Behavioral/Systems/Cognitive

Marginalization in Neural Circuits with Divisive Normalization

Jeffrey M. Beck,[1,2] **Peter E. Latham,**[1] **and Alexandre Pouget**[2,3]

[1]Gatsby Computational Neuroscience Unit, UCL, London WC1N 3AR, United Kingdom, [2]Department of Brain and Cognitive Sciences, University of Rochester, Rochester, New York 14627, and [3]Departement des Neurosciences Fondamentales, Geneva, CH-1211 Geneva 4, Switzerland

A wide range of computations performed by the nervous system involves a type of probabilistic inference known as marginalization. This computation comes up in seemingly unrelated tasks, including causal reasoning, odor recognition, motor control, visual tracking, coordinate transformations, visual search, decision making, and object recognition, to name just a few. The question we address here is: how could neural circuits implement such marginalizations? We show that when spike trains exhibit a particular type of statistics—associated with constant Fano factors and gain-invariant tuning curves, as is often reported *in vivo*—some of the more common marginalizations can be achieved with networks that implement a quadratic nonlinearity and divisive normalization, the latter being a type of nonlinear lateral inhibition that has been widely reported in neural circuits. Previous studies have implicated divisive normalization in contrast gain control and attentional modulation. Our results raise the possibility that it is involved in yet another, highly critical, computation: near optimal marginalization in a remarkably wide range of tasks.

Introduction

When driving to the airport to catch a flight, it is important to know how long it will take to get there. Driving, however, involves many essentially random events, so it is impossible to know exactly how long any particular trip will take. One can know, however, the probability distribution over driving times. To see how to calculate it, consider a very simple scenario in which you are going to the airport at 5:00 P.M., and you know that traffic either flows normally, and it takes 20 ± 5 min to get to there, or there is an accident, and it takes 60 ± 20 min. If the probability of an accident is 5%, then the probability distribution over driving times is, at least qualitatively, $0.95 \times (20 \pm 5) + 0.05 \times (60 \pm 20)$ min. More formally, if $p(t,C)$ is the joint distribution over the time it takes to get to the airport and traffic conditions, then $p(t)$, the probability distribution over driving times, is found by summing the joint distribution over all (in this case all two) traffic conditions,

$$p(t) = \sum_{c} p(t,C). \tag{1}$$

This computation is known as marginalization, since traffic conditions have been "marginalized" out, leaving only a marginal distribution over driving times. Here traffic conditions are a "nuisance" parameter, because they are something we do not really care about, but may influence our inference about driving time.

Marginalization arises in a wide range of seemingly unrelated computations faced by the brain, such as olfaction, internal models for motor control (Wolpert et al., 1995), tracking of visual objects, model selection in multisensory integration (Körding et al., 2007), function approximation, navigation, visual search, object recognition, causal reasoning in rats (Blaisdell et al., 2006), causal inference in human reasoning (Gopnik and Sobel, 2000; Griffiths and Tenenbaum, 2009), addition of numbers (Cordes et al., 2007), social cognition (Baker et al., 2009), and attentional priming (Zemel and Dayan, 1997). Therefore, understanding the neural basis of marginalization has extremely important implications for a wide array of problems in neuroscience and cognitive science.

Although marginalization looked quite simple in the above example, it is almost always difficult, primarily because the number of variables to be marginalized out is almost always large. For instance, consider the problem of identifying a person in low light. There are many things that do not matter at all, such as light level, location, and speed, and many more that can help to identify the person but are not definitive, such as clothes, hair color, and gait. All of them have to be marginalized out to produce a probability distribution over identity. In the case of neural circuits, variables are typically encoded in population activity, which further complicates the problem.

Here we show that biologically plausible networks can implement marginalization near-optimally for coordinate transformations, object tracking, simplified olfaction, and causal reasoning. In all cases, the networks we use are relatively standard multilayer recurrent networks with an intermediate layer that implements a

Received April 5, 2011; revised Aug. 22, 2011; accepted Aug. 31, 2011.

Author contributions: J.M.B. and A.P. designed research; J.M.B. performed research; J.M.B. analyzed data; J.M.B., P.E.L., and A.P. wrote the paper.

P.E.L. and J.M.B. were supported by the Gatsby Charitable Foundation and A.P. by Multidisciplinary University Research Initiative Grant N00014-07-1-0937, NIDA Grant #BCS0346785, and a research grant from the James S. McDonnell Foundation. This work was also partially supported by award P30 EY001319 from the National Eye Institute. The content of the manuscript is solely the responsibility of the authors and does not necessarily represent the official views of the National Eye Institute or the National Institutes of Health. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Correspondence should be addressed to Alexandre Pouget, Department of Brain and Cognitive Sciences, University of Rochester, Rochester, NY 14627. E-mail: alex@bcs.rochester.edu.

DOI:10.1523/JNEUROSCI.1706-11.2011

0270-6474/11/3115310-10$15.00/0

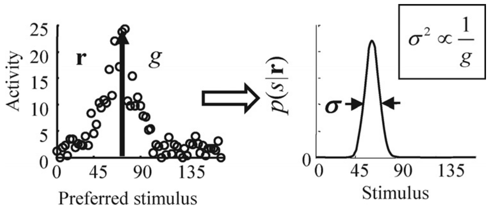

Figure 1. Relationship between neural activity and the posterior distribution for linear probabilistic population codes. The left panel shows the spike count on a single trial in response to a stimulus presented at $s = 67.5$. The height of the hill of activity corresponds to g, or gain. The right panel shows the posterior distribution of s given the activity, $\mathbf{r}$ (obtained through Bayes' rule). The variance of this posterior, σ^2, is inversely proportional to g.

quadratic nonlinearity and divisive normalization, the latter a nonlinearity that is widespread in the nervous system of both vertebrates and invertebrates.

Results

Linear probabilistic population codes

The first step in understanding how networks of neurons perform the kinds of sums (or, in the more general case, integrals) that are required of marginalization is to determine how neurons encode likelihood functions and probability distributions. We focus on a particular type of code—a linear probabilistic population code (PPC) (Ma et al., 2006)—although our approach can be generalized to other distributions over neural variability.

The central idea behind a PPC (linear or nonlinear) is that a neural pattern of activity, $\mathbf{r}$, encodes a function over s, as opposed to a single value of s. The encoded function is the so-called likelihood function, $p(\mathbf{r}|s)$. If the prior over s is flat, we can also think of $\mathbf{r}$ as encoding the posterior, $p(s|\mathbf{r})$; see Figure 1. When considered as a function of $\mathbf{r}$, $p(\mathbf{r}|s)$ specifies the form of the neural variability. This variability is often assumed to be independent and Poisson. Such an assumption, however, fails to capture the behavior of real neurons, which are, typically, not independent, and have Fano factors and coefficients of variation that are inconsistent with the Poisson assumption (Gershon et al., 1998; Maimon and Assad, 2009). For that reason, here we use a broader family, namely the exponential family with linear sufficient statistics, for which the probability distribution of response, $\mathbf{r}$, given the stimulus, s, takes the form

$$p(\mathbf{r}|s) = \phi(\mathbf{r})\exp(\mathbf{h}(s) \cdot \mathbf{r}), \tag{2}$$

where $\mathbf{h}(s)$ is a kernel, "·" denotes the standard dot product, and $\phi(\mathbf{r})$ is essentially arbitrary.

We focus on the class of neural variability described in Equation 2 because it is both consistent with *in vivo* recordings (Ma et al., 2006) and general enough to capture a range of distributions. In particular, independent Poisson neurons fall into this class; for these neurons,

$$\phi(\mathbf{r}) = \frac{\exp\left(-\sum_i f_i(s)\right)}{\prod_i r_i!}, \tag{3}$$
$$h_i(s) = \log f_i(s)$$

where the tuning curves, the $f_i(s)$, satisfy the relationship $\Sigma_i f_i(s) =$ constant. In this case, the ith element of the kernel, $h_i(s)$, is the log of the tuning curve of neuron i. In the general case, however, $h_i(s)$ depends on both the tuning curve and the covariance matrix of the neural response (Ma et al., 2006).

Although the distribution specified in Equation 2 does a better job capturing real neuronal responses than an independent Poisson distribution, it still needs to be extended to handle neuronal responses that depend on contrast (or, in fact, any nuisance parameter). That is because the response pattern associated with the encoding model given in Equation 2 corresponds to a hill of activity whose peak is determined by s, but there is no way, within the context of that model, to adjust the amplitude of the hill. In many realistic cases, however, contrast does just that [orientation-tuned cells in visual cortex being the classic example (Anderson et al., 2000)]. Consequently, we need to augment Equation 2 by allowing ϕ to depend on a set of nuisance parameters, like contrast, which we denote $\mathbf{g}$ because of their tendency to modulate the gain of the population pattern of activity (and, typically, control the variance of the posterior distribution; see Fig. 1). The encoding model we use, therefore, is a slight generalization of Equation 2,

$$p(\mathbf{r}|s,\mathbf{g}) = \phi(\mathbf{r},\mathbf{g})\exp(\mathbf{h}(s) \cdot \mathbf{r}). \tag{4}$$

The nuisance parameters are typically used to adjust the overall gain of the response, although other effects are possible in principle. Whenever the encoding model has the form given in Equation 4, the resulting code is known as a linear PPC: it is a PPC because the neural activity, $\mathbf{r}$, encodes a function of the stimulus, s (as opposed to a single estimate of s), and it is linear because the stimulus-dependent portion of $\log(p(\mathbf{r}|s))$ (the log likelihood) is linear in $\mathbf{r}$.

The fact that the nuisance parameters do not appear in the exponential (i.e., that $\mathbf{h}(s)$ depends on s but not $\mathbf{g}$) is consistent with observations that the value of the stimulus, s, controls the overall shape of the population response (the noisy hill of activity), while the nuisance parameters, like contrast, modulate the amplitude (Sclar and Freeman, 1982; Tolhurst et al., 1983; Albright, 1992; Gur et al., 1997; Buracas et al., 1998; Gershon et al., 1998; Anderson et al., 2000; Mazer et al., 2002). In addition, this fact has computational implications, since it means the posterior, $p(s|\mathbf{r})$ (obtained via Bayes' theorem), is independent of the nuisance parameters. This allows downstream neurons to process the neural activity optimally without having to know the value of the nuisance parameters, and is partly responsible for the near-perfect performance of the networks we describe below.

The problem we address here is as follows: how does the brain take variables represented as linear PPCs (Eq. 4), carry out a particular computation—marginalization—and encode the result as a linear PPC? The last step, encoding the result as a linear PPC, is motivated in large part by the fact that both neural variability and microcircuitry are similar across cortical areas. If the encoding model is also the same across areas, that provides a very parsimonious framework, as it allows essentially the same network operations to be used throughout the brain.

Linear coordinate transformations: theory

The first marginalization we consider is coordinate transformations, a central computation in sensorimotor transformations. We start with a simple transformation: computing the head-centered location of an object, x^{A}, from its eye-centered location, x^{R}, and eye position, x^{E}, a transformation that is given by $x^{\mathrm{A}} = x^{\mathrm{R}} + x^{\mathrm{E}}$. (Although linearity is important to derive exact results, it is not, as we show below, essential.) This transformation involves a marginalization because when we compute the probability distribution over x^{A}, we throw away—marginalize out—all information about x^{R} and x^{E}.

(To see this in a discrete setting, imagine throwing two dice and computing the probability that they sum to 4. This probability is found by enumerating all throws that sum to 4 and adding their probabilities: $p(\text{sum} = 4) = p(3,1) + p(2,2) + p(1,3)$. Coordinate transformations are conceptually the same; the only difference is that the sum on the right side turns into an integral.)

Figure 2*a* shows a network for this sensorimotor transformation. The input to the network consists of two layers: one for eye-centered position and the other for the position of the eyes, both encoded in linear PPCs with bell-shaped tuning curves. Our goal is to wire the network so that the output layer codes for the head-centered location of the stimulus, x^{A}, also in a linear PPC with bell-shaped tuning curves.

The requirement that x^{A} is encoded as a linear PPC implies that

$$p(\mathbf{r}^{\mathrm{A}}|x^{\mathrm{A}},g^{\mathrm{A}}) = \phi(\mathbf{r}^{\mathrm{A}},g^{\mathrm{A}})\exp(\mathbf{h}^{\mathrm{A}}(x^{\mathrm{A}}) \cdot \mathbf{r}^{\mathrm{A}}), \qquad (5)$$

where $\mathbf{r}^{\mathrm{A}}$ is the activity in the output layer. We also want to make the network optimal, in the sense that the output layer carries all the information about x^{A} that is contained in x^{R} and x^{E}. In probabilistic terms, optimality implies that

$$p(x^{\mathrm{A}}|\mathbf{r}^{\mathrm{A}}) = p(x^{\mathrm{A}}|\mathbf{r}^{\mathrm{R}},\mathbf{r}^{\mathrm{E}}), \qquad (6)$$

where $\mathbf{r}^{\mathrm{R}}$ and $\mathbf{r}^{\mathrm{E}}$ are the input layer activities that code for eye-centered location and eye position, respectively, and $\mathbf{r}^{\mathrm{A}}$ is the output of the network (Fig. 2*a*).

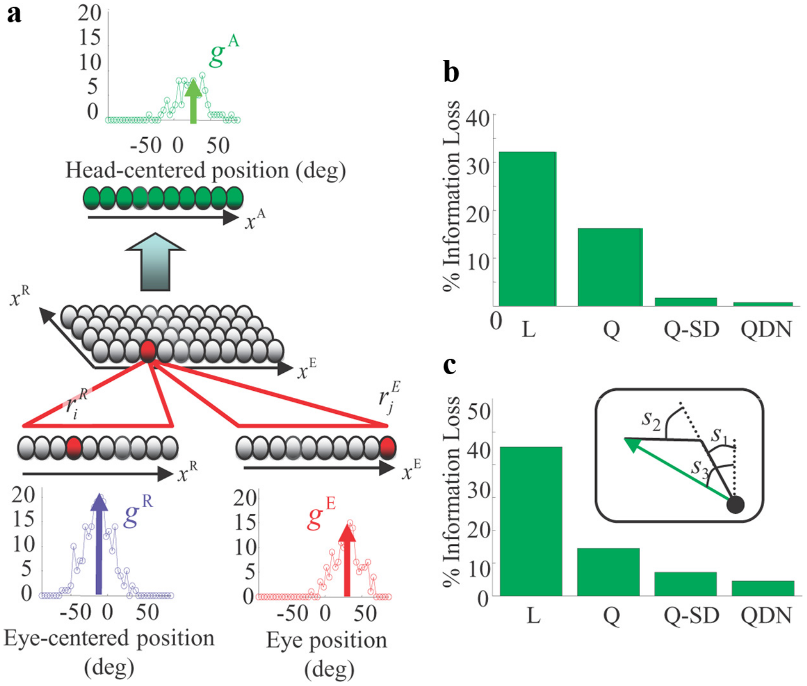

Figure 2. Marginalization for coordinate transformations. ***a***, The network takes as input linear probabilistic population codes for the eye-centered location of an object, $\mathbf{r}^{\mathrm{R}}$, and the current eye position, $\mathbf{r}^{\mathrm{E}}$ (bottom layers), and returns a population code for the head-centered location of the object, $\mathbf{r}^{\mathrm{A}}$ (top layer). The intermediate layer provides a set of basis functions of the eye-centered location and eye position; its activity, r_{ij}, is given in terms of $\mathbf{r}^{\mathrm{R}}$ and $\mathbf{r}^{\mathrm{E}}$ in Equation 9; the relationship between $\mathbf{r}^{\mathrm{A}}$ and r_{ij} is given immediately below that equation. When we perform simulations, we vary the gain of the eye-centered layer from trial to trial to mimic changes in image contrast. ***b***, Bar graph showing the information loss in the output layer compared to the information available in the input layers. QDN, Network with a quadratic divisive nonlinearity in the basis function layer. Information is estimated by a single decoder for all values of contrast. Q-SD, Network with a quadratic nonlinearity but no divisive normalization, using specialized decoders for each value of contrast. Q, Same as Q-SD but with a single decoder to estimate information for all values of contrast. L, Network with a linear rectified activation function and the same decoder for every gain. The QDN network is near optimal even when using a single decoder, which shows that it encodes a near-optimal posterior distribution and does so with a probabilistic population code. ***c***, Same as in ***b*** but for the nonlinear coordinate transform from the joint angles, (s_1, s_2), of a 2-D arm with lengths, (d_1, d_2), to the visual azimuth s_3 of the arm end-point: $s_3 = \tan^{-1}([d_1\sin s_1 + d_2\sin(s_1 + s_2)]/[d_1\cos s_1 + d_2\cos(s_1 + s_2)])$. See Notes.

Note that Equation 6 involves posterior distributions—probability distributions of x^{A} given activity—whereas so far all our analysis has been in terms of likelihoods. The two can be connected via Bayes' theorem; all we need are prior distributions over x^{A}, x^{R}, and x^{E}. Here we assume flat priors over all three variables, so the posteriors are proportional to the likelihoods. This implies that, in the case of a flat prior, a linear PPC encodes both the likelihood function and the posterior distribution.

Although determining the optimal network exactly is nontrivial, we can gain a great deal of insight into the form of the network from just one property of linear PPCs: reliability is proportional to the level of activity, or, more quantitatively (and as illustrated in Fig. 1), the variance of the encoded variable is inversely proportional to the height of the hill of activity (Ma et al., 2006). This is certainly a feature of Poisson neurons, for which higher firing rate means a higher signal-to-noise ratio. It is also relatively easy to see from Equation 4, which tells us that as the amplitude of the neural activity, $\mathbf{r}$, increases the right hand side becomes more and more sharply peaked in s.

The relationship between activity and reliability must hold in all layers of our network, so, using g to denote the overall height of the hill of activity (the gain), we have $\sigma_{\mathrm{A}}^2 \propto 1/g^{\mathrm{A}}$, $\sigma_{\mathrm{R}}^2 \propto 1/g^{\mathrm{R}}$, and $\sigma_{\mathrm{E}}^2 \propto 1/g^{\mathrm{E}}$. For the transformation $x^{\mathrm{A}} = x^{\mathrm{R}} + x^{\mathrm{E}}$, variances add $\sigma_{\mathrm{A}}^2 = \sigma_{\mathrm{R}}^2 + \sigma_{\mathrm{E}}^2$ (assuming no bias). If we substitute the inverse gains for the variances, we obtain $1/g^{\mathrm{A}} = 1/g^{\mathrm{R}} + 1/g^{\mathrm{E}}$, which may be written

$$g^{\mathrm{A}} = \frac{g^{\mathrm{R}}g^{\mathrm{E}}}{g^{\mathrm{R}} + g^{\mathrm{E}}}. \qquad (7)$$

Thus, whatever the form of the network, the gains must transform via a quadratic nonlinearity with divisive normalization.

It is worthwhile emphasizing that, despite its name, we do not use divisive normalization to normalize the probability distribution encoded by the linear PPCs (i.e., we do not use it to ensure that the probabilities sum up to 1). Nor do we use it to obtain a neural response that is independent of the nuisance parameters, like contrast (indeed, contrast has a multiplicative impact on the gain of the output activity). Instead, we use it to obtain a linear PPC in the output layer, that is, a pattern of activity that can be mapped linearly onto a log probability using a decoder that is independent of the nuisance parameter.

We cannot, of course, translate directly from the behavior of the gains to the exact form of the network—after all, the gain is just one number, whereas the population activity is characterized by the activity of many neurons. However, because activity is proportional to gain, we expect that the network will be the high-dimensional analog of Equation 7; one such analog is

$$r_k^{\mathrm{A}} = \frac{\sum_{ij} w_{ij}^k r_i^{\mathrm{R}} r_j^{\mathrm{E}}}{\sum_l c_l^{\mathrm{R}} r_l^{\mathrm{R}} + c_l^{\mathrm{E}} r_l^{\mathrm{E}}}, \tag{8}$$

where the w's and c's are parameters. Here r_i^{R} and r_j^{E}correspond to activity in the input layer and r_k^{A} to activity in the output layer (see Fig. 2*a*). As we show below (Eqs. 10–14), the network is optimal (Eq. 6) and x^{A} is encoded in a linear PPC (Eq. 5) if three conditions are met: first, the transformation is linear (which it is in our case: $x^{\mathrm{A}} = x^{\mathrm{R}} + x^{\mathrm{E}}$); second, the distributions $p(x^{\mathrm{R}}|\mathbf{r}^{\mathrm{R}})$ and $p(x^{\mathrm{E}}|\mathbf{r}^{\mathrm{E}})$ are Gaussian in x^{R} and x^{E}; and third, the weights are chosen correctly in Equation 8. Moreover, Equation 8 is easily implemented by incorporating a two-dimensional intermediate layer in which the activity of neuron *ij*, denoted, r_{ij}, is given by

$$r_{ij} = \frac{r_i^{\mathrm{R}} r_j^{\mathrm{E}}}{\sum_l c_l^{\mathrm{R}} r_l^{\mathrm{R}} + c_l^{\mathrm{E}} r_l^{\mathrm{E}}}. \tag{9}$$

This is the layer shown in Figure 2*a*. Given this relationship, we see from Equation 8 that the activity of the output neurons, r_k^{A}, is given by $r_k^{\mathrm{A}} = \Sigma_{ij} w_{ij}^k r_{ij}$.

Readers not interested in the mathematical details needed to demonstrate that Equation 8 does indeed implement an optimal network may want to skip Equations 10–14. For those who are interested, we start by noting that for Gaussian likelihood functions and a linear PPC, the encoding model has the form

$$p(\mathbf{r}^{\mathrm{M}}|x^{\mathrm{M}},\mathbf{g}^{\mathrm{M}}) = \tilde{\phi}(\mathbf{r}^{\mathrm{M}},\mathbf{g}^{\mathrm{M}})\exp\left[-\frac{(x^{\mathrm{M}} - \mu_{\mathrm{M}})^2}{2\sigma_{\mathrm{M}}^2}\right], \tag{10}$$

where M, which stands for modality, can be either A, R, or E, and

$$\mu_{\mathrm{M}} = \frac{\mathbf{b}^{\mathrm{M}} \cdot \mathbf{r}^{\mathrm{M}}}{\mathbf{a}^{\mathrm{M}} \cdot \mathbf{r}^{\mathrm{M}}} \tag{11}$$

and

$$\sigma_{\mathrm{M}}^2 = \frac{1}{\mathbf{a} \cdot \mathbf{r}^{\mathrm{M}}}. \tag{12}$$

Here $\mathbf{a}^{\mathrm{M}}$ and $\mathbf{b}^{\mathrm{M}}$ are an essentially arbitrary pair of linearly independent vectors. Expanding the term in the exponent, we see that the linear kernel has an especially simple form: $\mathbf{h}(x^{\mathrm{M}}) = -(x^{\mathrm{M}})^2/2\mathbf{a}^{\mathrm{M}} + x^{\mathrm{M}}\mathbf{b}^{\mathrm{M}}$. [Note that the usual prefactor, $\phi(\mathbf{r}^{\mathrm{M}},\mathbf{g}^{\mathrm{M}})$, is related to $\tilde{\phi}(\mathbf{r}^{\mathrm{M}},\mathbf{g}^{\mathrm{M}})$ via $\phi(\mathbf{r}^{\mathrm{M}},\mathbf{g}^{\mathrm{M}}) = \tilde{\phi}(\mathbf{r}^{\mathrm{M}},\mathbf{g}^{\mathrm{M}})\exp(-\mu_{\mathrm{M}}^2/2\sigma_{\mathrm{M}}^2)$.]

Combining the expressions for mean and variance given in Equations 11 and 12 with the fact that, for the independent variables we are considering here, the means and variances add ($\mu_{\mathrm{A}} = \mu_{\mathrm{R}} + \mu_{\mathrm{E}}$ and $\sigma_{\mathrm{A}}^2 = \sigma_{\mathrm{R}}^2 + \sigma_{\mathrm{E}}^2$), we see that

$$\begin{aligned} \frac{\mathbf{b}^{\mathrm{A}} \cdot \mathbf{r}^{\mathrm{A}}}{\mathbf{a}^{\mathrm{A}} \cdot \mathbf{r}^{\mathrm{A}}} &= \frac{\mathbf{b}^{\mathrm{R}} \cdot \mathbf{r}^{\mathrm{R}}}{\mathbf{a}^{\mathrm{R}} \cdot \mathbf{r}^{\mathrm{R}}} + \frac{\mathbf{b}^{\mathrm{E}} \cdot \mathbf{r}^{\mathrm{E}}}{\mathbf{a}^{\mathrm{E}} \cdot \mathbf{r}^{\mathrm{E}}} \\ \frac{1}{\mathbf{a}^{\mathrm{A}} \cdot \mathbf{r}^{\mathrm{A}}} &= \frac{1}{\mathbf{a}^{\mathrm{R}} \cdot \mathbf{r}^{\mathrm{R}}} + \frac{1}{\mathbf{a}^{\mathrm{E}} \cdot \mathbf{r}^{\mathrm{E}}}. \end{aligned} \tag{13}$$

We now simply need to write $\mathbf{r}^{\mathrm{A}}$ as a function of $\mathbf{r}^{\mathrm{R}}$ and $\mathbf{r}^{\mathrm{E}}$ such that Equation 13 is satisfied. Defining two new vectors $\mathbf{a}^{\mathrm{A}\dagger}$ and $\mathbf{b}^{\mathrm{A}\dagger}$ that obey the orthogonality conditions $\mathbf{a}^{\mathrm{A}\dagger} \cdot \mathbf{a}^{\mathrm{A}} = \mathbf{b}^{\mathrm{A}\dagger} \cdot \mathbf{b}^{\mathrm{A}} = 1$; $\mathbf{a}^{\mathrm{A}\dagger} \cdot \mathbf{b}^{\mathrm{A}} = \mathbf{b}^{\mathrm{A}\dagger} \cdot \mathbf{a}^{\mathrm{A}} = 0$, it is easy to show that the weights in Equation 8 are given by

$$w_{ij}^{\mathrm{A}} = a_i^{\mathrm{R}} a_j^{\mathrm{E}} a_k^{\mathrm{A}\dagger} + a_i^{\mathrm{R}} b_j^{\mathrm{E}} b_k^{\mathrm{A}\dagger} + b_i^{\mathrm{R}} a_j^{\mathrm{E}} b_k^{\mathrm{A}\dagger}, \tag{14}$$

and the coefficients in the divisive term, $\mathbf{c}^{\mathrm{R}}$ and $\mathbf{c}^{\mathrm{E}}$, are equal to $\mathbf{a}^{\mathrm{R}}$ and $\mathbf{a}^{\mathrm{E}}$. For the more general case in which there are priors on x^{R} and x^{E}, the weights turn out to be identical to those in Equation 14. The one small difference is that there are two extra terms: a piece that is linear in the firing rates is added to the numerator in Equation 8, and a constant is added to the denominator (as in Eq. 15 below). Thus, for linear transformations and Gaussian noise, the optimal network consists of a quadratic nonlinearity and divisive normalization.

Note that linear coordinate transformations are very different from multisensory integration, which we have considered previously (Ma et al., 2006). For a linear coordinate transformation, means and variances combine linearly; for multisensory integration, means and variances combine nonlinearly. Specifically, if we have two variables, say x^{A} and x^{V}, that provide information about the location of a single object, then the mean and variance of the location are given by $(\mu_{\mathrm{A}}/\sigma_{\mathrm{A}}^2 + \mu_{\mathrm{V}}/\sigma_{\mathrm{V}}^2)/(1/\sigma_{\mathrm{A}}^2 + 1/\sigma_{\mathrm{V}}^2)$ and $1/(1/\sigma_{\mathrm{A}}^2+1/\sigma_{\mathrm{V}}^2)$, respectively (Ma et al., 2006), rather than $\mu_{\mathrm{A}} + \mu_{\mathrm{V}}$ and $\sigma_{\mathrm{A}}^2 + \sigma_{\mathrm{V}}^2$, as in the case we just considered. Not surprisingly, then, the network that performs multisensory integration is very different from the network that performs linear coordinate transformations. For linear PPCs, gains are inversely proportional to variance (Fig. 1), so a simple sum of gains effectively implements the sum of inverse variances, $1/\sigma_{\mathrm{A}}^2 + 1/\sigma_{\mathrm{V}}^2$. Therefore, for multisensory integration, there is no need for a quadratic nonlinearity or divisive normalization; a sum of neural activity suffices as long as the inputs are linear PPCs.

Linear coordinate transformation: simulations

Equations 8 and 14 tell us the optimal network for neurons that are deterministic and analog. Real neurons are neither. To determine whether our optimal network for marginalization also works with real neurons, we performed network simulations using a type of spiking neuron known as LNP, or linear–nonlinear–Poisson (Gerstner and Kistler, 2002). Because the Poisson step in LNP neurons is probabilistic, there will necessarily be some information loss. However, Poisson spike generation is independent across neurons, so the information loss, as measured by the change in Fisher information, is inversely proportional to the number of neurons. Indeed, using only 20 neurons in the input and output layers, and 400 in the intermediate layer, the spiking network loses only 0.72% of the input information (Fig. 1*b*, bar marked "QDN"). Here, information loss is assessed by the Kullback–Leibler distance between the true posterior given by the input pattern of activity and the approximate posterior given by the output pattern of activity in the network. Importantly, in these simulations the contrast of the visual object changed from trial to trial, where contrast plays the role of a nuisance parameter that affects the quality of the representation of x^{R}. Thus, as discussed above, the network was able to perform near-optimally without explicit knowledge of the nuisance parameters.

How important are the two nonlinear features of Equation 8—the quadratic term in the numerator and the divisive normalization in the denominator? To determine the role of divisive normalization, we eliminated it and built a network in which the basis function layer had only quadratic terms. Such a network did not do as well; it lost 16% of the input information (Fig. 2*b*, bar labeled "Q"). This information loss could be due to two factors:

either the network encodes the optimal posterior but this posterior is not encoded as a linear PPC (Eq. 5), or it does not encode the optimal posterior distribution (Eq. 6). We found that it is the former: the correct distribution over the head-centered location of the stimulus can be recovered from activity in the basis function layer even in the absence of divisive normalization, but it cannot be recovered by a single linear decoder. Instead, a set of linear decoders, each specialized for a different contrast, is required. With multiple linear decoders, the information loss is only 1.4% (Fig. 2*b*, bar labeled “Q-SD”). Thus, virtually all the information in the basis function layer is available to a decoder that knows the contrast. Importantly, however, without divisive normalization, the population activity cannot be decoded optimally by a fixed linear decoder. This makes it difficult for downstream networks to make optimal use of the activity unless the contrast is also transmitted or estimated. The network with divisive normalization is immune to this problem.

To determine the role of the quadratic nonlinearity in the basis function layer, we replaced it with a rectified linear activation function, and at the same time eliminated the divisive nonlinearity. Such a network was also suboptimal; it lost 32% of the input information (Fig. 2*b*, bar labeled “L”). In this case, information was truly lost: even with linear decoders specialized for contrast and applied to the basis function layer, the information loss remained high (24%; Fig. 2*b*, bar labeled “L-SD”). These simulations indicate that the divisive nonlinearity is needed to ensure that the output of the network is easily decoded (even by a network that does not know the contrast), and the quadratic nonlinearity is needed to ensure optimality.

Nonlinear coordinate transformation

To determine whether these results generalize to nonlinear coordinate transformations and non-Gaussian probability distributions, we simulated a network that computes the azimuth of the end-point of a two-joint arm given the joint angles. This is a fundamentally nonlinear transformation (see Fig. 2*c*, inset) and, because the variables are periodic, we cannot assume Gaussian distributions for the joint angles, but instead use a Von Mises distribution (which, for the large variance we used, is very different from a Gaussian distribution).

The nonlinearity and non-Gaussian noise means that a network using linear PPCs can no longer perform optimally. However, as we show below, a network with a quadratic nonlinearity and divisive normalization provides a close approximation to the optimal solution as long as the network parameters are properly optimized. In fact, the network we use is almost identical to the one in Equation 8; the only difference is that we add a constant to the denominator. Thus, the output population activity, again denoted $\mathbf{r}^{\mathrm{A}}$, is related to the input populations, $\mathbf{r}^{\mathrm{R}}$ and $\mathbf{r}^{\mathrm{E}}$, via

$$r_k^{\mathrm{A}} = \sum_{ij} \frac{w_{ij}^k r_i^{\mathrm{R}} r_j^{\mathrm{E}}}{\alpha + \sum_l c_{\mathrm{R}l} r_l^{\mathrm{R}} + c_{\mathrm{E}l} r_l^{\mathrm{E}}} . \qquad (15)$$

The network parameters are the weights, the w_{ij}^k, the coefficients in the divisive term, $\mathbf{c}_{\mathrm{R}}$ and $\mathbf{c}_{\mathrm{E}}$, and the additional divisive term, α. These parameters were optimized so that the output layer captures as much information as possible.

As can be seen in Figure 2*c*, the network with a quadratic nonlinearity and divisive normalization (Eq. 15) performs well: the information loss is only 4%. By comparison, the quadratic network without divisive normalization loses 15% of the information (bar in Fig. 2*c* marked “Q”). As with the linear transformation, though, a network without divisive normalization can perform well if it uses a different, specialized, decoder for each contrast (bar marked “Q-SD”). Finally, a linear rectified network loses a large amount of information—about 40%.

Why should a network optimized for linear transformations and Gaussian noise perform so well for a nonlinear transformation and non-Gaussian noise? The answer has to do with the posteriors in the input layer: both are single peaked with widths that are reasonably small compared to the curvature associated with the nonlinearity. Under these conditions, the transformation is locally linear and Gaussian, and so our network is near-optimal. Note, though, that “small” can be fairly large: at low contrast, the width of the posterior was on the order of the curvature in the nonlinearity, and the noise was noticeably non-Gaussian. This suggests that networks implementing a quadratic nonlinearity with divisive normalization are robust with respect to departures from linearity and Gaussianity.

Time-varying quantities

In the above example, we focused on a static case—a transformation from eye-centered to head-centered coordinates of a stationary object relative to a stationary head with stationary eyes. In almost all real-world problems, however, variables change over time. A broad class of such problems involves noisy, time-dependent observations combined with an internal model. Such problems arise in the sensory domain (e.g., visually tracking an object), the motor domain (e.g., moving a limb), navigation (e.g., keeping track of one’s location in space given movements and sensory cues), and cognitive tasks (e.g., decision making, in which evidence is accumulated over time). In all cases the brain receives a continuous stream of noisy information about a time-dependent quantity via the visual system in the case of tracking and via proprioception in the case of limb movement. That information must be combined with an internal model (of the motion of the object or the limb) to yield a posterior distribution over the variable of interest. The marginalization in this case is over all possible trajectories. For example, if you move your hand in the dark, the probability that it will arrive at a particular position is found by enumerating each possible trajectories that ends in that position, and adding up their probabilities.

To understand how the brain might solve this class of problems, we first consider navigation, and ask how a rat could use place cells to keep track of its position on a one-dimensional track. We assume that the rat has an internal model of its position, $s(t)$, and that visual cues provide instantaneous (but noisy) information about that position. The visual cues are encoded in a linear PPC by a population of neurons, denoted $\mathbf{r}^{\mathrm{in}}(t)$, using a time-dependent version of Equation 10,

$$p(\mathbf{r}^{\mathrm{in}}|s,t,g) \propto \exp\left[-\frac{\mathbf{a}^{\mathrm{in}} \cdot \mathbf{r}^{\mathrm{in}}(t)}{2}\left(s - \frac{\mathbf{b}^{\mathrm{in}} \cdot \mathbf{r}^{\mathrm{in}}(t)}{\mathbf{a}^{\mathrm{in}} \cdot \mathbf{r}^{\mathrm{in}}(t)}\right)^2\right] \qquad (16)$$

(see Fig. 3*a*). For this model, as in Equations 11 and 12 above, the mean and variance associated with this likelihood function are given by $\mathbf{b}^{\mathrm{in}} \cdot \mathbf{r}^{\mathrm{in}}(t)/\mathbf{a}^{\mathrm{in}} \cdot \mathbf{r}^{\mathrm{in}}(t)$ and $1/\mathbf{a}^{\mathrm{in}} \cdot \mathbf{r}^{\mathrm{in}}(t)$, respectively. Here $r_i^{\mathrm{in}}(t)$is the number of spikes on neuron i that occurred between times t and $t + dt$ (eventually we will take the limit $dt \rightarrow 0$). Note that in most time intervals, there are no spikes, and so all components of $\mathbf{r}^{\mathrm{in}}(t)$ are zero. For these intervals, the variance is infinite, and the visual cues do not supply any information. Since the input spikes convey information only rarely, and have no memory, all memory must be encoded in the network.

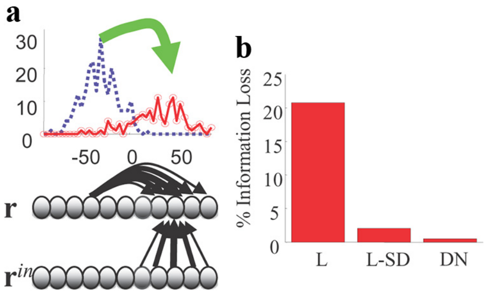

Figure 3. Kalman filter. ***a***, The input layer, $\mathbf{r}^{\text{in}}$, (corresponding to the spike times, $t_j^{k,\text{in}}$, in Eq. 19) provides evidence for the current position of the object in the form of a probabilistic population code. The output layer (corresponding to the spikes, $\mathbf{r}$, generated from the firing rate, $\boldsymbol{\nu}$, which evolves according to Eq. 19) encodes the position of a moving object on a spatial map. The lateral connections in the output layer, some of which are shown as black arrows, implement the internal model (for the optimal model, the dynamics is given in Eq. 19). The resulting pattern of activity consists of a moving hill that encodes, via a linear PPC, for the position of the object. In the top panel, the blue and red curves correspond to the output activity, $\mathbf{r}$, at time t and $t + \Delta t$, respectively. ***b***, Information loss in the output layer relative to the information available in the input layer. The network with effective divisive normalization, via the quadratic term in Equation 19, is near optimal (DN), while a network with linear inhibition ($\mathbf{a}\cdot\boldsymbol{\nu}$ replaced by a constant) decoded with a single decoder (L) performs poorly. However, performance is close to optimal for the linear network when using specialized decoders in which $\mathbf{a}\cdot\boldsymbol{\nu}$ is replaced by a constant that depends on the level of noise (L-SD). See Notes.

To understand how to combine the information from $\mathbf{r}^{\text{in}}$ with the animal's current estimate of its position, we need to specify an internal model. For simplicity, we assume that the animal is attracted toward a point on a one-dimensional track (taken to be the origin), but its movement is corrupted by noise. (Below we consider a two-dimensional version of this task.) The dynamics associated with this model has the form

$$\frac{ds}{dt} = -\gamma s + \eta(t), \qquad (17)$$

where s is position, γ determines how fast the animal is pulled toward the origin, and $\eta(t)$ represents Gaussian white noise.

Between spikes, the animal's estimate of its mean position drifts toward zero (because of the $-\gamma s$ term) and its variance grows (because there is no incoming information). When a spike occurs, the animal gets new information about position in the form of a mean, denoted μ_{in}, and variance, denoted σ_{in}^2 (which, as discussed above, are equal to $\mathbf{b}^{\text{in}}\cdot\mathbf{r}^{\text{in}}/\mathbf{a}^{\text{in}}\cdot\mathbf{r}^{\text{in}}$ and $1/\mathbf{a}^{\text{in}}\cdot\mathbf{r}^{\text{in}}$, respectively). Since new evidence regarding current position is conditionally independent of previous evidence about current position, combining that information with the current estimate is just a cue-integration problem. Thus, if the current mean and variance are μ and σ^2, they should be combined according to Ma et al. (2006), as follows:

$$\mu \rightarrow \frac{\mu/\sigma^2 + \mu_{\text{in}}/\sigma_{\text{in}}^2}{1/\sigma^2 + 1/\sigma_{\text{in}}^2} \qquad (18)$$
$$1/\sigma^2 \rightarrow 1/\sigma^2 + 1/\sigma_{\text{in}}^2.$$

Equation 18 tells us how the mean and variance should be updated when there is evidence. How should the neural activity be updated? We assume, as usual, that position, s, is encoded via a linear PPC, using the same encoding model as in Equation 16 but with $\mathbf{r}^{\text{in}}$ replaced by a deterministic firing rate, denoted $\boldsymbol{\nu}$, and $\mathbf{a}^{\text{in}}$ and $\mathbf{b}^{\text{in}}$ replaced by new vectors, $\mathbf{a}$ and $\mathbf{b}$. Then, as has been shown previously (Ma et al., 2006), the optimal cue combination network combines spikes linearly. Reasonably straightforward algebra (see Notes) indicates that $\boldsymbol{\nu}$ should evolve according to

$$\frac{d\nu_i}{dt} = -\sigma_\eta^2(\mathbf{a}\cdot\boldsymbol{\nu})\nu_i + \gamma\sum_j W_{ij}\nu_j + \sum_j M_{ij}\sum_k \delta(t - t_j^{k,\text{in}}), \qquad (19)$$

where the $t_j^{k,\text{in}}$ are the times at which spikes occur on input neuron j, $\delta(\cdot)$ is the Dirac delta function, σ_η^2 is the variance of the white noise, and the weights, $\mathbf{W}$ and $\mathbf{M}$, depend only on $\mathbf{a}$ and $\mathbf{b}$. The last term in this expression represents the addition of spikes, as is standard for cue integration (Ma et al., 2006). The origin of the first two terms is slightly more obscure. Briefly, the term proportional to $\sigma_\eta^2(\mathbf{a}\cdot\boldsymbol{\nu})$ acts to reduce the activity of the output population, and thus increase the variance of the posterior (as with the input population, the variance of the population coding for s is equal to $1/\mathbf{a}\cdot\boldsymbol{\nu}$). The term proportional to $\gamma\mathbf{W}$, on the other hand, tends to increase activity, and thus decrease variance. This is because the more strongly the animal is attracted toward the origin (the larger γ is), the more the animal knows where it is, and the smaller the variance.

Unlike Equations 8 and 15, Equation 19 does not explicitly contain any divisive normalization. However, because the first term in Equation 19 is quadratic in $\boldsymbol{\nu}$, in the limit that the input firing rate is large (the $t_j^{k,\text{in}}$ are closely spaced), the output firing rate is proportional to $\bar{\nu}_{\text{in}}^{1/2}$ where $\bar{\nu}_{\text{in}}$ is the average input firing rate. Without the quadratic term, the output firing rate is proportional to $\bar{\nu}_{\text{in}}$ in the limit of large input firing rate. Thus, the effect of the quadratic linearity is to divide the output firing rate by $\bar{\nu}_{\text{in}}^{1/2}$ (see Notes). It is, therefore, effectively divisive—even though there are no explicitly divisive terms in Equation 19.

By construction, if we integrate Equation 19 and compute the posterior distribution over s from Equation 16 (but with $\mathbf{r}^{\text{in}}$ replaced by $\boldsymbol{\nu}$ and $\mathbf{a}^{\text{in}}$ and $\mathbf{b}^{\text{in}}$ replaced by $\mathbf{a}$ and $\mathbf{b}$), we will recover the true posterior. However, neurons communicate by spikes, so we need to turn Equation 19 into a spiking network. We do this, as above, using LNP neurons. In addition, when we compute the posterior, we need to approximate the firing rate by counting spikes in small intervals (we use 10 ms). Because of this transformation to a spiking network, some information will be lost. However, even using only 20 input and 200 output neurons, the loss is small, just 1% (see in Fig. 3*b*, DN). By contrast, a network without the quadratic nonlinearity (i.e., a network in which the quadratic term in Equation 19, $(\mathbf{a}\cdot\boldsymbol{\nu})\nu_i$, is replaced with a linear term, $I_F\nu_i$, with I_F independent of the information in the input population) loses 22% of the information ("L" in Fig. 3*b*). When, on the other hand, I_F depends on the input information ("L-SD" in Fig. 3*b*), the information loss is 3%. This indicates that most of the information is preserved without divisive normalization, but it is no longer in a linear PPC format.

Although we have considered a one-dimensional track, this analysis easily extends to a two-dimensional field. To illustrate how our framework applies to two dimensions, however, we consider a different problem: tracking hand position in a linear track based only on proprioceptive feedback and knowledge of the motor command that drives the movement. This problem is two dimensional because subjects must infer both position and velocity.

One advantage of this task is that we can make a direct comparison to the behavioral experiment of Wolpert et al. (1995). In this experiment, subjects were shown a virtual representation of the location of their hand and asked to make a linear movement.

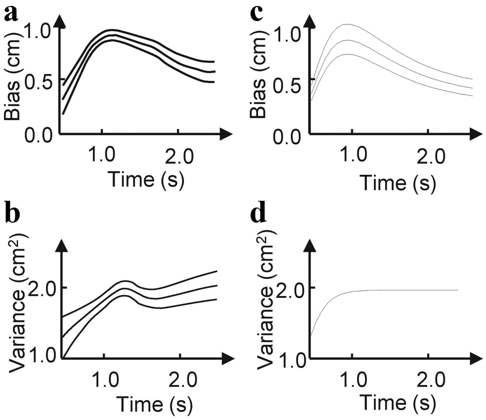

Figure 4. Experimental and simulation results for the hand tracking task of Wolpert et al. (1995). In that task, the hand starts at a known location, accelerates along a linear track until the subject hears a tone, and then decelerates until it stops, at which point subjects are asked to estimate their hand position. ***a***, Experimental bias of the estimate of hand position as a function of duration of movement. Following Wolpert et al. (1995), bias was induced by using the wrong internal model; see Notes. ***b***, Experimental variance of hand position estimate as a function of duration of movement. ***a*** and ***b*** are adapted from Wolpert et al. (1995). ***c***, ***d***, Same as in ***a*** and ***b*** but for the model with divisive normalization.

Upon initiation of the movement the visual representation of the hand disappeared, leaving proprioception as the only source of sensory information about arm position. After a variable delay (0–2 s), subjects ended their movement, and were asked to estimate the position of their (now stationary) hand.

As shown by Wolpert et al. (1995), the mean and variance of the subjects' estimate of endpoint position was consistent with the prediction of a 2-D Kalman filter estimating the velocity and position of the hand. We implemented this Kalman filter with LNP neurons, using a 2-D extension of Equation 19 above, and found that our network does indeed do a good job reproducing the observed pattern of mean and variance (Fig. 4). Thus, our network provides a neural solution for a 2-D Kalman filter that is consistent with these behavioral results.

Recently, Boerlin and Denève (2011) proposed a similar network for computing the posterior distribution in a related task that turns comparable rate equations into spikes via an integrate-and-fire-like mechanism rather via a Poisson process. Although not provably optimal in the limit of large networks, it had the advantage that it represented the posterior distribution with a relatively small number of spikes.

Discrete variables

So far we have considered continuous variables, but marginalization over discrete variables is also a common, and important, inference problem—consider, for example, the problem of detecting the smell of bacon. Here the problem is to infer the probability of a hidden cause (e.g., bacon) given a set of noisy observations of the various volatile chemicals (odorants) that make up both bacon and the other odor sources (henceforth referred to simply as odors) in a given olfactory scene.

At first glance, marginalization over discrete and continuous variables seems very different—if nothing else, the former involves sums and the latter integrals. However, if we use the same encoding model (linear PPCs) for discrete variables, then much of the machinery we used for continuous variables turns out to apply directly. This can be seen with olfaction, for which the problem is to build a network that can encode the marginal probabilities over each possible odor source (e.g., ham, turkey, bacon, etc.) given a set of observed odorants (encoded in the input layer). Here the stimulus of interest, $\mathbf{s}$, is a binary vector indicating the presence or absence of an odor ($s_k = 1$ when odor k is present and 0 otherwise), while c_k is intensity or concentration of odor k. Each odor is described by a specific pattern of odorants; for odor k, we use w_{ik} to denote that pattern. The complete olfactory scene is given by the mixture of the patterns of odorants associated with each odor, weighted by its concentration,

$$o_i = \sum_k w_{ik} c_k s_k. \tag{20}$$

The concentrations of each of the odorants are encoded via a linear probabilistic population code (Eq. 4 with s replaced by o_i and no nuisance parameters),

$$p(\mathbf{r}_i|o_i) = \phi(\mathbf{r}_i)\exp(\mathbf{h}_i(o_i)\cdot\mathbf{r}_i). \tag{21}$$

Here $\mathbf{r}_i$ is the activity of the neurons coding for the concentration of odorant i.

As with the continuous case, we seek a transformation from the $\mathbf{r}_i$ to a new population activity, $\mathbf{r}_k^{\mathrm{out}} = \mathbf{f}_k(\mathbf{r}_1, \mathbf{r}_2, \ldots)$, which codes, via a linear probabilistic population code, for the probability that odor k is present. In general, computing this probability is hard because we have to marginalize out all the other odors—something that involves integrating over the high dimensional space of concentrations and summing over the exponentially many possible combinations of odors. Not surprisingly, deriving the exact transformation with a linear PPCs is difficult, if not impossible. Motivated by our work with continuous variables, we asked whether an approximate solution could be implemented by networks with quadratic nonlinearities and divisive normalization. In fact, we use the same network as in Equation 15, except that there are additional indices to label the different odors,

$$\mathbf{r}_k^{\mathrm{out}} = \sum_{ij} \frac{\mathbf{r}_i \cdot \mathbf{w}_{ij}^k \cdot \mathbf{r}_j}{\alpha + \sum_l \theta_l^k \cdot \mathbf{r}_l}. \tag{22}$$

(see Fig. 5*a*). Note that this network is not meant to provide a detailed model of the olfactory system; instead, it illustrates a hard inference problem that captures the essence, if not the details, of the problem faced by the olfactory system.

We tested this network on four odorants ($i = 1, 2, 3, 4$) and four odors ($k = 1, 2, 3, 4$); the results are shown in Figure 5*b*. We found that for the network with a quadratic nonlinearity and divisive normalization, the information loss was only 6% (bar marked "QDN" in Fig. 5*b*). This should be compared to a 37% loss when we used only a quadratic nonlinearity but no divisive normalization ("Q" in Fig. 5*b*). Using specialized decoders improved that by only a few percent ("Q-SD" in Fig. 5*b*). Linear-threshold networks did even worse; the information loss was over 80% ("L" in Fig. 5*b*). Therefore, networks using quadratic divisive normalization can implement near optimal marginalization even for discrete classes. This extends the result reported in Figure 2*c*, and shows that divisive normalization can lead to near optimal inference even when the computations are nonlinear and the distributions are not Gaussian. Figure 5*b* also shows that, in contrast to our previous results, the performance of the network with quadratic units using specialized decoders (Q-SD, Fig. 5*b*) did not match the performance with quadratic divisive normalization; it lost about 35% of the information.

The basic structure of this task—hidden causes mixing linearly to produce observations—applies to a wide range of prob-

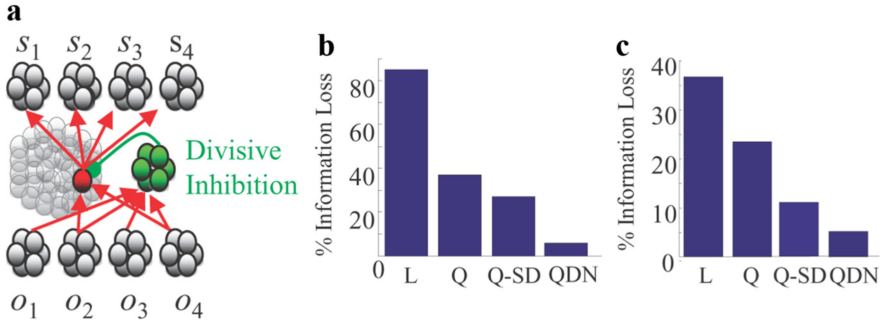

Figure 5. Marginalization for discrete classes. ***a***, Neural recognition network. For each observation ($o_1, o_2, \ldots$), the probability (of an odor source being present in the case of odor detection, or of a hypothesis in the case of blickets) is encoded in the output layer with a linear probabilistic population code. The intermediate units compute all possible products of the input units (e.g., the unit in red computes the product of units encoding o_2 and o_4; this is the quadratic step) and receive divisive normalization from a set of inhibitory neurons (green units) that compute the sum of the activity of the input units. The activities of the intermediate units are then combined linearly in the next layer to encode the marginal distributions over hypotheses. ***b***, Simulations of an olfaction task, in which the network contains four groups of input and output units. The network with quadratic divisive normalization (QDN) is very close to optimal, with only 6% information loss compared to the optimal solution for this task. Linear rectified (L) and quadratic (Q) networks perform much more poorly on the task. A quadratic network with specialized decoders (Q-SD) performs better than the quadratic network with a single decoder, but, in contrast to the previous cases, using specialized decoders does not lead to a performance comparable to the one obtained with a quadratic nonlinearity. Therefore, the quadratic network fails in two ways: it does not compute the optimal posterior distribution, and it does not encode its estimate of the posterior distribution in the form of a probabilistic population code. See Notes. ***c***, Same as in ***b***, but for the blicket experiment, in which the network contains two groups of input units and two groups of output units rather than the four shown in ***a***.

lems. Next we consider one that seems very different from olfaction: causal inference by children, as studied in the so called “blicket” experiment (Gopnik and Sobel, 2000; Griffiths and Tenenbaum, 2009). In this experiment, 4-year-old children are presented with a “blicket” detector, a machine that goes off whenever it is in the proximity of a blicket. Two objects, A and B, are placed on the detector at the same time and the detector goes off, and children are asked to evaluate whether objects A and B are blickets. As one might expect, most children say that both A and B are blickets. Then, object B alone is placed on the detector, which again goes off. After this second presentation, all children say that B is a blicket, and most say that A is not a blicket. This behavior makes perfect sense from a probabilistic point of view. The observation that the detector went off for B alone indicates that B is a blicket, in which case there is no need to posit that A is a blicket to explain why the detector went off when both A and B were presented together (A has been “explained away”). It requires marginalization because the children need to compute the individual (marginal) probabilities that A and B are blickets given knowledge that the detector went off when A and B were presented simultaneously, and that it went off again when B was presented alone. In the operant conditioning literature in rats, this is known as backward blocking (Shanks, 1985; Dayan and Kakade, 2000).

As with olfaction, we use the network shown in Figure 5*a* (but with two inputs and two outputs rather than four of each). Each node in the input layer encodes an observation (in this case the state of the blicket detector and which objects were placed on it); each node in the output layer encodes a hypothesis (in this case the probability that a particular object is a blicket). The observation units connect to an intermediate layer of units implementing, as with olfaction, a quadratic divisive normalization, which in turn projects to the hypothesis units. Simulations of the blicket experiment show that the network loses 8% of the information (Fig. 5*c*). As with all problems considered so far, networks with linear rectified units or quadratic units but no divisive normalization performed much worse (Fig. 5*c*).

Experimental predictions

Our framework makes a variety of experimental predictions. In the case of linear coordinate transformations, the most salient—and somewhat counterintuitive—prediction is that the firing rate at which one population of neurons saturates should depend on the reliability of the information in another population. This could be tested by, for example, recording from neurons in visual cortex that both fire in response to reaching targets and are modulated by arm position. Such neurons have been reported for instance in area V6a (Battaglia-Mayer et al., 2001); these neurons are believed to be involved in a coordinate transformation from visual to motor coordinates for the purpose of reaching. This transformation is similar to the one from eye-centered to head-centered coordinates that we considered earlier, and could be implemented with the architecture illustrated in Figure 2*a*. Our theory predicts that the firing rates of some of those neurons should be modulated by the reliability of the information provided to the sensory system regarding both the target and the arm position. Here reliability is defined as the inverse variance of the posterior distribution (see Fig. 1), which can be modified by manipulating the degree of blur. Specifically, the firing rates of these neurons, λ, should follow the relationship $\lambda \propto g_V g_{AP}/(g_V + g_{AP})$, where g_V and g_{AP} are proportional to the reliability of the evidence regarding visual target and arm position, respectively.

This prediction is specific to the combination of visual target and arm position signal for the purposes of inferring the head-centered location of the target. For instance, if the same neurons also respond to the auditory location of the reaching target (which is presumably already in head-centered coordinates), we predict that the visual and auditory signals should not be multiplied, but added. This is because the probabilistic inference required to combine the visual and auditory input optimally is multisensory integration, not marginalization. As we have shown previously, optimal multisensory integration with linear PPCs requires that we sum the visual and auditory activity (as opposed to taking the product as we did for the visual and arm position signals) (Ma et al., 2006).

The network implementing the Kalman filter, which we applied to place cells in the hippocampus and to tracking hand position based on proprioceptive feedback, not only predicts neural activity in the presence of sensory feedback, it also predicts activity in its absence. For place cells, sensory feedback can be eliminated by turning off the lights and eliminating all olfactory cues; in this case, our theory predicts that the firing rates of the place cells should decrease as a power law, $1/(c + t)$, where t is the time since the sensory cues vanished, assuming the correlations between place cells do not change significantly over time and the restoring force, γ in Equation 17, is zero. The inverse time dependence reflects two facts: first, in the absence of feedback, one’s position should execute a random walk, for which the variance increases linearly with time; second, in the linear PPC framework, firing rate is inversely proportional to variance. This is a very specific, and easily falsified, prediction, one that is made by no other theory we know of.

Finally, our work makes a very specific prediction about what happens when divisive normalization is removed from networks in which variables are encoded as linear PPCs, at least for the inference problems we considered: before removal, the activity should be optimally decodable with a single linear filter; after removal, the optimal linear filter should depend on reliability. This could be tested in insect olfaction, where preliminary results indicate an encoding model consistent with linear PPCs (Olsen and Wilson, 2008), and where it might be possible to selectivity block divisive normalization.

Discussion

Lateral inhibition is ubiquitous in the sensory systems of all animals, and is often thought to enhance stimulus selectivity [which, interestingly, is not always the case, see (Spiridon and Gerstner, 2001; Seriès et al., 2004)]. When lateral inhibition takes a specific form, divisive normalization, it has been suggested that it implements a form of gain control that keeps neurons within their preferred firing range (Heeger, 1992; Nelson et al., 1992; Gao and Vasconcelos, 2009). This form of gain control is also thought to promote contrast-invariant sensory representations in area V1 (Albrecht and Hamilton, 1982; Heeger, 1992; Busse et al., 2009; Ringach, 2010), spatial pattern-invariant velocity representation in area MT (Heeger et al., 1996; Simoncelli and Heeger, 1998), and concentration-invariant odorant representations in the olfactory system (Simoncelli and Heeger, 1998; Luo et al., 2010; Olsen et al., 2010). In addition, it has recently been implicated in attentional modulation (Simoncelli and Heeger, 1998; Winkowski and Knudsen, 2008; Reynolds and Heeger, 2009), and also in probabilistic computations, such as removing high-order correlations in neural responses to natural images (Schwartz and Simoncelli, 2001) and extracting the maximum likelihood estimate from a linear PPC (Deneve et al., 1999). Some of these studies have also explored how invariance promotes linear separability (Luo et al., 2010; Olsen et al., 2010).

Our results share some similarities with the previous studies in the sense that we also suggest that divisive normalization plays a role in probabilistic computations and yields representations that can be linearly decoded in a way that is invariant to nuisance parameters (like contrast or concentration). However, we have expanded these previous studies in two important directions. First, we have shown that quadratic nonlinearities with divisive normalization could play an important role in marginalization, a key operation for probabilistic inference. Second, the resulting representation does not encode just the value of the encoded variable, but encodes full probability distributions, whose log can be linearly decoded with a decoder invariant to nuisance parameters—a form of invariance that goes beyond the tuning curve invariance of previous studies. These results could explain why quadratic divisive normalization has been reported not only in early sensory areas but throughout the nervous system of mammals and insects, since marginalization is a form of probabilistic computation central to a remarkably wide range of seemingly unrelated tasks, including motor control, cognitive reasoning, decision making, navigation, and low-level perceptual learning.

What makes our results appealing from a biological point of view is that both divisive normalization and quadratic nonlinearities are commonly observed in neural circuits. Divisive normalization (the denominator in Eqs. 8, 15, and 22) has been reported in the primary visual cortex (Heeger, 1992; Carandini et al., 1997; Tolhurst and Heeger, 1997), the extrastriate visual cortex (Miller et al., 1993; Rolls and Tovee, 1995; Missal et al., 1997; Recanzone et al., 1997; Treue et al., 2000; Heuer and Britten, 2002; Zoccolan et al., 2005), the superior colliculus (Basso and Wurtz, 1997), and the antenna lobe of the *Drosophila* (Olsen et al., 2010). The circuit implementation of this normalization is still being debated, but several possibilities have been explored (Heeger, 1992; Nelson, 1994; Chance et al., 2002). With regard to the quadratic nonlinearity, the numerator in Equations 8, 15, and 22, and the quadratic term in Equation 19, require a specific form of quadratic interaction in which firing rates are combined either multiplicatively or via the action of a quadratic nonlinearity. Such multiplicative interactions between sensory evoked signals and posture signals have been reported in multiple locations, including V1(Trotter et al., 1996; Trotter and Celebrini, 1999), V3 (Galletti and Battaglini, 1989), MT (Bremmer et al., 1997), MST (Bremmer et al., 1997; Ben Hamed et al., 2003), LIP (Andersen et al., 1985; Stricanne et al., 1996), VIP (Avillac et al., 2005), V6a(Battaglia-Mayer et al., 2001), premotor cortex (Boussaoud et al., 1993), area 5 (Buneo et al., 2002), the primary auditory cortex (Werner-Reiss et al., 2003), and the inferior colliculus (Groh et al., 2001).

As we have seen, marginalization through the use of divisive normalization is near optimal for a linear PPC, a population coding scheme that is associated with constant Fano factors and contrast-invariant tuning curves. Both are consistent with experimental data (Sclar and Freeman, 1982; Tolhurst et al., 1983; Gur et al., 1997; Buracas et al., 1998; Gershon et al., 1998; Anderson et al., 2000; Maimon and Assad, 2009), so spike trains in cortex appear to be consistent with the requirements of our theory. Nevertheless, it is important to develop further tests to confirm whether the variability in cortex, and in other neural circuits, is indeed close to a linear PPC.

Finally, although here we have assumed that variables are encoded in linear PPCs, our approach can be extended to other forms of neural variability, such as tuning curves that are not contrast invariant, or variability that differs strongly from Equation 4. It will be particularly interesting to identify neural systems with variability that deviates strongly from linear PPCs, to see whether circuit nonlinearities take the appropriate form to implement near-optimal probabilistic inference, or perhaps transform these nonlinear PPCs into linear ones.

Notes

Supplemental material for this article is available at http://www.gatsby.ucl.ac.uk/beck_latham_pouget_SI.pdf. It contains all the derivations for the results presented here, along with the details of the implementation. This material has not been peer reviewed.

References

Albrecht DG, Hamilton DB (1982) Striate cortex of monkey and cat: contrast response function. J Neurophysiol 48:217–237.

Albright TD (1992) Form-cue invariant motion processing in primate visual cortex. Science 255:1141–1143.

Andersen RA, Essick GK, Siegel RM (1985) Encoding of spatial location by posterior parietal neurons. Science 230:456–458.

Anderson JS, Lampl I, Gillespie DC, Ferster D (2000) The contribution of noise to contrast invariance of orientation tuning in cat visual cortex. Science 290:1968–1972.

Avillac M, Denève S, Olivier E, Pouget A, Duhamel JR (2005) Reference frames for representing the location of visual and tactile stimuli in the parietal cortex. Nat Neurosci 8:941–949.

Baker CL, Saxe R, Tenenbaum JB (2009) Action understanding as inverse planning. Cognition 113:329–349.

Basso MA, Wurtz RH (1997) Modulation of neuronal activity by target uncertainty. Nature 389:66–69.

Battaglia-Mayer A, Ferraina S, Genovesio A, Marconi B, Squatrito S, Molinari

M, Lacquaniti F, Caminiti R (2001) Eye-hand coordination during reaching. II. An analysis of the relationships between visuomanual signals in parietal cortex and parieto-frontal association projections. Cereb Cortex 11:528–544.

Ben Hamed S, Page W, Duffy C, Pouget A (2003) MSTd neuronal basis functions for the population encoding of heading direction. J Neurophysiol 90:549–558.

Blaisdell AP, Sawa K, Leising KJ, Waldmann MR (2006) Causal reasoning in rats. Science 311:1020–1022.

Boerlin M, Denève S (2011) Spike-based population coding and working memory. PLoS Comput Biol 7:e1001080.

Boussaoud D, Barth TM, Wise SP (1993) Effects of gaze on apparent visual responses of frontal cortex neurons. Exp Brain Res 93:423–434.

Bremmer F, Ilg UJ, Thiele A, Distler C, Hoffman KP (1997) Eye position effects in monkey cortex. I: Visual and pursuit-related activity in extrastriate areas MT and MST. J Neurophysiol 77:944–961.

Buneo CA, Jarvis MR, Batista AP, Andersen RA (2002) Direct visuomotor transformations for reaching. Nature 416:632–636.

Buracas GT, Zador AM, DeWeese MR, Albright TD (1998) Efficient discrimination of temporal patterns by motion-sensitive neurons in primate visual cortex. Neuron 20:959–969.

Busse L, Wade AR, Carandini M (2009) Representation of concurrent stimuli by population activity in visual cortex. Neuron 64:931–942.

Carandini M, Heeger DJ, Movshon JA (1997) Linearity and normalization in simple cells of the macaque primary visual cortex. J Neurosci 17:8621–8644.

Chance FS, Abbott LF, Reyes AD (2002) Gain modulation from background synaptic input. Neuron 35:773–782.

Cordes S, Gallistel CR, Gelman R, Latham P (2007) Nonverbal arithmetic in humans: light from noise. Percept Psychophys 69:1185–1203.

Dayan P, Kakade S (2000) Explaining away in weight space. In: NIPS, pp 451–457. Vancouver: MIT Press.

Deneve S, Latham PE, Pouget A (1999) Reading population codes: A neural implementation of ideal observers. Nat Neurosci 2:740–745.

Galletti C, Battaglini PP (1989) Gaze-dependent visual neurons in area V3A of monkey prestriate cortex. J Neurosci 9:1112–1125.

Gao D, Vasconcelos N (2009) Decision-theoretic saliency: computational principles, biological plausibility, and implications for neurophysiology and psychophysics. Neural Comput 21:239–271.

Gershon ED, Wiener MC, Latham PE, Richmond BJ (1998) Coding strategies in monkey V1 and inferior temporal cortices. J Neurophysiol 79:1135–1144.

Gerstner W, Kistler WM (2002) Spiking neuron models. Cambridge: Cambridge UP.

Gopnik A, Sobel DM (2000) Detecting blickets: how young children use information about novel causal powers in categorization and induction. Child Dev 71:1205–1222.

Griffiths TL, Tenenbaum JB (2009) Theory-based causal induction. Psychol Rev 116:661–716.

Groh JM, Trause AS, Underhill AM, Clark KR, Inati S (2001) Eye position influences auditory responses in primate inferior colliculus. Neuron 29:509–518.

Gur M, Beylin A, Snodderly DM (1997) Response variability of neurons in primary visual cortex (V1) of alert monkeys. J Neurosci 17:2914–2920.

Heeger DJ (1992) Normalization of cell responses in cat striate cortex. Vis Neurosci 9:181–197.

Heeger DJ, Simoncelli EP, Movshon JA (1996) Computational models of cortical visual processing. Proc Natl Acad Sci U S A 93:623–627.

Heuer HW, Britten KH (2002) Contrast dependence of response normalization in area MT of the rhesus macaque. J Neurophysiol 88:3398–3408.

Körding KP, Beierholm U, Ma WJ, Quartz S, Tenenbaum JB, Shams L (2007) Causal inference in multisensory perception. PLoS ONE 2:e943.

Luo SX, Axel R, Abbott LF (2010) Generating sparse and selective third-order responses in the olfactory system of the fly. Proc Natl Acad Sci U S A 107:10713–10718.

Ma WJ, Beck JM, Latham PE, Pouget A (2006) Bayesian inference with probabilistic population codes. Nat Neurosci 9:1432–1438.

Maimon G, Assad JA (2009) Beyond Poisson: increased spike-time regularity across primate parietal cortex. Neuron 62:426–440.

Mazer JA, Vinje WE, McDermott J, Schiller PH, Gallant JL (2002) Spatial frequency and orientation tuning dynamics in area V1. Proc Natl Acad Sci U S A 99:1645–1650.

Miller EK, Gochin PM, Gross CG (1993) Suppression of visual responses of neurons in inferior temporal cortex of the awake macaque by addition of a second stimulus. Brain Res 616:25–29.

Missal M, Vogels R, Orban GA (1997) Responses of macaque inferior temporal neurons to overlapping shapes. Cereb Cortex 7:758–767.

Nelson JI, Salin PA, Munk MH, Arzi M, Bullier J (1992) Spatial and temporal coherence in cortico-cortical connections: a cross-correlation study in areas 17 and 18 in the cat. Vis Neurosci 9:21–37.

Nelson ME (1994) A mechanism for neuronal gain control by descending pathways. Neural Comput 6:242–254.

Olsen SR, Wilson RI (2008) Lateral presynaptic inhibition mediates gain control in an olfactory circuit. Nature 452:956–960.

Olsen SR, Bhandawat V, Wilson RI (2010) Divisive normalization in olfactory population codes. Neuron 66:287–299.

Recanzone GH, Wurtz RH, Schwarz U (1997) Responses of MT and MST neurons to one and two moving objects in the receptive field. J Neurophysiol 78:2904–2915.

Reynolds JH, Heeger DJ (2009) The normalization model of attention. Neuron 61:168–185.

Ringach DL (2010) Population coding under normalization. Vision Res 50:2223–2232.

Rolls ET, Tovee MJ (1995) The responses of single neurons in the temporal visual cortical areas of the macaque when more than one stimulus is present in the receptive field. Exp Brain Res 103:409–420.

Schwartz O, Simoncelli EP (2001) Natural signal statistics and sensory gain control. Nat Neurosci 4:819–825.

Sclar G, Freeman RD (1982) Orientation selectivity in the cat's striate cortex is invariant with stimulus contrast. Exp Brain Res 46:457–461.

Seriès P, Latham PE, Pouget A (2004) Tuning curve sharpening for orientation selectivity: coding efficiency and the impact of correlations. Nat Neurosci 7:1129–1135.

Shanks DR (1985) Forward and backward blocking in human contingency judgment. Q J Exp Psychol 37B:1–21.

Simoncelli EP, Heeger DJ (1998) A model of neuronal responses in visual area MT. Vision Res 38:743–761.

Spiridon M, Gerstner W (2001) Effect of lateral connections on the accuracy of the population code for a network of spiking neurons. Network 12:409–421.

Stricanne B, Andersen RA, Mazzoni P (1996) Eye-centered, head-centered, and intermediate coding of remembered sound locations in area LIP. J Neurophysiol 76:2071–2076.

Tolhurst DJ, Heeger DJ (1997) Comparison of contrast-normalization and threshold models of the responses of simple cells in cat striate cortex. Vis Neurosci 14:293–309.

Tolhurst DJ, Movshon JA, Dean AF (1983) The statistical reliability of signals in single neurons in cat and monkey visual cortex. Vision Res 23:775–785.

Treue S, Hol K, Rauber HJ (2000) Seeing multiple directions of motion-physiology and psychophysics. Nat Neurosci 3:270–276.

Trotter Y, Celebrini S (1999) Gaze direction controls response gain in primary visual-cortex neurons. Nature 398:239–242.

Trotter Y, Celebrini S, Stricanne B, Thorpe S, Imbert M (1996) Neural processing of stereopsis as a function of viewing distance in primate visual area V1. J Neurophysiol 76:2872–2885.

Werner-Reiss U, Kelly KA, Trause AS, Underhill AM, Groh JM (2003) Eye position affects activity in primary auditory cortex of primates. Curr Biol 13:554–562.

Winkowski DE, Knudsen EI (2008) Distinct mechanisms for top-down control of neural gain and sensitivity in the owl optic tectum. Neuron 60:698–708.

Wolpert DM, Ghahramani Z, Jordan MI (1995) An internal model for sensorimotor integration. Science 269:1880–1882.

Zemel RS, Dayan P (1997) Combining probabilistic population codes. In: JCAI-97: Fifteenth International Joint Conference on Artificial Intelligence, pp 1114–1119. San Francisco: Morgan Kaufmann.

Zoccolan D, Cox DD, DiCarlo JJ (2005) Multiple object response normalization in monkey inferotemporal cortex. J Neurosci 25:8150–8164.