LETTER Communicated by Peter Dayan

Computing with Populations of Monotonically Tuned Neurons

Emmanuel Guigon
guigon@ccr.jussieu.fr
INSERM U483, Université Pierre et Marie Curie, 75005 Paris, France

The parametric variation in neuronal discharge according to the values of sensory or motor variables strongly influences the collective behavior of neuronal populations. A multitude of studies on the populations of broadly tuned neurons (e.g., cosine tuning) have led to such well-known computational principles as population coding, noise suppression, and line attractors. Much less is known about the properties of populations of monotonically tuned neurons. In this letter, we show that there exists an efficient weakly biased linear estimator for monotonic populations and that neural processing based on linear collective computation and least-square error learning in populations of intensity-coded neurons has specific generalization capacities.

1 Introduction

Observing how neuronal discharge varies according to the value of a parameter has led to the distinction between two coding schemes (Ballard, 1986): (1) value coding, defined as a selective response of the neuron to an isolated part of the parameter range (receptive field), and (2) intensity coding, which is the representation of a parameter by monotonic variations in discharge frequency. These two coding schemes initially arise from the properties of the peripheral apparatus (Martin, 1991); however, they are further elaborated by the nervous system and can be considered as computational schemes, since they have no counterpart at the periphery (Maunsell & van Essen, 1983; Knudsen, du Lac, & Esterly, 1987; Lacquaniti, Guigon, Bianchi, Ferraina, & Caminiti, 1995; Helms Tillery, Soechting, & Ebner, 1996; Bremmer, Pouget, & Hoffmann, 1998). Although simple manipulations could transform monotonically responding neurons into tuned neurons (Bullock, Grossberg, & Guenther, 1993; Salinas & Abbott, 1995), both coding schemes appear to be actively maintained at multiple levels of processing in the nervous system. Thus, the question of the actual computational advantages of each coding scheme arises.

The value coding scheme has been thoroughly investigated (Baldi & Heiligenberg, 1988; Seung & Sompolinsky, 1993; Salinas & Abbott, 1994,

1995; Snippe, 1996; Pouget, Zhang, Deneve, & Latham, 1998; Zhang, Ginzburg, McNaughton, & Sejnowski, 1998; Baraduc & Guigon, 2002), as it closely reflects the distribution of neuronal representations in many sensory and motor systems, where large populations of broadly tuned neurons encode attributes of perception, action as well as cognitive operations (Georgopoulos, Lurito, Petrides, Schwartz, & Massey, 1989; Wilson & McNaughton, 1993). Computational properties are related to those of tabular representations (Atkeson, 1989). Broadly tuned populations support local learning generalization and interpolation (Ghahramani, Wolpert, & Jordan, 1996) but only limited extrapolation (Ghahramani et al., 1996; DeLosh, Busemeyer, & McDaniel, 1997).

The intensity coding scheme (e.g., sigmoid response functions) has been considered, together with the value coding scheme (e.g., gaussian response functions), in the general field of function approximation and basis functions (Girosi, Jones, & Poggio, 1995; Hornik, Stinchcombe, & White, 1989; Pouget & Sejnowski, 1994). In this framework, universal approximation capacities have been attributed to families of sigmoid and gaussian functions (Girosi et al., 1995; Hornik et al., 1989). They are also believed to share similar computational properties (Pouget & Sejnowski, 1997), since the sigmoid functions can be combined to reconstruct tuned functions (Girosi et al., 1995; Salinas & Abbott, 1995). The intensity coding scheme has been used to represent eye-, head-, and body-related postural parameters in the framework of sensorimotor transformations (Zipser & Andersen, 1988; Pouget & Sejnowski, 1994, 1997; Salinas & Abbott, 1995) for physiological reasons, but no particular computational role was assigned to it.

Intuitively, the use of monotonic response functions is closely related to the notion of structured representation (Atkeson, 1989), since manipulated variables are readily available in the discharge frequency of input and output neurons. Thus, an expected property of collective computation in populations of monotonically responding neurons is the global generalization of learning to novel situations (Baraduc, Guigon, & Burnod, 2001; Guigon & Baraduc, 2002). In this article, we first analyze linear decoding methods for monotonic populations. We show that a particular estimator (summation estimator) has interesting computational properties despite intrinsic limitations due to the nature of the coding scheme. Then we show that neural processing based on linear collective computation and least-square error learning in populations of intensity-coded neurons has built-in induction capacities that are strikingly different from those provided by populations of unimodal neurons.

2 Encoding

In the following, we consider a population of N neurons. Each neuron i has a discharge x_i whose mean value varies monotonically with a parameter x

in $[a; b]$ according to

$$x_i = f_i(x) = f(x, \lambda_i, s_i) = \frac{D_i}{1 + e^{-(x-\lambda_i)/s_i}}, \tag{2.1}$$

where λ_i characterizes the recruitment threshold of the neuron, s_i is related to the steepness of f, and D_i is the maximum discharge rate. The λ_i are considered as either a set of fixed numbers or realizations of a random variable with a uniform distribution (in this latter case, index i is removed). In general, we take $[a; b] = [0; 1]$, $s_i = s$, $D_i = 1$.

The sigmoid function was chosen to permit analytical derivations, although other similar functions would lead to qualitatively similar results.

3 Decoding

In this section, we discuss how a population of monotonically responding neurons can be decoded. A mechanism suitable for biological computation should be as simple as possible and preferably linear (Salinas & Abbott, 1994). A possible decoding method is the unbiased optimal linear estimator (uOLE), which provides the smallest decoding bias of any linear method (Salinas & Abbott, 1994; Pouget et al., 1998). The uOLE is defined by the set of weights $\mathbf{w} = \{w_i\}$ that minimizes

$$\int \left(x - \sum_i w_i x_i \right)^2 P(\mathbf{x} \mid x) P(x)\, dx\, d\mathbf{x},$$

where $\mathbf{x} = \{x_i\}$ are the observed firing rates, $P(\mathbf{x} \mid x)$ the probability of these responses given the value x of the encoded parameter, and $P(x)$ the probability of the parameter. The weights were calculated numerically for different s using $\mathbf{Q}\mathbf{w} = \mathbf{l}$, where $\mathbf{Q}$ is the correlation matrix of firing rates, and $\mathbf{l}$ the vector containing the center-of-mass of response functions (Salinas & Abbott, 1994). For uncorrelated gaussian noise, the variance was calculated as

$$\sigma^2 \sum_i w_i^2,$$

where σ is the standard deviation (SD) of noise. We observed that the variance of the uOLE is at least one order of magnitude larger than the smallest possible variance of any estimator (see the derivation below). Thus, the uOLE is an unbiased but highly inefficient decoding method.

The OLE that minimizes a sum of squared bias and variance provides an optimal trade-off between bias and variance. It can be obtained as a single-layer neural network $y = \mathbf{w}^T\mathbf{x}$, which is trained to recover the encoded value x in observed firing rates $\mathbf{x} = \{x_i\}$ by using

$$(x - \mathbf{w}^T\mathbf{x})^2 + \alpha \mathbf{w}^T\mathbf{w} \tag{3.1}$$

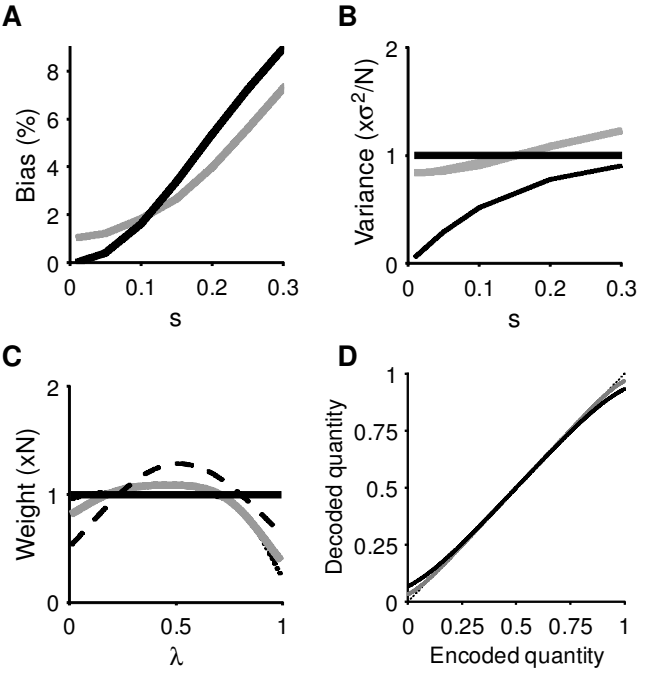

Figure 1: (A) Bias (mean decoding error over [0;1] in % of the range) of the summation estimator (black) and the OLE (gray) as a function of s. (B) Variance of the summation estimator (black) and the OLE (gray). Minimal variance of biased estimators (thin line). Parameter $\sigma = 0.5$. (C) Weight profile for the summation estimator (thick black) and the OLE ($s = .05$: dotted; $s = .1$: gray; $s = .2$: dashed). Note that the results for the OLE depend on α ($\alpha = 1$). α was chosen so that bias and variance for the OLE are in the range of bias and variance of the summation estimator. (D) Noise-free decoding of a population of $N = 100$ neurons using the summation estimator. The dark line is for $s = 0.1$. The gray line is for $s = 0.05$.

as an objective function and α as a parameter. Numerical simulations show that the bias of the OLE is a small fraction of the range (see Figure 1A), and its variance (under gaussian noise) is of the same order of magnitude as the smallest possible variance over a large range of s (see Figure 1B). However, the structure of the OLE is a complex function of both λ for a given s and of s. This is illustrated in Figure 1C, where the weights defining the OLE (see equation 3.1) are plotted as a function of the recruitment threshold. For comparison, Figure 1 also shows the results for the summation estimator (SE), which is introduced in equation 3.3. Note in particular the simpler, uniform structure of the SE (see Figure 1C).

To derive a simpler but yet efficient decoding method for a monotonic population, we calculated the maximum likelihood (ML) estimate for the parameter x based on observed firing rates $\mathbf{x}$. For gaussian noise (SD σ), we obtained

$$\frac{\partial}{\partial x}[\ln P(\mathbf{x} \mid x)] = \frac{1}{\sigma^2} \sum_{i=1}^{N} f_i'(x)[x_i - f_i(x)] = 0.$$

For linear response functions ($x_i = f_i(x) = x + 0.5 - \lambda_i$), this equation led to

$$x = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{3.2}$$

as

$$\frac{1}{N} \sum_{i=1}^{N} \lambda_i = 0.5$$

for uniformly distributed λ_is.

We can arrive at the same result letting $s \to 0$ and $N \to +\infty$. In this case, the response function is the Heaviside function $\mathcal{H}(x - \lambda)$, and we search for a linear estimator,

$$L(x) = \int_0^1 w(\lambda)\mathcal{H}(x - \lambda)\, d\lambda.$$

From

$$\frac{dL(x)}{dx} = \int_0^1 w(\lambda)\delta(x - \lambda)\, d\lambda = w(x)$$

and $L(x) = x$, we obtain $w(\lambda) = 1$. Thus,

$$L(x) = \int_0^1 \mathcal{H}(x - \lambda)\, d\lambda,$$

which is the continuous version of equation 3.2. Here, $L(x)$ is, in fact, the OLE, since it is the only unbiased linear estimator.

For the general nonlinear case, we consider the following linear estimator (summation estimator, SE),

$$L_N(x, s) = \frac{1}{N} \sum_{i=1}^{N} x_i. \tag{3.3}$$

Noise-free decoding on [0; 1] is illustrated in Figure 1D for two slopes ($s = 0.1$ and $s = 0.05$). Decoded quantity was close to encoded quantity in both cases. Mean error was 1.5% and 0.4% of the range, respectively. Errors were

confined to the extreme parts of the coding range. These results suggest that SE is close to an unbiased estimator for small s.

For large N, the SE can be written as

$$L(x, s) = \int_0^1 f(x, \lambda, s)\, d\lambda.$$

Thus,

$$L(x, s) = 1 - s \ln \frac{1 + e^{(1-x)/s}}{1 + e^{-x/s}}. \tag{3.4}$$

If we let $s \to 0$, then $L(x, s) \to x$. Thus, $L(x, s)$ is an unbiased estimator of x for small s.

The quality of the SE is determined by its bias and variance. The bias was calculated from equation 3.4 as a function of s (see Figure 1A). The bias remained below 2% for $s < 0.1$ and increased quasi-linearly and rapidly for $s > 0.1$.

The variance can be calculated analytically and compared to the minimal variance of all decoding methods (V_{CR}, Cramér-Rao bound). For gaussian noise, we have

$$V_{CR}(x, s) = \frac{(1 + \partial b(x, s)/\partial x)^2}{I(x, s)},$$

where

$$b(x, s) = L(x, s) - x$$

is the bias and

$$I(x, s) = \frac{1}{\sigma^2} \sum_{i=1}^{N} f'(x, \lambda_i, s)^2$$

is the Fisher information, f' is the derivative with respect to x. Using the continuous approximation, we obtain

$$I(x, s) = \frac{N}{\sigma^2} \int_0^1 f'(x, \lambda, s)^2\, d\lambda = \frac{N}{s\sigma^2} \left(F\left(\frac{x}{s}\right) - F\left(\frac{x-1}{s}\right) \right),$$

where

$$F(u) = \frac{1}{3(1 + e^u)^3} - \frac{1}{2(1 + e^u)^2}.$$

Approximation for small s gives

$$V_{CR}(x, s) \approx \frac{6s\sigma^2}{N}.$$

For comparison, the Cramér-Rao variance in the case of a population of broadly tuned neurons is (with the same notations) $2s\sigma^2/N\sqrt{\pi}$ (s is the width of the tuning curve and σ the SD of noise; Snippe, 1996).

The variance of the SE is

$$V_{SE} = \mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N} f(x, \lambda_i, s)\right) = \frac{1}{N^2}\sum_{i=1}^{N} \mathrm{Var}(f) = \frac{\sigma^2}{N} \tag{3.5}$$

and is independent of s. The variances are plotted together in Figure 1A. For low values of s, the SE is unbiased, but its variance is far larger than the minimum attainable variance (e.g., 30 times for $s = 0.005$). For larger s ($s > 0.05$), the SE is biased, but its variance is closer to the minimum variance. For $s = 0.1$, the estimator variance is approximately twice the minimum variance (in comparison, the variance of the population vector in similar conditions is more than five times the minimum variance; Pouget et al., 1998). Similar calculations can be made for Poisson noise (see the appendix).

The SE and the OLE have a comparable decoding performance, although they rely on a different bias-variance trade-off (see Figures 1A and 1B). The main difference lies in the very simple structure of the SE (see Figure 1C), which makes it attractive for neural computation.

4 Extensions

This scheme can be immediately extended in three directions:

1. The maximum discharge rate (D_i) is not normalized. A generalized SE is obtained after normalization by D_i.
2. The steepness of the response function varies among neurons. The decoding method still applies if the distribution of steepness is the same for each λ_i.
3. The maximum discharge rate varies among neurons. Again, the decoding method applies if the distribution of D_i is the same for each λ_i.

In fact, the decoding method can work appropriately in more general conditions (e.g., uniformly distributed s_i and D_i in some intervals). This is illustrated in Figure 2A for the set of response functions in Figure 2B. The OLE is not as flexible as the SE since the weights vary with s_i and thus must be chosen appropriately for each neuron.

A more general description of neuronal discharge (see equation 2.1) should include a baseline firing rate,

$$x_i = f_i(x) + b_i, \tag{4.1}$$

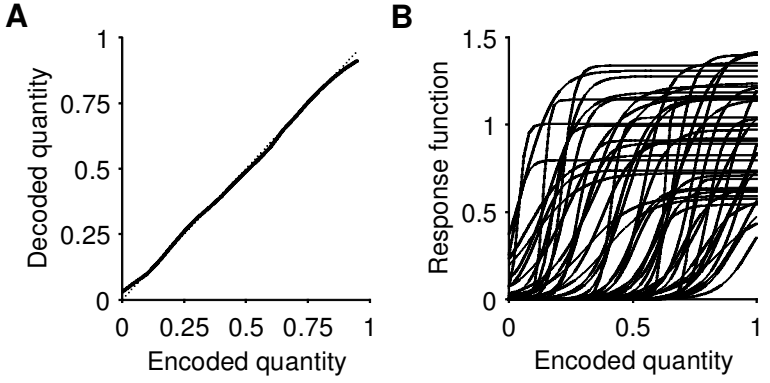

Figure 2: (A) Noise-free decoding of a population of $N = 100$ neurons with uniform distribution of slope (s_i) in [0.01;0.1] and uniform distribution of maximal activity (D_i) in [0.5;1.5] (see equation 2.1). (B) Ensemble of response functions.

where $\{b_i\}$ are realizations of a random variable b. We immediately find that for large N, a generalized estimator is

$$\frac{1}{N}\sum_{i=1}^{N} x_i - \langle b \rangle,$$

since $(1/N)\sum b_i$ tends in probability toward $\langle b \rangle$ when N tends to infinity (as long as b has finite first- and second-order moments). This estimator has the same bias and variance as the SE. Simulations show that for the OLE, the bias is larger with equation 4.1 than with equation 2.1, although the variance is the same.

A main limitation to the efficiency of the SE is related to the fact that saturated units provide no Fisher information about the stimulus, since each neuron contributes to the variance of the estimate in proportion to the derivative of the response function (Seung & Sompolinsky, 1993; Pouget et al., 1998). This problem is intrinsic to the nature of the response function. However, this property could be profitably exploited. Since the derivative of the response function is close to zero outside [0; 1] over a large range of s, an encoding-decoding method based on a larger coding interval can be used to reduce the decoding bias while preserving a consistent relation to the lower bound of decoding methods.

5 Monotonic Versus Unimodal Tuning

A unimodal response function can be obtained by subtracting two monotonic response functions with different recruitment thresholds. A unimodal population profile can be obtained by a linear transformation (derivation)

on a monotonic population profile. Although these operations could likely be realized by the nervous system, we explain why it may not be always profitable to transform monotonic profiles into unimodal profiles. We consider the neural representation of a real function $y = h(x)$ (for simplicity $x \in [0; 1]$) obtained as a linear mapping $y = \mathbf{w}^T\mathbf{x}$, where $\mathbf{x} = \{x_i\} \in \mathbb{R}^N$ is a vector obtained from x by an encoding scheme (see below) and $\mathbf{w} \in \mathbb{R}^N$ a vector of weights. The weights were identified by least-square error learning for a given training set $\{x_t, y_t = h(x_t), 1 \leq t \leq P\}$. The theoretical form of the weights was $\mathbf{w} = \langle \mathbf{x}\mathbf{x}^T \rangle^{-1} \langle \mathbf{x}y \rangle$, where $\langle\rangle$ denotes expectation over the training set (Shynk, 1995).

We compared two encoding schemes, a monotonic scheme and a gaussian-like unimodal scheme, and in order to do so properly, an equivalence between the slope of the former scheme and the width of the latter must be established. From an information-theoretic point of view, a sigmoid-like function can be considered as the response function of a neuron that maximizes output entropy for a given gaussian-like probability distribution of inputs. This is actually the case whenever the sigmoid function is the integral of the input distribution (Dayan & Abbott, 2001). Thus, we chose

$$x_i = \exp\left(-\frac{(x - \lambda_i)^2}{2\sigma^2}\right)$$

for the gaussian scheme and

$$x_i = \frac{1}{2}\left[\mathrm{erf}(-\frac{(x - \lambda_i)}{\sigma\sqrt{2}}) + 1\right]$$

for the monotonic scheme. Similar results would be obtained using the sigmoid function (see equation 2.1) and its derivative for the monotonic and gaussian schemes, respectively.

The results for learning a function from a restricted training set ($P = 2$) are shown in Figure 3. Although the shape of the induced function depends on the width and slope of the response function, the monotonic and unimodal schemes have clearly different generalization capacities.

6 Discussion

We have shown that populations of monotonically responding neurons (those that respond according to an intensity code) can be efficiently manipulated by a simple scheme, SE. This method is attractive, since it is linear and can be easily computed by a neuron and could allow the nervous system to represent a variable without having to convert rate codes into place codes. However, the SE has two shortcomings: (1) it is biased (see Guigon & Baraduc, 2002), and (2) it is not optimal. With respect to the first point, in ML estimation, each neuron contributes to the variance of the estimate in

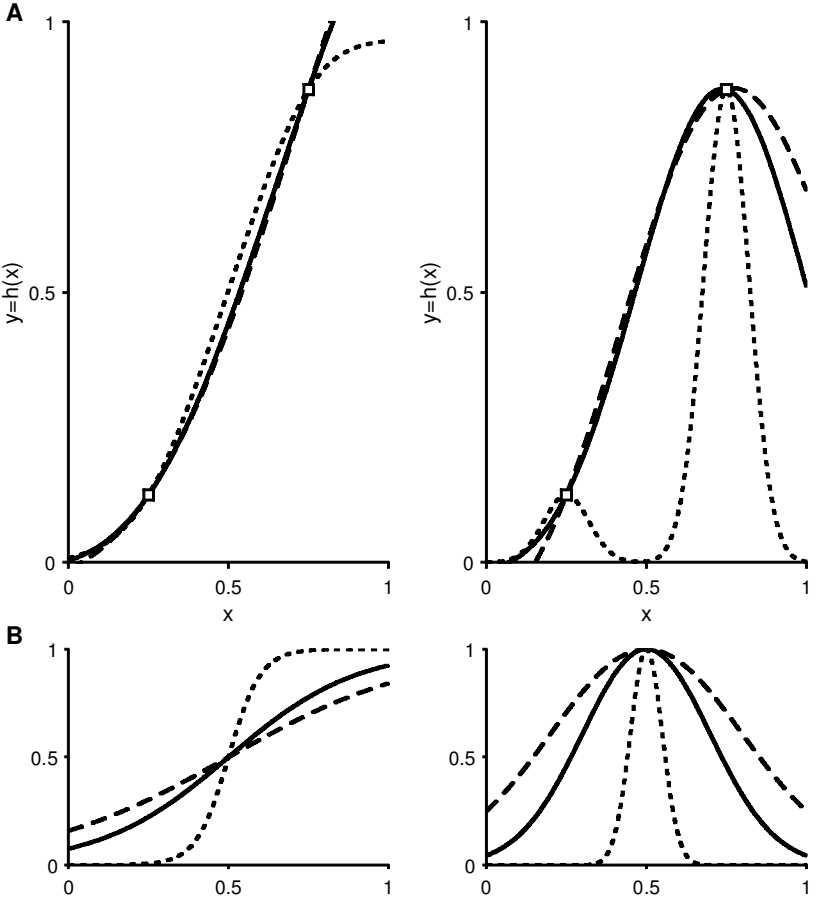

Figure 3: Induced functions (A) for three monotonic response functions (B, left panel), and three gaussian response functions (B, right panel): $\sigma = .2$ (solid), $\sigma = .05$ (dotted), $\sigma = .3$ (dashed). The training pairs are indicated by a square.

proportion to the derivative of the response function (Seung & Sompolinsky, 1993; Pouget et al., 1998). Since this derivative is close to zero outside [0; 1] over a large range of s, an encoding-decoding method based on a larger coding interval could be used to reduce the decoding bias, while preserving a consistent relation to the lower bound of decoding methods. With respect to the second point, the SE appears to be an efficient linear estimator in the sense that it offers an almost optimal bias-variance compromise. Furthermore, its variance is independent of the steepness of the response function (see equation 3.5 and Figure 1B). This property allows the SE to work with

equal efficiency for all the inputs. Other estimators do not have such a property as their efficiency can change dramatically with the width of the input (Seung & Sompolinsky, 1993; Pouget et al., 1998).

Although computation with monotonic response functions could in theory be replaced by computation with unimodal response functions, we have shown that the two schemes have different generalization capacities. This issue can be relevant for a better understanding of how the nervous system handles novel situations using limited prior experience (Bedford, 1989; DeLosh et al., 1997; Shinn-Cunningham, Durlach, & Held, 1998; Guigon & Baraduc, 2002).

Appendix: Poisson Noise

For uncorrelated Poisson noise, Fisher information is

$$I^P(x, s) = \sum_{i=1}^{N} \frac{f'(x, \lambda_i, s)^2}{f(x, \lambda_i, s)}.$$

Using the continuous approximation, we obtain

$$\begin{aligned} I^P(x, s) &= N \int_0^1 \frac{f'(x, \lambda, s)^2}{f(x, \lambda, s)}\, d\lambda \\ &= f(x, 1, s) - f(x, 0, s) - 0.5[f^2(x, 1, s) - f^2(x, 0, \lambda)]. \end{aligned}$$

Approximation for small s gives

$$V^P_{CR}(x, s) \approx \frac{2s}{N}.$$

The variance of the SE is

$$V^P_{SE}(x) = \mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N} f(x, \lambda_i, s)\right) = \frac{1}{N^2}\sum_{i=1}^{N} \mathrm{Var}(f) = \frac{L(x, s)}{N}.$$

For small s,

$$V^P_{SE}(x) = \frac{x}{N}$$

and is independent of s.

Acknowledgments

I thank Pierre Baraduc, Romain Brette, and Etienne Koechlin for fruitful discussions and Christine Lienhart for revising my English.

References

Atkeson, C. (1989). Learning arm kinematics and dynamics. *Annual Review of Neuroscience*, *12*, 157–183.

Baldi, P., & Heiligenberg, W. (1988). How sensory maps could enhance resolution through ordered arrangements of broadly tuned receivers. *Biological Cybernetics*, *59*, 313–318.

Ballard, D. (1986). Cortical connections and parallel processing. *Behavioral and Brain Sciences*, *9*, 67–120.

Baraduc, P., & Guigon, E. (2002). Population computation of vectorial transformations. *Neural Computation*, *14*(4), 845–871.

Baraduc, P., Guigon, E., & Burnod, Y. (2001). Recoding arm position to learn visuomotor transformations. *Cerebral Cortex*, *11*(10), 906–917.

Bedford, F. (1989). Constraints on learning new mappings between perceptual dimensions. *Journal of Experimental Psychology: Human Perception and Performance*, *15*, 232–248.

Bremmer, F., Pouget, A., & Hoffmann, K. P. (1998). Eye position encoding in the macaque posterior parietal cortex. *European Journal of Neuroscience*, *10*(1), 153–160.

Bullock, D., Grossberg, S., & Guenther, F. (1993). A self-organizing neural model of motor equivalence reaching and tool use by a multijoint arm. *Journal of Cognitive Neuroscience*, *5*, 408–435.

Dayan, P., & Abbott, L. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems.* Cambridge, MA: MIT Press.

DeLosh, E., Busemeyer, J., & McDaniel, M. (1997). Extrapolation: The sine qua non for abstraction in function learning. *Journal of Experimental Psychology: Learning Memory and Cognition*, *23*(4), 968–986.

Georgopoulos, A., Lurito, J., Petrides, M., Schwartz, A., & Massey, J. (1989). Mental rotation of the neuronal population vector. *Science*, *243*, 234–236.

Ghahramani, Z., Wolpert, D., & Jordan, M. (1996). Generalization to local remappings of the visuomotor coordinate transformation. *Journal of Neuroscience*, *16*(21), 7085–7096.

Girosi, F., Jones, M., & Poggio, T. (1995). Regularization theory and neural networks' architectures. *Neural Computation*, *7*(2), 219–269.

Guigon, E., & Baraduc, P. (2002). A neural model of perceptual-motor alignment. *Journal of Cognitive Neuroscience*, *14*(4), 538–549.

Helms Tillery, S., Soechting, J., & Ebner, T. (1996). Somatosensory cortical activity in relation to arm posture: Nonuniform spatial tuning. *Journal of Neurophysiology*, *76*(4), 2423–2438.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*, 359–366.

Knudsen, E., du Lac, S., & Esterly, S. (1987). Computational maps in the brain. *Annual Review of Neuroscience*, *10*, 41–65.

Lacquaniti, F., Guigon, E., Bianchi, L., Ferraina, S., & Caminiti, R. (1995). Representing spatial information for limb movement: Role of area 5 in the monkey. *Cerebral Cortex*, *5*(5), 391–409.

Martin, J. (1991). Coding and processing of sensory information. In E. Kandel, J. Schwartz, & T. Jessel (Eds.), *Principles of neural science*, (3rd ed., pp. 329–340). New York: Elsevier.

Maunsell, J., & van Essen, D. (1983). Functional properties of neurons in middle temporal visual area of the macaque monkey. I. Selectivity for stimulus direction, speed, and orientation. *Journal of Neurophysiology, 49*, 1127–1147.

Pouget, A., & Sejnowski, T. (1994). A neural model of the cortical representation of egocentric distance. *Cerebral Cortex*, *4*, 314–329.

Pouget, A., & Sejnowski, T. (1997). Spatial transformations in the parietal cortex using basis functions. *Journal of Cognitive Neuroscience*, *9*(2), 222–237.

Pouget, A., Zhang, K., Deneve, S., & Latham, P. (1998). Statistically efficient estimation using population coding. *Neural Computation*, *10*(2), 373–401.

Salinas, E., & Abbott, L. (1994). Vector reconstruction from firing rates. *Journal of Computational Neuroscience*, *1*, 89–107.

Salinas, E., & Abbott, L. (1995). Transfer of coded information from sensory to motor networks. *Journal of Neuroscience*, *15*(10), 6461–6474.

Seung, H., & Sompolinsky, H. (1993). Simple models for reading neuronal population codes. *Proceedings of the National Academy of Science USA*, *90*, 10749–10753.

Shinn-Cunningham, B., Durlach, N., & Held, R. (1998). Adapting to supernormal auditory localization cues. II. Constraints on adaptation of mean response. *Journal of the Acoustical Society of America*, *103*(6), 3667–3676.

Shynk, J. (1995). Adaptive filtering. In M. Arbib (Ed.), *The handbook of brain theory and neural networks* (pp. 74–78). Cambridge, MA: MIT Press.

Snippe, H. (1996). Parameter extraction from population codes: A critical assessment. *Neural Computation*, *8*(3), 511–529.

Wilson, M., & McNaughton, B. (1993). Dynamics of the hippocampal ensemble code for space. *Science*, *261*, 1055–1058.

Zhang, K., Ginzburg, I., McNaughton, B., & Sejnowski, T. (1998). Interpreting neuronal population activity by reconstruction: Unified framework with application to hippocampal place cells. *Journal of Neurophysiology*, *79*(2), 1017–1044.

Zipser, D., & Andersen, R. (1988). A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, *331*, 679–684.

Received July 9, 2002; accepted March 5, 2003.