-\rho^2} w_j^{sn}$ with $\rho = \frac{w_j^n - w_{mid}^n}{w_j^{sn} - w_{mid}^{sn}}$, $\forall j$, $j \neq mid$. Evidently, this leads to the correct solution when p_{sn} and p_n are Gaussians; for other unimodal, symmetric distributions, with small higher-than-second-order moments, it is an approximation.

Finally, we note that the estimation of p_{sn} and p_n is a common procedure in model-based noise reduction schemes: *e.g.* in speech applications, one obtains estimates of (environmental) noise when no speech signal is present. Knowledge of the underlying input distributions greatly improves the quality of *e.g.* blind signal identification (Pajunen *et al.*, 1996), speech enhancement (Xie and Van Compernolle, 1994), speech recognition (Rabiner and Juang, 1993) and also for hybrid systems (based on mutual information maximization, see Rigoll, 1994), adaptive waveform coding (Martinez and Yang, 1995), and adaptive channel equalization (Adali *et al.*, 1995).

4 Simulations

We run two networks, one for p_{sn} and another for p_n, until convergence and determine the weights of p_s as explained in the previous section. We then compute numerically the neural activations for p_s given these weights. Finally, we display the performance of our MER-based procedure. Since entropy is not a very sensitive metric, due to its logarithmic nature, we plot the mean squared error (MSE) between the present neural activations for p_s and the desired, equiprobable ones. For MER, we use the fast version described in Appendix 2; the learning rate $\eta = 0.0001$.

We will give two examples. Firstly, we consider the standard case where the samples from p_s and p_n are randomly and independently drawn from zero-mean Gaussians with standard deviations σ_s and σ_n, respectively. The result is shown in Fig. 2A for $\sigma_s = 0.5$ and different values for σ_n and for $N = 17$ and 33. We observe that our MER-based procedure yields a rather accurate estimate of p_s up to $\sigma_n = 0.5$ (signal-to-noise ratio (SNR) of 0 dB).

Secondly, in order to show that our MER-based procedure also works for non-Gaussian distributions, we consider the case where p_s is a zero-mean symmetric product distribution and p_n a zero-mean uniform distribution. The symmetric product distribution is a strongly-peaked *p.d.f.* that is generated by taking the product of two random numbers that are uniformly-distributed within the ranges $[0, 1/\sqrt{2})$

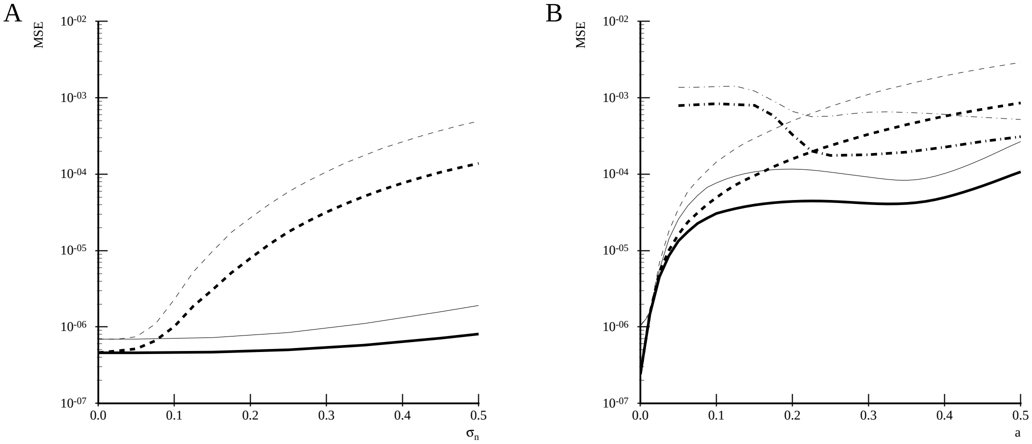

Figure 2: Mutual information maximization performance of the MER-based procedure. The performance is expressed as the MSE between the neural activations for the clean signal distribution and the desired, equiprobable ones. The thin and thick full lines show the MSE result for $N = 17$ and 33, respectively. The thin and thick dashed lines show the corresponding MSE results in case only p_{sn} is maximized. This serves as a reference against which the performance of our MER-based procedure is to be compared. (A) Result in case of zero-mean Gaussians for the clean- and the noise signal with standard deviations σ_s and σ_n, respectively; $\sigma_s = 0.5$. (B) Result in case of the symmetric product distribution with range $(-1/2, 1/2)$ for the clean signal and the uniform distribution with range $[-a, a)$ for the noise signal; $\sigma_s \cong 0.166$ and $\sigma_n = \frac{a}{\sqrt{3}}$. The $4th$ moment of the clean signal $\cong 2.5\ 10^{-3}$. The thin and thick dot-dash lines denote the results obtained in case of deconvolution for $N = 17$ and 33, respectively.

or $(-1/\sqrt{2}, 0]$; both ranges are equally probable. The range of the product distribution is $(-1/2, 1/2)$ and that of the uniform distribution is $[-a, a)$. The product distribution falls off log-hyperbolically from the center. The result is shown in Fig. 2B. We observe that our MER-based procedure yields a significant improvement at least up to $a = 0.5$ (SNR = -4.8 dB). In order to verify the effectiveness of our "Gaussian" approximation, we have applied a deconvolution procedure in which MER is used for estimating p_{sn} and p_n (the deconvolution procedure is detailed in Appendix 3). However, we observe that deconvolution yields inferior results, at least for the range of a shown (dot-dash lines). The inferior performance is due to the presence of the sharp peak in p_s which makes the latter difficult to reconstruct. We also observe that the procedure breaks down below $a = 0.2$.

Finally, we note that the $I(\mathbf{y})$ values obtained in our simulations were typically within 2 % of the theoretically-attainable range, even though the weights could be significantly different from the optimal ones. Hence, we feel that performing gradient ascent on $I(\mathbf{y})$ directly is not recommended by its logarithmic nature, especially when the (decrease in the) progress made in $I(\mathbf{y})$ is used in the stopping criterion.

Acknowledgements

The author wishes to thank the anonymous reviewers for their valuable comments. M.M.V.H. is a research associate of the National Fund for Scientific Research (Belgium) and is supported by research grants received from the National Fund (G.0185.96) and the European Commission (ECVnet EP8212).

References

Adali, T., Liu, X., Li, N., and Sönmez, M.K. (1995). A maximum partial likelihood framework for channel equalization by distribution learning. *Proc. IEEE NNSP95* (Cambridge, MA, 1995), pp. 541-550.

Atick, J.J., and Redlich, A.N. (1991). Predicting ganglion and simple cell receptive field organizations. *Int'l J. Neural Syst.*, **1**, 305-315.

Földiák, P. (1989). Adaptive network for optimal linear feature extraction. In *Int'l Joint Conference on Neural Networks* (Washington 1989), vol. I, pp. 401-405.

Linsker, R. (1989). How to generate ordered maps by maximizing the mutual information between input and output signals. *Neural Computation*, **1**, 402-411.

Linsker, R. (1992). Local synaptic learning rules suffice to maximize mutual information in a linear network. *Neural Computation*, **4**, 691-702.

Martinez, D., and Yang, W. (1995). A robust backward adaptive quantizer. *Proc. IEEE NNSP95* (Cambridge, MA, 1995), pp. 531-540.

Oppenheim, A.V., and Schafer, R.W. (1975). *Digital Signal Processing.* Prentice-Hall: Englewood Cliffs, New Jersey.

Pajunen, P., Hyvärinen, A., and Karhunen, J. (1996). Nonlinear Blind Source Separation by Self-Organizing Maps. Submitted to ICONIP'96.

Rabiner, L., and Juang, B.H. (1993). *Fundamentals of Speech Recognition.* Englewood Cliffs, New Jersey: Prentice Hall.

Rigoll, G. (1994). Maximum Mutual Information Neural Networks for Hybrid Connectionist-HMM Speech Recognition Systems. *IEEE Trans. Speech and Audio Processing*, **2**, 175-184.

Ritter, H., Martinetz, T., & Schulten, K. (1992). *Neural computation and self-organizing maps: An introduction.* Reading, Mass: Addison-Wesley.

Rubner, J., and Schulten, K. (1990). Development of feature detectors for self-organization. *Biol. Cybern.*, **62**, 193-199.

Sanger, T. (1989). An optimality principle for unsupervised learning. In *Advances in Neural Information Processing Systems 1*, D.S. Touretzky, ed., pp. 11-19. Morgan Kaufmann, San Mateo, CA.

Silverman, B.W. (1986). *Density Estimation for Statistics and Data Analysis.* Chapman and Hall: London.

Xie, F., and Van Compernolle, D. (1994). A family of MLP based nonlinear spectral estimators for noise reduction. *Proc. ICASSP'94*, vol. II, pp. 53-56.

Appendix 1: Properties of MER

Proposition 1: For a discrete and statistically-stationary input probability density $p(\mathbf{v})$, eq. (2) on average (for small enough η) performs (stochastic) gradient descent on the cost function:

$$E = \sum_{k=1}^{M} \sum_{i=1}^{N} \sum_{H_j \in S_i} \mathbb{1}_{H_j}(\mathbf{v}^k) \mid \mathbf{v}^k - \mathbf{w}_i \mid, \tag{4}$$

with $\{\mathbf{v}^k\}$ the set of input signals, and $\mid \mathbf{v}^k - \mathbf{w}_i \mid = \sum_{j=1}^{d} \mid v_j^k - w_{ij} \mid$ (*i.e.* the absolute "per-letter" distortion).

Proof: Following the definition of gradient descent, the weights are updated so as to reduce the cost term introduced by the active quantization regions (defined by the weight vectors of the previous iteration step). Hence, since eq. (4) is continuous and piecewise differentiable, we obtain that:

$$< \Delta \mathbf{w}_i >_V = -\eta \frac{\partial E}{\partial \mathbf{w}_i} = \eta \sum_{k=1}^{M} \sum_{H_j \in S_i} \mathbb{1}_{H_j}(\mathbf{v}^k) \; Sgn(\mathbf{v}^k - \mathbf{w}_i), \quad \forall i \in A, \tag{5}$$

and the latter exactly corresponds to the right hand side of eq. (2), summed over all input patterns. QED.

Now since E is always positive definite and its partial derivatives eq. (5) are continuous, eq. (4) is a Liapunov function.

For the case where $p(\mathbf{v})$ is continuous, we modify the *parallel* update scheme of MER to a *sequential* one: 1) one quadrilateral out of the set of active quadrilaterals is chosen randomly, and 2) one vertex out of the vertices of the randomly chosen quadrilateral is updated using MER.

Proposition 2: For a continuous and statistically-stationary input *p.d.f.* $p(\mathbf{v})$, and a sequential update scheme, a Liapunov function exists for averaged MER.

Proof: The derivatives of the average learning rule comprise a matrix $HE = [HE_{ij}]$, $HE_{ij} = \frac{\partial < \Delta \mathbf{w}_i >_V}{\partial \mathbf{w}_j}$, with components: $HE_{ij} = 0$, $i \neq j$, and $HE_{ii} \neq 0$. Since the matrix HE is symmetric, it is a Hessian. Hence, if and only if HE is a Hessian, a Liapunov function exists. QED.

Proposition 3: In the one-dimensional case, MER is guaranteed to yield a set of equiprobable neurons.

Proof: Consider a one-dimensional lattice with quantization intervals $H_1, ..., H_{N+1}$ defined by the weights $w_1, ..., w_N$. Due to the existence of a Liapunov function, we know that MER will converge on average. We have that:

$$< \Delta w_i >_V = 0 = < \mathbb{1}_{H_{i+1}} - \mathbb{1}_{H_i} >_V = \frac{2 \, P(H_{i+1})}{n_{H_{i+1}}} - \frac{2 \, P(H_i)}{n_{H_i}}, \quad i = 1, ..., N, \tag{6}$$

at convergence, with $P(H_i)$ the probability that H_i is active (*i.e.* $P(\mathbb{1}_{H_i} \neq 0)$). Now since, by definition, neuron i is activated when either H_i or H_{i+1} or both are activated, the probability that neuron i is activated becomes: $P(i) = \frac{P(H_i)}{n_{H_i}} + \frac{P(H_{i+1})}{n_{H_{i+1}}}$ (where we have normalized to keep $\sum_i P(i) = 1$). If we now substitute for the $P(H_i)$-equalities obtained from eq. (6), we obtain that: $P(1) = ... = P(i) = ...P(N)$, and thus a set of equiprobable neurons because $n_{H_i} = 2$, $1 < i < N+1$ and $n_{H_1} = n_{H_{N+1}} = 1$. QED.

Corollary 1: In the one-dimensional case, the averaged neuron weights $w_1, ..., w_N$ at convergence are the medians of the intervals they delimit.

Proof: Follows directly from the equiprobable quantization. QED.

Define a kink at neuron i as the topological defect for which either $w_i > w_{i+1}$ and $w_i > w_{i-1}$, or $w_i < w_{i+1}$ and $w_i < w_{i-1}$.

Proposition 4: In the one-dimensional case, MER is guaranteed to converge on average to a one-dimensional lattice without kinks.

Proof by indirect demonstration: Assume the inverse: a one-dimensional lattice with at least one kink is a stable solution. Assume that there is a kink at neuron i hence, the intervals H_i and H_{i+1} overlap and are located at the same side of i. This signifies that neuron i will only receive weight updates in the direction of both overlapping intervals, and never in the opposite direction. By consequence, neuron i will keep on shifting until the overlap is removed. Hence, the kink at neuron i is not a stable configuration. QED.

Since kinks are the only topological defects of one-dimensional lattices, the latter two propositions also signify that MER converges to a *unique* set of equiprobable neurons: $P(1) = ... = P(i) = ...P(N) = \frac{1}{N}$.

Before we turn to the d-dimensional case, we need an additional definition. Since MER performs weight updates componentwise, we can divide the region within which neuron i can be activated into 2^d *quadrants*: for every input falling in the same quadrant, the magnitude and direction of the weight update vector is the same (*cf.* the sign functions in MER).

For the higher-than-one-dimensional case, one can easily verify that, for each neuron i, the (averaged) weight vector $\mathbf{w}_i$ at convergence represents the d-dimensional "median" of S_i on condition that the "median" is defined as the vector of the (scalar) medians in each of the d input dimensions separately. (Note that there exists no unique definition of median in d dimensions.) The weights satisfying this condition are stable equilibrium points of our averaged learning rule. Furthermore, when all opposite quadrants of a neuron carry non-zero activation probabilities, these quadrants will be equiprobable.

Proposition 5: In the d-dimensional case, the output (weight) density $\lambda(\mathbf{w}_i)$ at convergence is proportional to the input density $p(\mathbf{v})$, when N grows large and given that the neurons' activation regions S_i span non-zero volumes in V-space.

Proof: In the asymptotic situation where N is large, the discrete weight distribution can be expected to approximate a continuous density function λ having unit volume. Then $\lambda \ \Delta Vol$ may be taken as the fraction of weight vectors (or neurons) located in an incremental volume element ΔVol. Thus the volume of the quantization region S_i associated with neuron i is given approximately by:

$$Vol(S_i) \approx \frac{1}{N \ \lambda(\mathbf{w}_i)} \tag{7}$$

for every bounded region S_i ($\lambda \neq 0$). (Note that the denominator is the number of points in the neighborhood of $\mathbf{w}_i$ so that the reciprocal is the volume per neuron.) For large N it is reasonable to assume that most of the regions S_i will be bounded sets and the contribution from the "overload" regions will be negligible. Furthermore, also for large N, we can make the approximation that $p(\mathbf{v}) \sim p(\mathbf{w}_i)$, $\forall \mathbf{v} \in S_i$ (given $Vol(S_i) \neq 0$). We then obtain for the probability for neuron i to be active:

$$P(i) \doteq Vol(S_i) \ p(\mathbf{w}_i) \approx \frac{p(\mathbf{w}_i)}{N\lambda(\mathbf{w}_i)} \tag{8}$$

after substitution of eq. (7).

We can also rewrite the contribution made by neuron i to the cost function associated with MER as follows:

$$E_i = P(i) \int_{S_i} \mid \mathbf{v} - \mathbf{w}_i \mid d\mathbf{v} = \frac{p(\mathbf{w}_i)}{N\lambda(\mathbf{w}_i)} \int_{S_i} \mid \mathbf{v} - \mathbf{w}_i \mid d\mathbf{v} \tag{9}$$

(or with a summation over S_i instead of an integral in case of a discrete input distribution).

We know that E is a Liapunov function hence, at convergence, $\frac{\partial E_i}{\partial \mathbf{w}_i} = 0$, and $\mathbf{w}_i$ will be the "median" of S_i (our definition). We have also assumed that the volume $Vol(S_i) \neq 0$, hence, the integral term in eq. (9) will always be positive. Taking the derivative of eq. (9) yields:

$$\frac{\partial E_i}{\partial \mathbf{w}_i} = 0 = \int_{S_i} \mid \mathbf{v} - \mathbf{w}_i \mid d\mathbf{v} \ \frac{\partial}{\partial \mathbf{w}_i} \frac{p(\mathbf{w}_i)}{N\lambda(\mathbf{w}_i)} + \frac{p(\mathbf{w}_i)}{N\lambda(\mathbf{w}_i)} \ \frac{\partial}{\partial \mathbf{w}_i} \int_{S_i} \mid \mathbf{v} - \mathbf{w}_i \mid d\mathbf{v}. \tag{10}$$

We know that the derivative of the integral equals zero, since $\mathbf{w}_i$ is the "median" of S_i, and that the integral is positive and non-zero. Hence, the derivative of the ratio $\frac{p(\mathbf{w}_i)}{\lambda(\mathbf{w}_i)}$ must be equal to zero: in other words, the ratio is a constant. In conclusion, we have that $\lambda(\mathbf{w}_i) \propto p(\mathbf{v})$. QED.

Furthermore, since each quadrilateral is common to 2^d adjacent neurons, λ will be a smooth function.

Corollary 2: In the d-dimensional case, we have an equiprobable quantization for large N, given λ a smooth function.

Proof: Since, for large N and λ smooth, the ratio $\frac{p}{\lambda}$ reduces to a constant independent of the lattice index i, we have that each S_i contributes equally in terms of activation. Hence, we have an equiprobable quantization. QED.

Proposition 6: In the d-dimensional case, a tangled lattice is not a stable solution.

Proof by indirect demonstration: Assume the inverse: a lattice with at least one topological defect is a stable solution. Without loss of generality, we can consider a two-dimensional lattice. In case of a topological defect, *e.g.* at neuron i, at least 2 quadrilaterals will overlap. Assume that (part of) the overlap is located in the upper-left quadrant Q_{UL} and that an input falls in it: hence, quadrant Q_{UL} is activated *once*. However, due to the two-fold overlap, neuron i will be updated *twice*. (Note that a single weight update is the same for each quadrant.) As a result, the update probabilities do not match the activation probabilities of the assumed equiprobable quadrants and the average learning rule cannot be stable. Hence, the locally-tangled lattice cannot be a stable configuration. QED.

Appendix 2: Fast version of 1-dimensional MER

In the one-dimensional case, MER becomes:

$$\Delta w_i = \eta(\mathbb{1}_{H_{i+1}} - \mathbb{1}_{H_i}), \quad i = 1, ..., N. \tag{11}$$

The average squared change of weights per iteration (intensity of variation) at convergence is $\frac{2\eta^2}{N+1}$, for $i = 2, N-1$, and $\frac{3\eta^2}{N+1}$ for $i = 1$ or N. The rate of convergence for a *p.d.f.* with bounded support and for which the weights are initialized outside the *p.d.f.*'s range, is on the order of $\mathcal{O}(\frac{\eta}{N^2})$.

A faster version of MER is found by updating all weights at each time step:

$$\Delta w_i = \eta\left(\sum_{k=i+1}^{N+1} \frac{\mathbb{1}_{H_k}}{N+1-i} - \sum_{k=1}^{i} \frac{\mathbb{1}_{H_k}}{i}\right), \quad i = 1, ..., N. \tag{12}$$

By considering the homogeneous system of averaged eqs. (12) at convergence, it can be easily shown that the faster version of MER also yields a set of equiprobable neurons. The intensity of variation is now $\frac{\eta^2}{(N+1-i)i}$, for $i = 2, N-1$, and $\frac{\eta^2(2N+1)}{(N+1)N}$ for $i = 1$ or N. The rate of convergence is on the order of $\mathcal{O}(\frac{\eta}{N})$. Hence, for the same N, the faster version of MER converges N times faster and generates an N times smaller intensity of variation at convergence. Hence, we have used this rule in the simulations.

Appendix 3: Deconvolution

The samples taken from p_n and p_{sn} are quantized using MER. At the midpoints of the resulting intervals, the density estimate is located and the shape of the underlying density curve is approximated using cubic splines. This approximation is then resampled into 8192 equidistant points in the range $[-4, 4]$. Deconvolution is performed via the complex cepstrum (Oppenheim and Schafer, 1975).

The process of deconvolution is quite sensitive to the accuracy with which the shapes of p_n and p_{sn} are obtained. Therefore, we additionally use optimal filtering: we apply a band-pass filter to p_{sn} of which the (integer) low and high cut-off frequencies are chosen in such a way that the convolution of the estimated shape of p_s with the shape of p_n is as closely as possible to the shape of p_{sn} (in MSE sense). The adaptation of the cut-off frequencies is non-recursive: it simply yields a table from which the best combination is chosen.

Once the best estimated shape of p_s is obtained, the equiprobable intervals are determined numerically by integration (extended trapezoidal rule). The correctness of these intervals is then verified using the true shape of p_s: the difference from an equiprobable quantization is expressed in terms of a MSE value and plotted in Fig. 2B.