INSTITUTE OF PHYSICS PUBLISHING NETWORK: COMPUTATION IN NEURAL SYSTEMS

Network: Comput. Neural Syst. **13** (2002) 447–456 PII: S0954-898X(02)52994-8

Threshold behaviour of the maximum likelihood method in population decoding

Xiaohui Xie

Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA

E-mail: xhx@ai.mit.edu

Received 22 November 2001, in final form 29 July 2002
Published 16 September 2002
Online at stacks.iop.org/Network/13/447

Abstract

We study the performance of the maximum likelihood (ML) method in population decoding as a function of the population size. Assuming uncorrelated noise in neural responses, the ML performance, quantified by the expected square difference between the estimated and the actual quantity, follows closely the optimal Cramer–Rao bound, provided that the population size is sufficiently large. However, when the population size decreases below a certain threshold, the performance of the ML method undergoes a rapid deterioration, experiencing a large deviation from the optimal bound. We explain the cause of such threshold behaviour, and present a phenomenological approach for estimating the threshold population size, which is found to be linearly proportional to the inverse of the square of the system's signal-to-noise ratio. If the ML method is used by neural systems, we expect the number of neurons involved in population coding to be above this threshold.

1. Introduction

Information in neural systems is often encoded by populations of neurons. By using population codes, neural systems gain robustness against damage to individual or a small number of neurons, and reduce errors caused by neuronal noise [1–3].

Examples of population codes exist in a variety of sensory and motor domains, such as intended movement directions encoded in the primate motor cortex [4], orientations and directions encoded in the primate visual cortex [5], physical locations encoded in hippocampal place cells [6, 7] and wind directions encoded in the cricket cercal sensory system [8].

Several approaches have been proposed for a neural system to 'read out' the population encoded information. The most straightforward is the centre of mass approach, which estimates the encoded information by the weighted sum of the preferred values of individual neurons with weights specified by neural activities [9, 10]. The second approach is the population vector,

 © 2002 IOP Publishing Ltd Printed in the UK

which estimates the information as the phase of the first Fourier component of the neural activity [4, 11, 12]. Although they have the appealing properties of being simple and easy to implement, the above two methods are often not efficient [13]. The maximum likelihood (ML) method estimates the encoded information as the one with the largest likelihood. In terms of Bayesian inference, ML is the maximum *a posteriori* (MAP) method with an uninformative prior on encoded information [1, 3].

Although it is still a subject of debate whether the ML method is used by neural systems, there have been suggestions that ML could be implemented by a biologically plausible recurrent network [12, 14]. The performance of the ML estimator, quantified by the expected square error between the estimated and the true information, reaches an optimal lower bound asymptotically when the number of neurons in the population is sufficiently large, provided that noise in neural responses is uncorrelated [12, 15]. Because of this, the ML method is asymptotically efficient and widely used in engineering signal estimation problems.

Many studies have been performed on the asymptotic properties of the ML method in population decoding by assuming a sufficiently large number of neurons. However, in neural systems the number of neurons involved in one particular encoding scheme is constrained. How many neurons are necessary to guarantee a good ML estimator performance?

In this paper, we address this question by studying the performance of the ML estimator as a function of population size. We show that when the number of neurons decreases below a certain threshold, the performance of the ML estimator deteriorates dramatically. We also present a phenomenological method to estimate this threshold, and find that it is linearly proportional to the inverse of the square of the system's signal-to-noise ratio (SNR).

Although this threshold behaviour has not been described in the context of population coding, it has long been observed in frequency estimation studies of analogue communications [16, 17]. Taking advantage of the simplicity of frequency estimation in the spectrum after Fourier transform, people are able to calculate the distribution of the ML estimated values directly [17]. However, the methods used are specialized to frequency estimation and cannot be generalized to population decoding, in which neural activities typically follow a bell-shaped function on the encoded information. In contrast, we take a phenomenological approach on the threshold estimation that can be applied to many forms of encoding.

Next, we first present some simulation results on the threshold behaviour of the ML method. Then we discuss the cause of such threshold behaviour. Finally, we present an approach to analytically estimate the threshold number.

2. Performance of the maximum likelihood estimator

Consider a population of N neurons with the activities represented by a vector $\boldsymbol{r} = (r_1, r_2, \ldots, r_N)$, r_i being the activity of the ith neuron. Suppose a static stimulus s is encoded in such a population, and the population activities can be modelled by the distribution with probability density specified by $p(\boldsymbol{r}|s)$. Furthermore, assume that for a given stimulus, the activities among all neurons are conditionally independent, so

$$p(\boldsymbol{r}|s) = \prod_{i=1}^{N} p(r_i|s). \tag{1}$$

An estimator $\hat{s}(\boldsymbol{r})$ estimates the value of the true stimulus from the population responses $\boldsymbol{r}$. A quantity measuring the performance of the estimator is the expected square error between the estimated and the true stimulus, denoted by $C(s) \equiv E[(\hat{s}(\boldsymbol{r}) - s)^2]$, where the expectation is with respect to the density function $p(\boldsymbol{r}|s)$.

The estimation error $C(s)$ of any estimator has a strict lower bound given by the Cramer–Rao bound [15],

$$C(s) \geqslant \frac{[1+b'(s)]^2}{I(s)} + b^2(s), \tag{2}$$

where $b(s) \equiv E[\hat{s}(\boldsymbol{r}) - s]$ is the bias of the estimator and $I(s)$ is the Fisher information defined as

$$I(s) \equiv E\left[\frac{\partial}{\partial s}\log p(\boldsymbol{r}|s)\right]^2 = -\sum_{i=1}^{N} E\left[\frac{\partial^2}{\partial s^2}\log p(r_i|s)\right], \tag{3}$$

where the second equality holds provided that $p(r_i|s)$ is twice differentiable in s and the neural activities are conditionally independent. When an estimator is unbiased, the estimation error $C(s)$ is the variance of the estimator and is lowered bounded by the inverse of the Fisher information.

The ML method estimates the stimulus to be the one with the largest likelihood, that is,

$$\hat{s}_{\mathrm{ML}} \equiv \operatorname*{argmax}_s \; p(\boldsymbol{r}|s) = \operatorname*{argmax}_s \; \mathcal{L}(\boldsymbol{r}; s), \tag{4}$$

where the log likelihood function $\mathcal{L}(\boldsymbol{r}; s) \equiv \log p(\boldsymbol{r}|s) = \sum_i \log p(r_i|s)$.

The ML estimator is unbiased and efficient in the large N limit, reaching the Cramer–Rao bound asymptotically [15, 18]. More precisely, $\hat{s}_{\mathrm{ML}}$ converges in distribution to the normal distribution with mean s and variance $1/I(s)$ when $N \to +\infty$ [18]. Since the Fisher information is an extensive quantity, that is, $I(s) \sim N$, the decoding error $C(s) \sim 1/N$ can be made arbitrarily small by choosing a sufficiently large population size. In this sense, the ML estimator is often called asymptotically efficient.

2.1. Threshold behaviour

Despite the desirable properties of the ML method with a large population size, its performance with small numbers of neurons can be quite different. One way of studying this is to check the changes in the ML performance as the population size is gradually decreased.

We perform this experiment in the following context. Assume the stimulus s is a continuous variable with $s \in [-\pi, \pi)$. An example of such a stimulus is the orientation of a bar or grating, commonly used in visual physiology studies. Furthermore, assume the response of a neuron is a nonlinear function of s plus a Gaussian noise with zero mean and variance σ^2,

$$r_i = f_i(s) + \eta_i \tag{5}$$

for the ith neuron. Here f_i is the tuning curve of this neuron for the stimulus, typically being bell shaped, and η_i is the Gaussian noise added to neuron i. In the following, we take f_i to be the circular normal distribution function:

$$f_i(s) = r_{\max} \exp[-\beta(1 - \cos(s - s_i))], \tag{6}$$

where s_i is the preferred stimulus of neuron i, $r_{\max}$ is the maximum in the tuning curve, and β determines the width of the turning curve. Taking f_i in the above form enforces a periodic boundary condition on the stimulus. Throughout this paper, we assume that the preferred stimuli are uniformly distributed across the whole range of the stimulus. We keep $r_{\max}$ and β fixed and the same for all neurons, and study the performance of the ML method by varying noise variance σ^2. Under these considerations, we can define the SNR of an individual neuron simply as $\gamma \equiv r_{\max}/\sigma$.

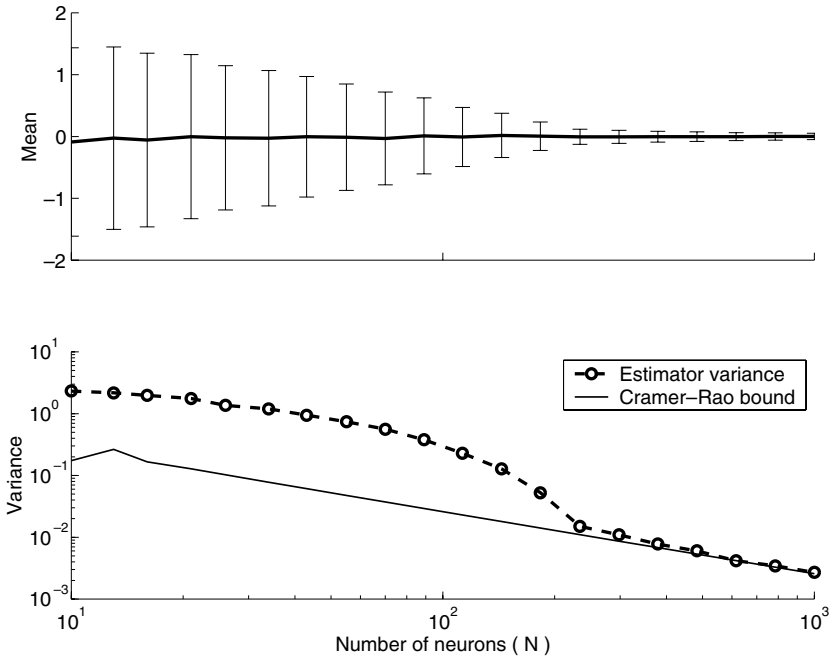

Figure 1. Threshold phenomenon in the performance of the ML estimator. The first panel shows the mean of the ML estimator as a function on the number of neurons (N). The error bar is the standard deviation of the estimator, shown further as a function of N in the second panel, and compared to the Cramer–Rao bound. When N decreases below a certain threshold, the performance of the ML estimator as measured by its variance exhibits a rapid deterioration, deviating rapidly from the Cramer–Rao bound. The tuning function used is circular normal shaped with $r_{\max} = 20$ and $\beta = 8$ for all simulations. The actual stimulus is located at 0. The SNR $\gamma = 1$.

Suppose that the noise added to different neurons is uncorrelated. Then the log likelihood function can be written as

$$\mathcal{L}(\boldsymbol{r}; s) = \sum_{i=1}^{N}\left[-\frac{(r_i - f_i(s))^2}{2\sigma^2} - \frac{1}{2}\ln(2\pi\sigma^2)\right]. \tag{7}$$

Finding the argmax_s of the $\mathcal{L}(\boldsymbol{r}; s)$ in ML method is equivalent to finding $\hat{s}_{\mathrm{ML}} = \operatorname{argmax}_s \mathcal{V}(s)$, with $\mathcal{V}(s)$ defined to be $\mathcal{V}(s) = \sum_{i=1}^{N}[2r_i f_i(s) - f_i^2(s)]/N$, obtained after discarding terms in $\mathcal{L}$ that are independent of s. The evaluation of the log-likelihood $\mathcal{V}(s)$ essentially consists of a template matching.

For this model, we simulated the performance of the ML estimator as N changes. The simulation result is shown in figure 1. Plotted in the top panel is the mean of the ML estimator, which is approximately unbiased when N is large. The variance of the estimator shown in the bottom panel keeps a close track of the Cramer–Rao bound when N is large, however, deviates prominently after N is below a certain threshold.

The threshold number of neurons, above which the ML estimator variance closely tracks the optimal bound, depends critically on the level of noise added to the population responses. As expected, smaller SNRs tend to have higher thresholds (figure 2).

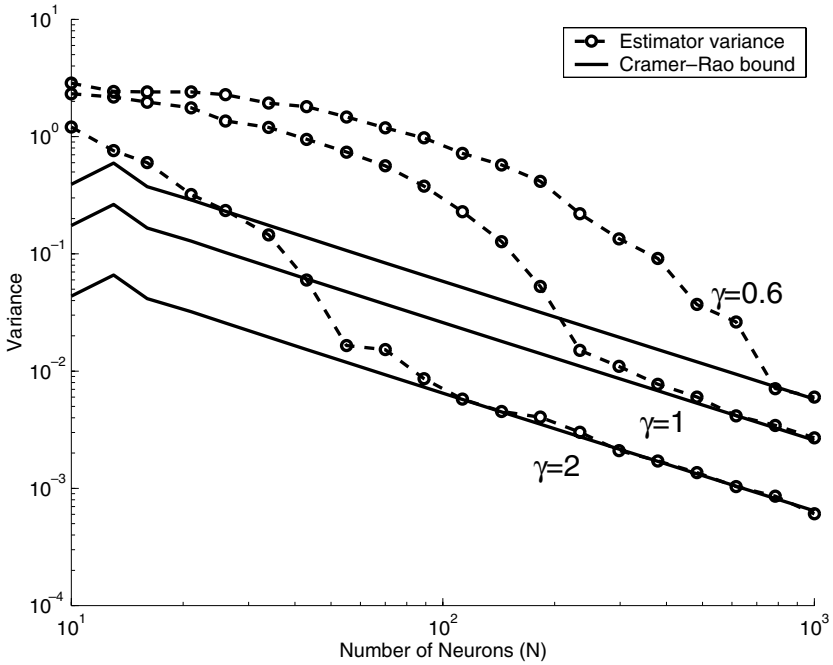

Figure 2. The variance of the ML estimator drawn as a function of N for different SNRs γ. When γ decreases, the threshold number of neurons increases accordingly.

2.2. Origin of the threshold behaviour

Examples of $\mathcal{V}(s)$ using the circular normal shaped tuning function are shown in figures 3(a), (c) for two different SNRs. When the SNR is high ($\gamma = 2$), $\mathcal{V}$ is well peaked around the true stimulus (figure 3(a)). In contrast, when the SNR is small ($\gamma = 0.6$), local maxima away from the actual stimulus can be elevated to a value higher than the one around the stimulus (figure 3(c)). Identification of those peaks as the actual stimulus by the ML method causes a significant increase in the estimation error. The failure of the ML method is due to the fact that the stimulus cannot be reliably detected when the SNR is too small. Such detection failure and its relation to perception have been studied before [19].

For further illustration, we plot the probability density of the peak of $\mathcal{V}(s)$ for the above two cases, obtained from simulations, in figures 3(b), (d). When the noise is small, the probability density is well located around the true stimulus. However, in the large noise case, the probability density distribution includes values across the whole range of stimulus with nonvanishing probabilities. Consequently, the estimation error deviates from the optimal bound.

The probability that the peak of $\mathcal{V}(s)$ occurs at a local maxima away from the actual stimulus depends on the shape of the tuning functions. For bell-shaped f_i, $E[\mathcal{V}(s)]$ is plotted in the dashed curves presented in figures 3(a), (c). Observe that the flank away from the stimulus location is almost flat. In contrast, if f_i was linear, $E[\mathcal{V}(s)]$ would be a quadratic function, and therefore the probability that the peak occurs decays rapidly as the location moves away from the actual stimulus. Indeed, the ML method for linear tuning functions is always unbiased and efficient. The threshold behaviour of the ML method appears only when the tuning functions are nonlinear.

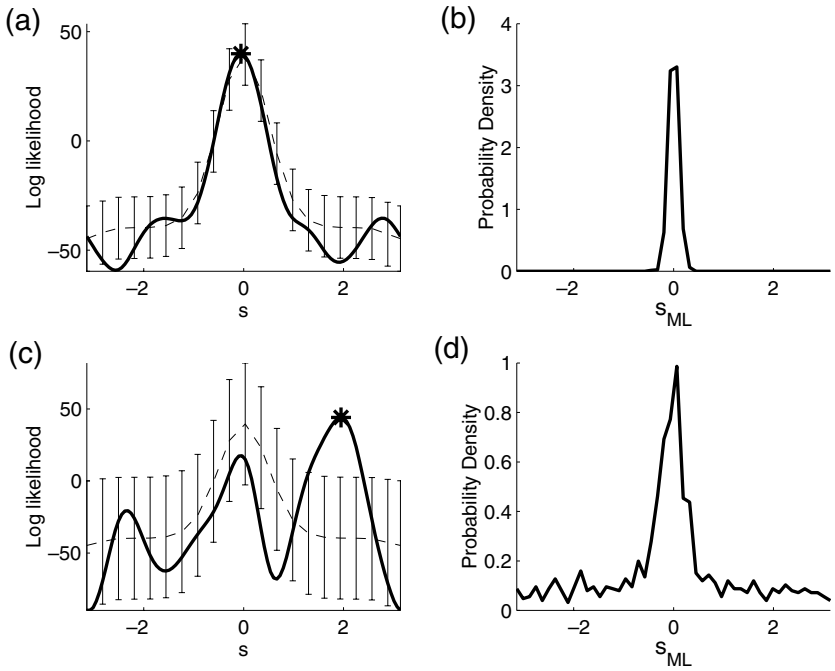

Figure 3. ML estimation in two different SNRs. In (a) and (c) we show examples of likelihood functions $\mathcal{V}(s)$ for SNR $\gamma = 2$ and 0.6 respectively. $N = 80$ for both cases. The dashed curves are expectations of $\mathcal{V}(s)$ with the error bar denoting their standard deviations. The ML estimator values are indicated by '*'. Part (c) demonstrates that a small SNR can produce a peak in the likelihood functions $\mathcal{V}(s)$, far away from the real stimulus, and thus cause a dramatic increase in the estimation error. In (b) and (d) we plot distributions of the estimated $\hat{s}_{\rm ML}$, obtained from simulations, for these two SNRs. In (b) we present the distribution with SNR $\gamma = 2$, in which case the distribution is well localized around the true stimulus 0. In contrast, when SNR is small, as shown in (d) with $\gamma = 0.6$, the estimated $\hat{s}_{\rm ML}$ are distributed, including values far away from real stimulus.

These results are simulated with a population of 80 neurons. We expect the threshold number of neurons for $\gamma = 2$ to be less than 80, and the threshold number for $\gamma = 0.6$ to be larger. Indeed, using a threshold estimation method that we introduce in the next section, we find the threshold number for $\gamma = 2$ is around 50, and for $\gamma = 0.6$ it is roughly 650.

If the ML method is used by neural systems in population decoding, the number of neurons involved in stimulus encoding should be above the threshold number. Therefore, it is important to find a general method to estimate this threshold.

3. Estimation of the threshold

We first present a general threshold estimation method and then apply it to the additive Gaussian noise case discussed above.

Since the ML method estimates $\hat{s}_{\rm ML}$ to be the one with the largest likelihood, that is, $\hat{s}_{\rm ML} = \text{argmax}_s\, \mathcal{L}(\boldsymbol{r}; s)$, at the maximum, $\partial \mathcal{L}(\boldsymbol{r}; \hat{s}_{\rm ML})/\partial s = 0$, which can be rewritten, using the mean value theorem, as

$$\frac{\partial \mathcal{L}(\boldsymbol{r}, \hat{s}_{\rm ML})}{\partial s} = \frac{\partial \mathcal{L}(\boldsymbol{r}; s)}{\partial s} + \frac{\partial^2 \mathcal{L}(\boldsymbol{r}; s)}{\partial s^2}(\hat{s}_{\rm ML} - s) + \frac{1}{2}\frac{\partial^3 \mathcal{L}(\boldsymbol{r}; s^*)}{\partial s^3}(\hat{s}_{\rm ML} - s)^2, \quad (8)$$

for some s^* between s and $\hat{s}_{\rm ML}$, and s is the real stimulus.

The expansion coefficients in equation (8) are all sums of uncorrelated random variables. According to the central limit theorem, when N is large, they fall into the normal distributions under some very general conditions. In particular, $\partial\mathcal{L}(\boldsymbol{r}, s)/\partial s \sim \mathcal{N}(0, I(s))$, where the variance $I(s)$ is the Fisher information.

When N is sufficiently large, the quadratic term in equation (8) is relatively small and can be neglected. In this case, the estimator $\hat{s}_{\mathrm{ML}}$ converges in distribution to the normal distribution, i.e. $\hat{s}_{\mathrm{ML}} - s \sim \mathcal{N}(0, 1/I(s))$.

However, when N is small, the quadratic term in equation (8) cannot be neglected, and contributes to the deviations of the ML estimator from the Cramer–Rao bound. Since when N is above the threshold number, the performance of the ML estimator closely tracks the optimal bound, we should have

$$\left|\frac{1}{2}\frac{\partial^3\mathcal{L}(\boldsymbol{r}, s^*)}{\partial s^3}(\hat{s}_{\mathrm{ML}} - s)^2\right| \ll \left|\frac{\partial^2\mathcal{L}(\boldsymbol{r}, s)}{\partial s^2}(\hat{s}_{\mathrm{ML}} - s)\right|, \tag{9}$$

which implies that

$$|s - \hat{s}_{\mathrm{ML}}| \ll \left|2\frac{\partial^2\mathcal{L}(\boldsymbol{r}, s)}{\partial s^2}\left(\frac{\partial^3\mathcal{L}(\boldsymbol{r}, s^*)}{\partial s^3}\right)^{-1}\right| \tag{10}$$

$$\approx \frac{2\tilde{I}(s)}{\tilde{M}(s^*)}, \tag{11}$$

where the approximation is performed using the expected values of the two terms, with $\tilde{I}(s) = I(s)/N$ being the averaged Fisher information and M defined as

$$\tilde{M}(s^*) = \frac{1}{N}\sum_{i=1}^{N} E\left[\frac{\partial^3}{\partial s^3}\mathcal{L}(r_i; s^*)\right]. \tag{12}$$

When N is above threshold, the variance of $\hat{s}_{\mathrm{ML}}$ can be approximated by

$$|s - \hat{s}_{\mathrm{ML}}| \sim 1\Big/\sqrt{N\tilde{I}(s)}. \tag{13}$$

Therefore, for the ML estimator to closely follow the optimal bound, the following has to be satisfied:

$$N \gg \frac{[\tilde{M}(s^*)]^2}{4[\tilde{I}(s)]^3}, \tag{14}$$

which is derived by substituting equation (13) into (11).

3.1. Uncorrelated additive Gaussian noise

Next we apply our threshold estimation method to the simulation context discussed in section 2.1, with the neural activities given by equation (5).

Since the tuning functions are taken to be of the form $f_i(s) = f(s - s_i)$, where $f(s) \equiv r_{\max}\exp[-\beta(1 - \cos(s))]$, and the preferred stimuli s_i are uniformly distributed, we can approximate the summation of variables in $\tilde{I}(s)$ and $\tilde{M}(s)$ by their integrals.

Therefore, the averaged Fisher information $\tilde{I}(s)$ can be derived as

$$\tilde{I}(s) = \frac{1}{N\sigma^2}\sum_{i=1}^{N} f_i'(s)^2 \approx \frac{1}{\sigma^2}\int_{\Omega} f'(s - s')^2\,\mathrm{d}s', \tag{15}$$

with the integration domain $\Omega = [-\pi, \pi)$.

Similarly, $\tilde{M}(s^*)$ can be calculated using

$$\tilde{M}(s^*) = \frac{1}{N\sigma^2}\sum_{i=1}^{N}\{[f_i(s) - f_i(s^*)]f_i'''(s^*) - 3f_i'(s)f_i''(s^*)\} \tag{16}$$

$$\approx \frac{1}{\sigma^2}\int_{\Omega}\{[f(s-s') - f(s^*-s')]f'''(s^*-s') - 3f'(s-s')f''(s^*-s')\}\,\mathrm{d}s'. \tag{17}$$

Thus far, we have left s^* in the above calculation undetermined. The exact value of s^* depends on the realization of random variables $\boldsymbol{r}$. To find an approximate estimation of the threshold number of neurons, we take the largest value of $M(s^*)$ among all possible values of s^*, and express the threshold number of neurons as

$$N_{th} \approx k\frac{\max\{M(s^*)\}^2}{4[\tilde{I}(s)]^3} \propto \frac{1}{\gamma^2}, \tag{18}$$

where k is a constant scale estimating the contributions from the quadratic term compared to the linear term in equation (9). In an approximation, we choose $k = 1/10$. Note that the exact value of k is not important, since the threshold estimation is only correct in order.

As indicated by equation (6), the threshold number N_{th} is linearly proportional to the inverse of γ^2. To test the quality of this prediction, we run the simulation to numerically find the threshold numbers under different SNRs, and compared the results with the theoretical predictions in equation (18). The results are plotted in figure 4, which shows a good consistency between our theoretical estimations and the numerical simulation results.

The relationship between N_{th} and γ can also be understood in the following way. The evaluation of the log-likelihood function essentially consists of a template matching. The mean response of such template is proportional to $Nr_{\max}$, whereas the standard deviation of the response is proportional to $\sqrt{N}\sigma$. For reliable estimation of the stimulus, the mean should be a few times larger than the standard deviation. Hence, the threshold number N_{th} for reliable estimation should be inversely proportional to γ^2.

The threshold estimation method does not rely on information about specific forms of the tuning functions, except that they need to be smooth and three times differentiable. Therefore, this method is applicable to other forms of bell-shaped or even non-bell-shaped tuning functions. This is possible because we estimate the threshold by examining the consequences of the ML estimation through equation (8), without touching the detailed causes as discussed in section 2.2.

4. Discussion

We have shown that population decoding using ML estimation performs poorly for bell-shaped tuning functions f_i, when the number of neurons is small. When the number of neurons crosses some threshold, the performance improves to nearly optimal, according to the Cramer–Rao bound. We show that this threshold is closely related to the SNR of the system and present a simple method to estimate this threshold.

ML is essentially a template matching method, with the template being the tuning functions, as compared to the population vector approach where the template is sinusoid functions. It is possible for the template matching to be achieved through some mechanism of gradient dynamics implemented in recurrent networks, as suggested by Deneve *et al* [14]. From another perspective, there have been many examples indicating that the brain tends to take optimal or near optimal approaches. In this sense, it is also reasonable to conjecture the utilization of ML estimation by neural systems.

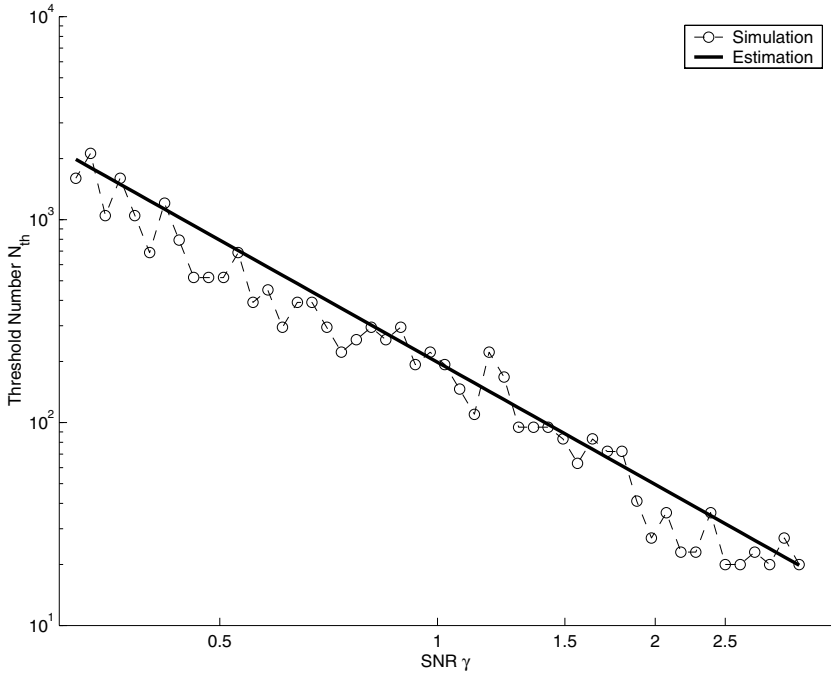

Figure 4. Estimation of the threshold number of neurons N_{th}. The estimated N_{th} are drawn as a function of SNR γ. The solid line is the theoretical prediction; the dashed curve is from simulation. The threshold number from simulation for a particular SNR γ is obtained by measuring the ML estimation variances for 50 population sizes uniformly distributed in a logarithmic scale between 10 and 10^4. Ordering the variances in decreasing order of N, we estimate the threshold number to be the first one after which the ML estimator variance is four times larger than the inverse of the Fisher information.

In this paper, we have assumed that the noise in neural responses is uncorrelated. The correlations may have various different effects on the Fisher information of population codes, depending on the exact forms [20–24]. In general, for correlated noise in neural activities, the ML method is not asymptotically efficient any more, except when the correlation is uniform or local, or the noise is sufficiently small [25]. An interesting open question is how the performance of the ML method depends on the population size when the noise is correlated.

Tuning curves and SNRs usually can be measured experimentally. Suppose the ML method is indeed used by neural systems. The methods presented here give a rough estimate of the minimum number of neurons involved in coding a specific stimulus. Knowing the threshold number is also helpful in practice for the reconstruction of the stimulus from the neuron responses. It sets a limit on how many neurons need to be recorded from experiments in order to get an accurate estimation.

Acknowledgments

I thank S Seung, M Goldman and R Tedrake for very helpful discussions.

References

[1] Dayan P and Abbott L 2001 *Theoretical Neuroscience* (Cambridge, MA: MIT Press)
[2] Pouget A, Dayan P and Zemel R 2000 Information processing with population codes *Nat. Rev. Neurosci.* **1** 125–32

[3] Paradiso M A 1988 A theory for the use of visual orientation information which exploits the columnar structure of striate cortex *Biol. Cybern.* **58** 35–49
[4] Georgopoulos A P, Kalaska J F, Caminiti R and Massey J T 1982 On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex *J. Neurosci.* **2** 1527–37
[5] Maunsell J H and Van Essen D C 1983 Functional properties of neurons in middle temporal visual area of the macaque monkey i. Selectivity for stimulus direction, speed, and orientation *J. Neurophysiol.* **49** 1127–47
[6] O'Keefe J and Dostrovsky J 1971 The hippocampus as a spatial map preliminary evidence from unit activity in the freely-moving rat *Brain Res.* **34** 171–5
[7] Best P J, White A M and Minai A 2001 Spatial processing in the brain: the activity of hippocampal place cells *Annu. Rev. Neurosci.* **24** 459–86
[8] Theunissen F E and Miller J P 1991 Representation of sensory information in the cricket cercal sensory system ii. Information theoretic calculation of system accuracy and optimal tuning-curve widths of four primary interneurons *J. Neurophysiol.* **66** 1690–703
[9] Zohary E 1992 Population coding of visual stimuli by cortical neurons tuned to more than one dimension *Biol. Cybern.* **66** 265–72
[10] Pouget A, Zhang K, Deneve S and Latham P E 1998 Statistically efficient estimation using population coding *Neural Comput.* **10** 373–401
[11] Salinas E and Abbott L F 1994 Vector reconstruction from firing rates *J. Comput. Neurosci.* **1** 89–107
[12] Seung H S and Sompolinsky H 1993 Simple models for reading neuronal population codes *Proc. Natl Acad. Sci. USA* **90** 10 749–53
[13] Snippe H P 1996 Parameter extraction from population codes: a critical assessment *Neural Comput.* **8** 511–29
[14] Deneve S, Latham P E and Pouget A 1999 Reading population codes: a neural implementation of ideal observers *Nat. Neurosci.* **2** 740–5
[15] Cover T M and Thomas J A 1991 *Elements of Information Theory* (New York: Wiley)
[16] Van Trees H 1968 *Detection, Estimation and Modulation Theory* vol 1, 2 (New York: Wiley)
[17] Rife D C and Boorstyn R R 1974 Single-tone parameter estimation from discrete-time observations *IEEE Trans. Inf. Theory* **20** 591–8
[18] Akahira M and Takeuchi K 1981 *Asymptotic Efficiency of Statistical Estimators: Concepts and High Order Asymptotic Efficiency (Springer Lecture Notes in Statistics vol 7)* (Berlin: Springer)
[19] Pelli D G 1985 Uncertainty explains many aspects of visual contrast detection and discrimination *J. Opt. Soc. Am.* A **2** 1508–32
[20] Snippe H and Koenderink J 1992 Information in channel-coded systems: correlated receivers *Biol. Cybern.* **67** 183–90
[21] Shadlen M and Newsome W 1994 Noise, neural codes and cortical organization *Curr. Opin. Neurobiol.* **4** 569–79
[22] Abbott L F and Dayan P 1999 The effect of correlated variability on the accuracy of a population code *Neural Comput.* **11** 91–101
[23] Yoon H and Sompolinsky H 1998 The effect of correlations on the fisher information of population codes *Advances in Neural Information Processing Systems* vol 11 (Cambridge, MA: MIT Press)
[24] Wu S, Nakahara H and Amari S 2001 Population coding with correlation and an unfaithful model *Neural Comput.* **13** 775–97
[25] Wu S, Amari S and Nakahara H 2002 Population coding and decoding in a neural field: a computational study *Neural Comput.* **14** 999–1026