Probability Interpretation of Population Codes

Terence D. Sanger
Massachusetts Institute of Technology
Room E25-526
Cambridge, MA 02139
(617)253-5769 tds@ai.mit.edu

Abstract

Although theoretical neural network models frequently assume that the firing rates of individual neurons carry the analog signals used for computations, it is becoming clear that in many sensory and motor systems, information is encoded as patterns of activity over large populations of cells (Georgopoulos *et al.* 1988, for example). I seek to address several different but related questions concerning such populations. I investigate the computational properties of certain types of population codes by using a probabilistic interpretation of their structure. I show the existence of simple algorithms for nonlinear supervised learning as well as for unsupervised learning based on information maximization (Linsker 1988, Linsker 1989). And I show that although in general it is not possible to find a unique external variable that is coded by a population (Sanger 1994), it is possible to use information maximization principles to find the external variable that is represented *best* by the population.

1 INTRODUCTION

I define a "population code" or "distributed representation" to be a set of units for which the pattern of activity indicates the value of an underlying coded variable. This is in contrast to an "analog representation" in which the underlying variable is represented directly by the firing rates of one or more cells. The difference is illustrated in figure 1 (see (van Gelder 1991) for discussion). Perhaps the most common

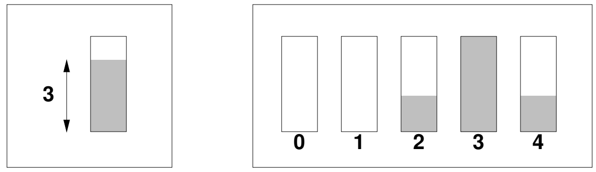

Figure 1: a. "Analog", and b. "Population" coding of the number 3.

computational example of a distributed representation is the binary number system, in which a pattern of 1's and 0's represents an approximation to an analog value. In this paper I will consider a particular form of distributed representation in which each unit is active for only a small contiguous range of input values. A radial basis function representation with local basis functions has this form, as does the unary (but not the binary) number system. I will refer to such representations as having "local receptive fields".

Population codes have been proposed as a model for information representation in certain regions of primate cortex (Georgopoulos *et al.* 1988, Andersen and Zipser 1988). This paper seeks to provide the beginnings of a theoretical foundation for understanding such representations and their computational properties. In particular, I attempt to answer the following three questions:

1. How can we compute functions of population coded variables, and are there supervised learning algorithms that can approximate desired functions?
2. What is an appropriate training criterion for unsupervised learning and do the necessary algorithms exist?
3. How can we interpret measurements made on biological population codes with reference to externally measurable variables?

I address these issues by first re-interpreting population codes with local receptive fields as representing estimates of the probability density of the input distribution. Using this interpretation, it becomes possible to apply the tools of probability theory to the analysis of population codes, and these tools yield simple yet powerful results that address the questions above.

2 PROBABILITY ANALOGY

Consider a scalar variable x that represents some measurable external quantity. Suppose that sensory processing encodes the variable x over a population of cells $\sigma_i(x)$ such that each cell has a localized "receptive field" for values of x. For example, we could use a Gaussian shaped receptive field such as

$$\sigma_i(x) = \frac{1}{s\sqrt{2\pi}} e^{-(x-m_i)^2/s^2}$$

where the standard deviation s determines the specificity and overlap of the receptive fields, and the means m_i determine the centers. Then any value of x is represented as a local pattern of activity over the set σ_i.

Now, define $\hat{x} = x + \eta$ to be a noisy measured value of x with additive independent random noise η. Let the probability distribution of η be equal to the receptive field shape σ_i. For example, if $\sigma_i(x)$ is as above, then let η be Normal with variance s^2. Then we have

$$\sigma_i(\hat{x}) = P[\hat{x}|x = m_i]$$

In other words, each basis function output can be interpreted as the probability that the noisy measurement $\hat{x}$ was in fact generated by the uncorrupted value $x = m_i$. The value of m_i for which $\sigma_i(\hat{x})$ is maximized is the maximum likelihood (ML) estimate of x given $\hat{x}$.

We can compute the maximum *a posteriori* (MAP) estimate of x given $\sigma(\hat{x})$ using Bayes' rule:

$$P[x = m_i|\hat{x}] = \frac{\sigma_i(\hat{x})P[x = m_i]}{P[\hat{x}]}$$

Note that this is proportional to $\sigma_i(\hat{x})$ if $P[x]$ and $P[\hat{x}]$ are uniform. In this case, the set of σ_i's gives the probability density of x for each measurement $\hat{x}$.

We can average the basis function outputs over multiple values of x to obtain the expected value:

$$E[\sigma_i(x)] = \int \sigma_i(x)P[x]dx \tag{1}$$

and this expression gives a smoothed and subsampled version of $P[x]$. In particular, if the receptive fields are all the same shape so that $\sigma_i(x) = \sigma(x - m_i)\forall i$, then equation 1 describes sampled values from the convolution $\sigma * P$ and approximates $P[x = m_i]$. Thus the expected values $E[\sigma_i(x)]$ give a sampled and smoothed approximation to the input probability density $P[x]$ in a manner similar to Parzen Windows density estimation. ($E[\sigma_i]$ can also be interpreted as a sampled but not smoothed version of $P[\hat{x}]$.)

The above analogy provides the theoretical basis for the analysis of this type of population code. Note that it applies only to population codes described by local receptive fields. The noise term η is an artificial theoretical construction that allows the analogy to be made. A receptive field in the population code corresponds to the noise distribution in the probability analogy, and a wider receptive field is equivalent to a higher noise variance.

3 SUPERVISED LEARNING

Consider a nonlinear smooth function between two variables y and x given by $y = f(x)$. If x is a random variable with probability density $P_x[x]$, then this induces a probability density $P_y[y]$ and we seek the functional that maps P_x into P_y. This immediately yields

The Density Mapping Theorem: *For any smooth function $y = f(x)$ there exists a piecewise linear transformation between probability representations for x and y given by*

$$P_y[y] = \int K(x,y) P_x[x] dx \tag{2}$$

where $K(x,y) = \delta(y - f(x)) f'(x)^{-1}$ for each monotonic segment of f.

This remarkably simple result shows that any nonlinear function between two analog variables has a representation as a linear functional between their probability densities. In terms of population codes, this provides an important computational property since only linear operations will be needed.

Note that the theorem only applies to the original continuous densities P_x and P_y, not to the smoothed and sampled versions given by population codes $\sigma_i(x)$ and $\phi_j(y)$. However, it can be shown that the best approximation to ϕ_j given σ_i is linear. The optimal linear transformation can be described by $\phi(y) \approx A\sigma(x)$ where the matrix A is given by

$$A = \int \int \phi(y) K(y,x) \tau(x)^T dy dx, \tag{3}$$

$K(y,x)$ is as above, $\phi(y)$ and $\sigma(x)$ are column vectors, and $\tau(x)$ is the vector function that minimizes the Hilbert space norm $||\delta(x' - x) - \tau(x')^T \sigma(x)||$.

Given population codes $\phi_j(y)$ and $\sigma_i(x)$ we can easily learn the linear mapping between them using either direct computation (via matrix pseudo-inversion) or iterative learning algorithms. Suitable algorithms include the Widrow-Hoff delta rule (Widrow and Hoff 1960) or recursive least squares estimation (Ljung and Soderstrom 1983).

4 UNSUPERVISED LEARNING

Suppose that the function $f(x)$ is not specified and we wish to find the variable $y = f(x)$ such that the mutual information between y (with additive noise) and x (which is bounded) is maximized, according to the "infomax" principle first suggested by Linsker (1988). In order to use the probability analogy, define the noise variable ν to have probability density equal to the shape of the receptive fields $\phi_j(y)$. Then we can interpret ϕ_j as the probability density of $\hat{y} = f(x) + \nu$ given x. $E[\phi_j]$ can be interpreted as sampled values of the probability density $P[\hat{y}]$.

The mutual information between $\hat{y}$ and x is given by

$$I(\hat{Y}; X) = H(\hat{Y}) - H(\hat{Y}|X)$$

where H is the entropy of the density of its argument. Since $H(\hat{Y}|X) = H(\nu)$ which is a constant determined by the shape of the receptive fields ϕ_j, maximization of

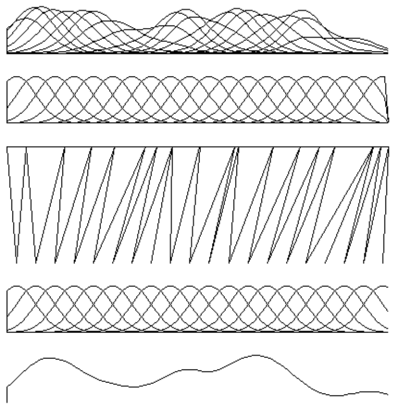

Figure 2: Synaptic diffusion algorithm structure and results. The mapping input density is chosen randomly, and the learned mapping is computed iteratively.

mutual information is achieved by maximizing $H(\hat{Y})$. This maximum occurs when $P[\hat{y}]$ is a constant. In this case, we also have that $E[\phi_j(y)]$ is equal for all j.

This gives us a principle for choosing an optimal linear transformation kernel K between densities, and thereby an optimal nonlinear transformation f. Effectively, the solution performs "histogram equalization" of the input density (see (Nadal and Parga 1994) for a detailed theoretical analysis). In the context of population codes, this is equivalent to adjusting cortical maps and cellular resources in such a way that all cells are used at the same average frequency.

Note that maintaining constant output expectation does not uniquely define the function f. However, if we add the requirement that the output units continue to have localized receptive fields we can obtain a unique solution. In particular, in the continuous case the kernel K needs to have the form shown in equation 2. In the discrete (sampled) case we use equation 3.

We can use a form of "synaptic diffusion" to approximate the optimal solution through an iterative algorithm. Consider the structure shown in figure 2. The "sensory receptive fields" are the functions $\sigma_i(x)$. These outputs are remapped onto a large vector $\psi = B\sigma$ via a stochastic matrix B whose column sums are all 1 and for which each row contains exactly one nonzero value. B is chosen so that the expected values $E[\psi_k]$ are all equal. This means that the connections from σ to ψ spread out the activity of each σ_i over a region of the vector ψ proportional to the expected value $E[\sigma_i]$. A subsequent step of calculating the ϕ_j's as a smoothed and subsampled version of ψ yields the new population code.

The receptive field shapes for a randomly chosen input density are shown at the top of figure 2. Note that in terms of the original input coordinates the output receptive fields are wider in regions of low data density and narrower in regions of high data density. This is appropriate, since the narrower fields will convey more information about the input data and thus should be allocated to more likely regions. A similar result is shown in more detail in (Linsker 1989).

The algorithm for finding B simply involves allowing the synapses from σ to ψ to "diffuse" down a gradient determined by the product of the local synaptic density B_{ji} and the expected value $E[\sigma_i]$. The one-dimensional example shown required 1.2 million diffusion steps (approximately 15 seconds on a 100Mhz 80486). Many other algorithms to solve this problem are possible and might perform faster.

The extension of this type of algorithm to high-dimensional input is theoretically straightforward but computationally difficult. Suppose that the input variable x is a 2-dimensional vector. Then the optimal y will also be 2-dimensional, with $P[y]$ a constant. If y is bounded in a square region, then $P[y]$ is a separable function that can be written as $P_1[y_1]P_2[y_2]$ and thus the two components y_1 and y_2 of y are independent. To achieve this, let $y_1 = f_1(x_1)$ and choose a monotonic f_1 such that $P[y_1]$ is constant. Now for each value of x_1 compute $P[x_2|x_1]$ and choose a function f_2 such that $y_2 = f_2(x_2, x_1)$ has $P[y_2|x_1] = P[y_2|y_1]$ constant. Then $P[y_2, y_1] = P[y_2|y_1]P[y_1]$ which is a constant as desired. The computational difficulty lies in the computation of $f_n(x_n, \ldots, x_1)$ when the number of dimensions n becomes large. Future work will investigate efficient approximations for this procedure.

5 CORTICAL REPRESENTATIONS

In the investigation of biological population codes, investigators frequently seek to identify the "coordinate system" in which the underlying variable x is coded. Recent work (Sanger 1994) has shown that it is not possible to define a unique coordinate system for a population code. In the current framework, this result can be summarized by noting that the density mapping theorem shows that different population codes can be mapped into each other using linear transformations that perform shift operations, so that it is relatively easy to change coordinate systems merely by re-labelling the values assigned to cell outputs.

The probability technique proposed here allows us to go beyond the lack of a unique coordinate system to investigate whether there is a *best* coordinate system about which the population gives maximal information. I propose the following

Definition: *An "optimal" coordinate system has the property that a uniform distribution of input samples with respect to this coordinate system yields a maximal entropy density over the distributed representation.*

For example, suppose that a set of cells have local receptive fields that are uniformly distributed in polar coordinates with even spacing along the r and θ dimensions. If sample inputs are drawn from a uniform distribution in Cartesian (x, y) coordinates, then many of the cells with receptive fields near the origin will be under-stimulated, and the resulting output density will not have maximum entropy. But if the input samples are selected uniformly in (r, θ) then the population code will have maximum entropy.

To find the optimal coordinate system, choose inputs uniformly with respect to some coordinate system x. Measure the cell outputs $\sigma_i(x)$, and compute a transformation kernel $K(y, x)$ such that $E[\phi_j(y)]$ is constant. Let f be the function represented by $K(y, x)$. Then the optimal coordinate system is given by $y = f(x)$.

Note that optimality is determined by the distribution of cell receptive field centers. If this distribution is uniform with respect to a candidate coordinate system, then that coordinate system is optimal. For instance, the distribution of preferred hand movement directions for cells in motor cortex is approximately uniform in Cartesian coordinates (Georgopoulos 1986), but the distribution of preferred hand force directions is not (Kalaska *et al.* 1989).

An interesting consequence of these results is that we can test whether or not an animal has in fact optimized a distributed representation. If the representation is optimal, then the actual distribution of sensory inputs should be approximately uniform within the optimal coordinate system. If the distribution is significantly different then the representation is not optimal for the actually occurring environment statistics.

6 CONCLUSIONS

The re-interpretation of population codes as representing sampled probability distributions allows a set of powerful techniques from probability theory to be applied. In particular, we can access mapping theorems on probability Hilbert spaces, as well as results from Information Theory. I propose that this re-interpretation allows the development of a computational theory of population codes within which we can formulate supervised and unsupervised learning algorithms, determine optimality, and interpret neurophysiological data.

References

Andersen R. A., Zipser D., 1988, The role of the posterior parietal cortex in coordinate transformations for Visual-Motor integration, *Can. J. Physiol. Pharmacol.*, 66:488–501.

Georgopoulos A. P., Kettner R. E., Schwartz A. B., 1988, Primate motor cortex and free arm movements to visual targets in three-dimensional space. II. Coding of the direction of movement by a neuronal population, *J. Neurosci*, 8(8):2928–2937.

Georgopoulos A. P., 1986, On reaching, *Ann. Rev. Neurosci.*, 9:147–170.

Kalaska J. F., Cohen D. A. D., Hyde M. L., Prud'homme M., 1989, A comparison of movement direction-Related versus load Direction-Related activity in primate motor cortex, using a two-dimensional reaching task, *J. Neurosci.*, 9(6):2080–2102.

Linsker R., 1988, Self-organization in a perceptual network, *Computer*, 21(3):105–117.

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

Ljung L., Soderstrom T., 1983, *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, MA.

Nadal J.-P., Parga N., 1994, Nonlinear neurons in the low-noise limit: A factorial code maximizes information transfer, *Network*, 5:565–581.

Sanger T. D., 1994, Theoretical considerations for the analysis of population coding in motor cortex, *Neural Computation*, 6(1):12–21.

van Gelder T., 1991, What is the D in PDP? a survey of the concept of distribution, In *Philosophy and Connectionist Theory*, pages 33–59, Lawrence Erlbaum Associates, Hillsdale, NJ.

Widrow B., Hoff M. E., 1960, Adaptive switching circuits, In *IRE WESCON Conv. Record, Part 4*, pages 96–104.