

Solomon, J. A. Connecting psychophysical performance to neuronal response properties I: Discrimination of suprathreshold stimuli. *Journal of Vision*, 15(6), p. 8. doi: 10.1167/15.6.8



**CITY UNIVERSITY
LONDON**

[City Research Online](#)

Original citation: Solomon, J. A. Connecting psychophysical performance to neuronal response properties I: Discrimination of suprathreshold stimuli. *Journal of Vision*, 15(6), p. 8. doi: 10.1167/15.6.8

Permanent City Research Online URL: <http://openaccess.city.ac.uk/6532/>

Copyright & reuse

City University London has developed City Research Online so that its users may access the research outputs of City University London's staff. Copyright © and Moral Rights for this paper are retained by the individual author(s) and/ or other copyright holders. All material in City Research Online is checked for eligibility for copyright before being made available in the live archive. URLs from City Research Online may be freely distributed and linked to from other web pages.

Versions of research

The version in City Research Online may differ from the final published version. Users are advised to check the Permanent City Research Online URL above for the status of the paper.

Enquiries

If you have any enquiries about any aspect of City Research Online, or if you wish to make contact with the author(s) of this paper, please email the team at publications@city.ac.uk.

Connecting psychophysical performance to neuronal response properties I: Discrimination of suprathreshold stimuli

Keith A. May

Centre for Applied Vision Research,
City University London, London, UK



Joshua A. Solomon

Centre for Applied Vision Research,
City University London, London, UK



One of the major goals of sensory neuroscience is to understand how an organism's perceptual abilities relate to the underlying physiology. To this end, we derived equations to estimate the best possible psychophysical discrimination performance, given the properties of the neurons carrying the sensory code. We set up a generic sensory coding model with neurons characterized by their tuning function to the stimulus and the random process that generates spikes. The tuning function was a Gaussian function or a sigmoid (Naka-Rushton) function. Spikes were generated using Poisson spiking processes whose rates were modulated by a multiplicative, gamma-distributed gain signal that was shared between neurons. This doubly stochastic process generates realistic levels of neuronal variability and a realistic correlation structure within the population. Using Fisher information as a close approximation of the model's decoding precision, we derived equations to predict the model's discrimination performance from the neuronal parameters. We then verified the accuracy of our equations using Monte Carlo simulations. Our work has two major benefits. Firstly, we can quickly calculate the performance of physiologically plausible population coding models by evaluating simple equations, which makes it easy to fit the model to psychophysical data. Secondly, the equations revealed some remarkably straightforward relationships between psychophysical discrimination performance and the parameters of the neuronal population, and give deep insights into the relationships between an organism's perceptual abilities and the properties of the neurons on which those abilities depend.

Introduction

The key motivation behind this work is to facilitate the construction of physiologically plausible models of psychophysical performance that are mathematically tractable. Pioneering work in the 1960s and 70s led to a standard type of model of early vision consisting of a bank of independent channels selective for different stimulus feature values (Graham, 1989). Within each channel, the signal strength increased (often nonlinearly) with stimulus contrast. Performance was limited by the addition of (usually Gaussian) noise somewhere in the model. This kind of model has the advantage of mathematical simplicity: The relationships between the model's parameters and its psychophysical performance can be described by simple mathematical expressions, allowing us (1) to easily fit the model to data, and (2) to understand why the model behaves as it does. Unfortunately, it has become increasingly clear that the connection between real neurons and the psychophysical channels in this kind of model is tenuous at best (Goris, Putzeys, Wagemans, & Wichmann, 2013). Recent modelling work has shown that more physiologically plausible population-coding models of early vision can provide a unified account of many diverse psychophysical phenomena, while at the same time providing a stronger connection with physiology than has traditionally been the case (Goris et al., 2013). However, the complexity of this kind of model can sometimes obscure the relationships between properties of the model and characteristics of performance, making it difficult to understand the model's behaviour. In this paper, we describe sensory coding models that have a close connection to physiology but are nevertheless simple enough to be described by equations that reveal straightforward relationships between the neuronal parameters and psychophysical performance.

We assume a generic sensory coding model in which the observer monitors a set of spiking neurons and makes a maximum-likelihood estimate of the stimulus from the set of spike counts generated by those neurons. Thus, we find the best possible decoding performance that the encoding scheme allows. The reciprocal of the variance of the estimated stimulus values is called the *precision*. The precision determines the expected performance on perceptual tasks, and so, if we can estimate the precision from the neuronal parameters, then we can estimate task performance.

For an unbiased maximum-likelihood decoder, the precision cannot exceed a quantity called the *Fisher information*. This limit is known as the Cramér-Rao bound (Cramér, 1946; Rao, 1945; see Dayan & Abbott, 2001, pp. 120-121). For a reasonable spike rate or number of neurons, the precision of the generic sensory coding model that we describe in this paper is actually very close to the Cramér-Rao bound, so we can use the Fisher information as an approximation of the decoding precision. The Fisher information is calculated from the properties of the neurons, and therefore forms a bridge between the neuronal properties and perceptual performance.

For each model parameterization that we studied, we set up a Monte Carlo simulation which showed that the model's performance is very close to that predicted from the Fisher information. We used two measures of model performance: One was the precision, defined earlier, and the other was the discrimination threshold in a simulated 2-alternative forced-choice (2AFC) discrimination experiment. Both were very well predicted from the Fisher information.

Our work has two main benefits. One is a practical one: We can quickly calculate the performance of realistic population coding models by evaluating simple equations, rather than using slow and laborious Monte Carlo simulations, which can be difficult and error-prone to program. This allows us to quickly fit the model to data. The second benefit is that the equations give a deep insight into the relationships between psychophysical performance and the properties of the neurons that carry the sensory code, showing us why these relationships occur, and how generally they apply.

Our equations give the upper limit on the possible psychophysical performance level, given the neuronal parameters. There is plenty of evidence that human perceptual performance falls far short of the maximum level that is theoretically possible (Banks, Geisler, & Bennett, 1987; Geisler, 1989; Pelli, 1990; Pelli & Farell, 1999; Dakin, 2001; Pelli, Burns, Farell, & Moore-Page, 2006; Morgan, Chubb, & Solomon, 2008; Solomon, 2010). Nevertheless, there are several reasons why it is useful to know this theoretical limit, and we outline these reasons in remainder of this Introduction.

Firstly, knowing the theoretical maximum performance allows us to compare different encoding schemes. By "encoding scheme", we mean the population of neurons that are used to encode the stimulus. It is of great interest to know why neurons have the properties that they do, and a lot of progress has been made by showing that observed physiological properties reflect encoding strategies that are optimal in some way (Laughlin, 1981; Srinivasan, Laughlin, & Dubs, 1982; Atick & Redlich, 1990, 1992; Atick, 1992; Atick, Li, & Redlich, 1992, 1993; Tadmor & Tolhurst, 2000). Given a particular encoding scheme, the psychophysical performance will depend on how appropriate the decoding algorithm is for that particular encoding scheme (Beck, Ma, Pitkow, Latham, & Pouget, 2012). A fair way to compare two encoding schemes is to compare the maximum performance that each encoding scheme allows.

A second reason for wanting to know the maximum performance level that a set of neurons can support is that this knowledge allows us to perform *sequential ideal observer analysis*. This approach, developed by Geisler (1989), calculates the efficiency with which the information at each stage is processed. Roughly speaking, efficiency is the proportion of the available information that the observer appears to use. If we have a sufficiently good model of processing up to a particular point in the processing stream, we can construct an ideal observer for processing the information known to exist at that point, and then compare the ideal observer's performance against that of a real observer. An efficiency of 1 would mean that the real observer's performance matched that of the ideal observer, so no further information was lost beyond that point in the processing stream. We can proceed like this from very early stages, e.g. the optics of the eye, through to later stages, seeing at each stage what proportion of the available information is lost by later stages of processing. To perform this kind of analysis, we need to be able to calculate the best possible performance allowed by the information at each stage.

Sometimes, it is not necessary to calculate the absolute efficiency. A third reason why it is useful to know the maximum performance theoretically possible is that it can explain patterns of performance. If we assume that, after a certain point in the processing stream, efficiency is constant across different stimulus values, then we can predict the detection or discrimination threshold up to a multiplicative factor. This kind of approach has been used to explain observed patterns of psychophysical performance, such as the shapes of the contrast sensitivity function and wavelength discrimination function (Banks et al., 1987; Geisler, 1989; Zhaoping, Geisler, & May, 2011). If we know the *proportions* of neurons of different types at a particular point in the processing stream, we can calculate the decoding precision up to a multiplicative factor; as-

suming constant efficiency of processing beyond that point, this gives us the expected variance of the decoded stimulus values (and hence discrimination thresholds) up to a multiplicative factor. As we will show, the neuronal parameters (or functions of the neuronal parameters) have multiplicative effects on the decoding precision, and thus will have the same effect on the ideal decoder's performance as they will on the performance of the decoder with constant inefficiency.

Many of the derivations and technical details are included in appendices in the supplementary information. Each supplementary appendix is labelled with a letter. Equations and figures in an appendix are labelled with the appendix's letter, followed by a dot, followed by the equation or figure number within that appendix. Supplementary Appendix A provides a list of the main symbols used in the text, and their meanings.

The sensory coding model

Throughout this paper, we represent random variables using upper case letters, and their values on particular trials¹ using corresponding lower case letters. X is a random variable representing the stimulus level, and its value is x . R is a random variable representing a neuron's mean number of spikes, and its value is $r(x)$, the output of the neuron's tuning function, which gives the neuron's mean spike count for a given stimulus, x (note that the tuning function's output is often measured in spikes per second, but we find it more convenient to use units of spikes, without making assumptions about the time period over which the neuron's output is integrated; this is equivalent to measuring the output in spikes per unit of time using units of time scaled so that the spike integration period is one unit). N is a random variable representing the spike count of a neuron, i.e. the number of spikes produced by the neuron, and n is its value on a particular trial. Because we will often be dealing with populations of neurons, we use bold letters, \mathbf{N} and \mathbf{n} , to represent vectors holding the spike counts of all the neurons in the population. \mathbf{N} is a random variable representing the population response, and \mathbf{n} is its value on a particular trial.

We assume a generic sensory coding model that has three elements: (1) a tuning function, $r(x)$, for each neuron, which specifies the neuron's mean spike count for a given stimulus, x ; (2) a random process that generates spikes at the given mean rate; (3) a method of decoding the spike counts of the neurons to give an estimate of the stimulus. We now describe each of these processes in turn.

The tuning function

In this paper, we consider two different tuning functions, the Naka-Rushton function and the Gaussian function. Both have parameters r_0 , r_{\max} , z , q , and b , which serve the same or analogous purposes in the two functions.

The Naka-Rushton tuning function

For visual stimulus contrast, the tuning function is called the *contrast-response function*. It usually has a sigmoidal shape that is well described by the Naka-Rushton function, also known as the hyperbolic ratio function (Naka & Rushton, 1966; Albrecht & Hamilton, 1982):

$${}_{\text{N-R}} r(c) = \frac{r_{\max} c^q}{c_{1/2}^q + c^q} + r_0. \quad (1)$$

c is the contrast in linear (e.g. Michelson) units, r_0 is the spontaneous firing rate in response to zero contrast, and r_{\max} is the asymptotic increment from r_0 as contrast increases. The term $c_{1/2}$, called the *semi-saturation contrast*, is the contrast for which the mean response exceeds r_0 by $r_{\max}/2$. We use a left subscript on $r(\cdot)$ to indicate the form of the tuning function, in this case "N-R" for "Naka-Rushton".

Examples of the Naka-Rushton function are plotted in Figure 1A. Some readers may not think of the Naka-Rushton function as a "tuning function", but we use the latter term in its most general sense, to mean the function that gives the mean response for a particular stimulus value.

For our purposes, it is convenient to represent the contrast in units of log Michelson contrast, so the stimulus value, x , is given by

$$x = \log_b c, \quad (2)$$

which gives

$$c = b^x. \quad (3)$$

Using Equation (3) to substitute for c in Equation (1), we can re-express the Naka-Rushton function as a function of log contrast, x :

$${}_{\text{N-R}} r(x) = \frac{r_{\max} b^{qx}}{b^{qz} + b^{qx}} + r_0, \quad (4)$$

where

$$z = \log_b c_{1/2}. \quad (5)$$

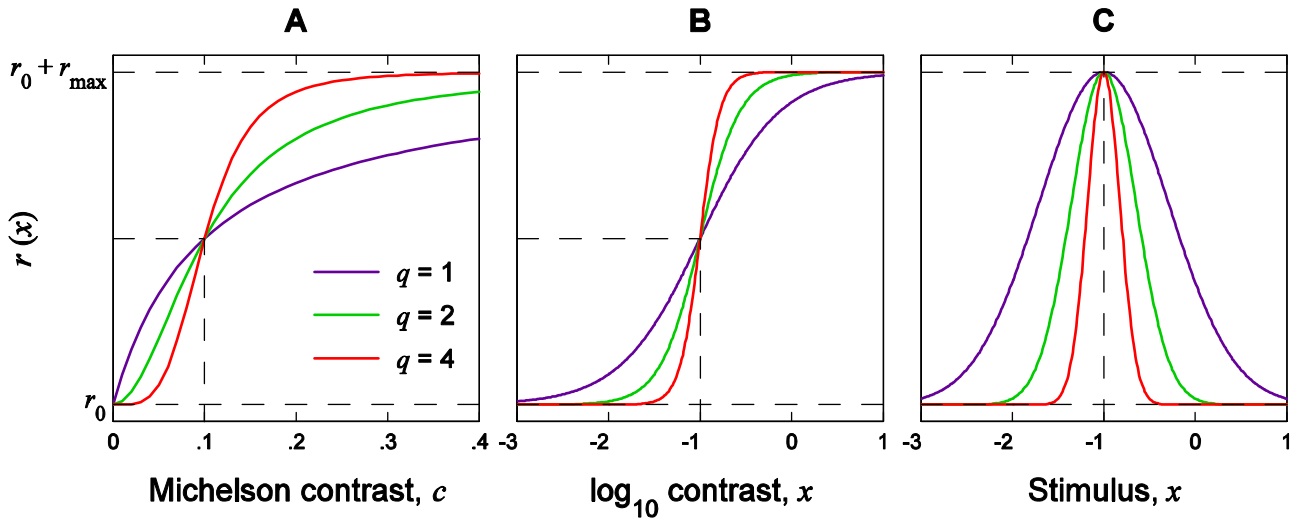


Figure 1. Tuning functions. (A) Three Naka-Rushton functions, with q equal to 1, 2, and 4. Each function has $c_{1/2} = 0.1$. $r(c_{1/2})$ falls halfway between r_0 and $r_0 + r_{\max}$. (B) The same three Naka-Rushton functions as in (A), plotted as functions of \log_{10} contrast. The log semi-saturation contrast is given by $z = -1$. (C) Three Gaussian tuning functions, with $z = -1$ and q equal to 1, 2, and 4.

Figure 1B plots the Naka-Rushton function as a function of log contrast. On the log contrast scale, the gradient of the Naka-Rushton function peaks at a log contrast of $x = z$. In what follows, whenever we use the term “contrast” without specifying the units, we mean “log Michelson contrast”. In all our simulations, we used log to base 10 (i.e. $b = 10$), but our equations are derived for any b .

The Gaussian tuning function

We will use z to represent the stimulus value corresponding to the centre (peak) of the tuning function. This is analogous to the semi-saturation contrast, z , which is at the centre (peak of gradient) of the Naka-Rushton contrast-response function when expressed in terms of log contrast. The Gaussian tuning function is given by

$$r_{\text{Gauss}}(x) = r_{\max} \exp\left\{-[q(x-z)]^2\right\} + r_0, \quad (6)$$

where x is a value along the (unspecified) stimulus dimension, r_{\max} is the maximum increment from the spontaneous firing rate, r_0 , and q is a “tuning sharpness” parameter, analogous to the exponent, q , of the Naka-Rushton function. As before, the left subscript on $r(x)$ (in this case, “Gauss”, for Gaussian) indicates the form of the tuning function. Examples of the Gaussian tuning function are plotted in Figure 1C.

If σ_{tuning} is the standard deviation of the Gaussian tuning function, then its bandwidth (full width at half height), w , is given by $w^2 = (8 \ln 2) \sigma_{\text{tuning}}^2$. Thus, we have

$$q = \frac{1}{\sigma_{\text{tuning}} \sqrt{2}} = \frac{2\sqrt{\ln 2}}{w}. \quad (7)$$

If x is measured in units that are the log to base b of the physical stimulus units, then w will also be in \log_b units. We can convert w to octaves (i.e., \log_2 units) by multiplying by $\log_2 b$ (this is because $\log_b y \times \log_2 b = \log_2 y$). If ω is the bandwidth in octaves, then

$$\omega = w \log_2 b = \frac{w \ln b}{\ln 2}. \quad (8)$$

Using Equation (8) to substitute for w in Equation (7) gives

$$q = \frac{2 \ln b}{\omega \sqrt{\ln 2}}. \quad (9)$$

The random process for spike generation

In our model, the spikes are generated by a model of neuronal spiking recently proposed by Goris, Movshon and Simoncelli (2014). Each model neuron has a Poisson spiking process whose mean spike rate is modulated by a multiplicative gain mechanism that is also a random process, so the overall spiking process is a doubly stochastic Poisson process. Specifically, for each model neuron, the rate of its Poisson spiking process is determined by the

output of its tuning curve, $r(x)$, multiplied by a gain value that changes randomly from trial to trial. We let G represent the gain random variable, and let g represent its value on a particular trial. The spiking distribution conditioned on the gain is an ordinary Poisson distribution:

$$P(N = n | R = r(x), G = g) = \frac{(gr(x))^n \exp(-gr(x))}{n!}. \quad (10)$$

In Goris et al.'s (2014) spiking model, the gain values are sampled from a gamma distribution with a mean of 1, and standard deviation that is a free parameter, σ_G . The gamma gain distribution and Poisson spiking distribution combine to produce a gamma-Poisson mixture distribution that has the form of a negative binomial distribution of spike counts, given by

$$P(N = n | R = r(x)) = \frac{\Gamma(n+1/\sigma_G^2)}{n! \Gamma(1/\sigma_G^2)} \times \frac{(\sigma_G^2 r(x))^n}{(\sigma_G^2 r(x) + 1)^{(n+1/\sigma_G^2)}}, \quad (11)$$

where $\Gamma(\cdot)$ is the gamma function. The distribution in Equation (11) fits well to the spike distributions obtained in physiological recordings (see Goris et al., 2014, Figure 1c and 1d). Because the mean gain is 1, the mean of the spike distribution in Equation (11) is $r(x)$.

Goris et al. (2014) allowed both the spiking process and the gain process to be correlated between neurons. The covariance between the spike rates of neurons i and j is then given by

$$\text{cov}[N_i, N_j] = \rho_{P_{ij}} \sqrt{r_i(x)r_j(x)} + \rho_{G_{ij}} \sigma_{G_i} \sigma_{G_j} r_i(x)r_j(x), \quad (12)$$

where $r_i(x)$ and $r_j(x)$ are the outputs of the tuning curves of neurons i and j given a stimulus x , σ_{G_i} and σ_{G_j} are the standard deviations of the two neurons' gain processes, $\rho_{P_{ij}}$ is the Pearson correlation between the two neurons' Poisson spiking processes, and $\rho_{G_{ij}}$ is the Pearson correlation between the two neurons' gain processes. If we let $i = j$ in Equation (12), then we obtain the variance of neuron j :

$$\text{var}[N_j] = \text{cov}[N_j, N_j] = r_j(x) + (\sigma_{G_j} r_j(x))^2. \quad (13)$$

This can be converted to the neuron's Fano factor (i.e. the ratio of variance to mean) by dividing by the mean response, $r_j(x)$:

$$\text{Fano factor } [N_j] = 1 + \sigma_{G_j}^2 r_j(x). \quad (14)$$

Equation (14) shows that the model neuron's Fano factor is an affine (i.e., straight line) function of the spike rate, with gradient equal to the variance of the gain process: The Fano factor is around 1 for very low spike rates, and increases linearly with the spike rate.

In order to meet our stated goal of mathematical simplicity, we imposed three restrictions on the aforementioned parameters:

1. each neuron has the same gain standard deviation, which is set as a model parameter, σ_G ;
2. $\rho_{G_{ij}} = 1$ for all i, j ;
3. $\rho_{P_{ij}} = 0$ for all $i \neq j$.

The second and third restrictions state, respectively, that each neuron shares the same fluctuating gain signal, and each neuron's Poisson spiking process is independent. This is an extreme version of Goris et al.'s (2014) model, but it seems not too far from the truth for many pairs of neurons: Very high-quality electrophysiological recordings indicate that, in awake monkeys whose only task is to fixate the stimulus, V1 neurons "show virtually no correlated variability" (Ecker et al., 2010, p. 584), suggesting that $\rho_{P_{ij}}$ is usually close to zero. Ecker et al.

(2014) argued that the very much higher correlations shown in other circumstances, especially under anesthesia, can be accounted for by a single gain variable that varies from trial to trial. They showed that a simple model with independent Poisson spiking processes and a single random gain signal shared between all the neurons was able to capture the dependence of spike count correlations on firing rate and tuning similarity shown by real neurons. Thus, a model very similar to ours, in which $\rho_{P_{ij}} = 0$ and $\rho_{G_{ij}} = 1$, has been shown to account for real neuronal data.

These restrictions help us to simplify the analysis of the model considerably. Firstly, instead of specifying the entire covariance matrix of the population, we can just specify the gain standard deviation, σ_G . The covariance matrix itself still has a complicated structure because each term in the covariance matrix depends on σ_G and the sensitivities of the two neurons to the stimulus, but all of this complexity can be reduced to a single scalar random variable, G , with a single parameter, σ_G . Secondly, as ex-

plained above, we aim to calculate the performance of the decoder that makes best use of the encoded signals; and the best possible decoder will have access to the shared gain signal. If $\rho_{P_{ij}} = 0$, then, although the neurons' spike counts are correlated due to the shared gain signal, the *conditional* spiking probabilities (conditioned upon the gain signal) are independent. Thus, a decoder that knows the gain signal can express the spike distributions as independent Poisson distributions, which considerably simplifies both the maximum-likelihood decoding algorithm and our mathematical analysis of its performance. We also investigated the performance of two suboptimal decoders that did not know the gain signal. Loss of knowledge of the gain signal greatly impaired performance with the Naka-Rushton tuning function but, with the Gaussian tuning function, decoding performance was only slightly affected.

With the three restrictions listed above, we can rewrite Equation (12) as

$$\text{cov}[N_i, N_j] = \sigma_G^2 r_i(x) r_j(x). \quad (15)$$

The Pearson correlation between the two neurons' spike counts is then given by

$$\rho_{ij} = \frac{\text{cov}[N_i, N_j]}{\sqrt{\text{var}[N_i] \text{var}[N_j]}} \quad (16)$$

$$= \sigma_G^2 \sqrt{\frac{r_i(x) r_j(x)}{(1 + \sigma_G^2 r_i(x))(1 + \sigma_G^2 r_j(x))}}. \quad (17)$$

As noted by Goris et al. (2014), their model elegantly captures two key physiological findings, which both follow from their equations: Pair-wise correlations generally increase with both firing rate (de la Rocha, Doiron, Shea-Brown, Josic, & Reyes, 2007; Ecker et al., 2010, 2014; Cohen & Kohn, 2011) and tuning similarity (van Kan, Scobey, & Gabor, 1985; Zohary, Shadlen, & Newsome, 1994; Smith & Kohn, 2008; Ecker et al., 2010, 2014). Supplementary Appendix B confirms that, for nonzero σ_G , both of these characteristics emerge naturally from Equation (17). The dependence of ρ_{ij} on tuning similarity is particularly noteworthy, since, in our parameterizations of the model, we have forced the Poisson and gain correlations, $\rho_{P_{ij}}$ and $\rho_{G_{ij}}$, to be constant across all pairs of neurons, regardless of tuning similarity.

The decoding method

In our sensory coding model, a stimulus triggers a set of spike counts, \mathbf{n} , in the neurons being monitored by the observer: \mathbf{n} is a vector holding the spike counts of all the different neurons. We investigated three methods for decoding these spike counts.

The first is a maximum-likelihood method that uses knowledge of the gain signal to express the spike distributions as independent Poisson distributions; in this case, maximum-likelihood decoding is straightforward because the likelihood function for the population response is simply the product of the likelihood functions for the individual responses. We call this the "Known Gain" method.

We were also interested in finding out how well the spike counts could be decoded without knowledge of the gain signal, to see how critically the accuracy of our equations depended on the assumption that the gain signal is known. If the decoder does not know the gain signal, then it can only express the spike distributions as correlated gamma-Poisson mixture distributions. Maximum-likelihood decoding in this case would require an expression for the multivariate gamma-Poisson mixture distribution, so that we could calculate the likelihood function for the population response. Unfortunately, a closed-form expression for this distribution beyond the bivariate case is not known, so we used two suboptimal methods, which we refer to collectively as the "Unknown Gain" methods. The first of these methods simply ignores the correlations and decodes the neurons as if they were independent (an approach also taken by Goris et al., 2013). This approach is not always suboptimal: Averbeck, Latham and Pouget (2006) showed that, even in cases where correlations cause a large performance deficit, it can still be optimal to decode the neuronal responses as if they were independent (see Averbeck et al.'s Figure 3a). More generally, however, it is more informative to decode correlated neurons as a group rather than individually. The second of our "Unknown Gain" methods uses an analytical form for the bivariate gamma-Poisson mixture distribution to decode the population in a way that takes into account all pairwise statistical dependencies, but not higher-order dependencies. We refer to these two "Unknown Gain" methods as the "Univariate" and "Bivariate" methods, respectively. We find that the "Bivariate" method performs better than the "Univariate" method. Surprisingly, when the tuning function is a Gaussian, both methods are almost as good as the "Known Gain" method.

The "Known Gain" decoder

Here, the spike counts of the neurons are decoded using a maximum-likelihood method in which the gain signal is known. In maximum-likelihood decoding, we find

the stimulus value, x , which had the highest probability of giving rise to the obtained set of spike counts, i.e. the value of x that maximizes the probability $P(\mathbf{N} = \mathbf{n} | X = x, G = g)$. Because of the statistical independence of the spiking distributions conditioned on the gain signal, we can write

$$P(\mathbf{N} = \mathbf{n} | X = x, G = g) = \prod_{j=1}^K P(N_j = n_j | X = x, G = g) \quad (18)$$

$$= \prod_{j=1}^K P(N_j = n_j | R_j = r_j(x), G = g). \quad (19)$$

where K is the number of neurons, and the neurons are indexed by j . The second equality (Equation (19)) arises because each $r_j(x)$ is a deterministic function of the stimulus value, x . For large populations, the product in Equation (19) can be too small to represent using floating-point values on a standard computer, so instead we calculated the logarithm of this value, which peaks at the same x -value, and is given by

$$\ln P(\mathbf{N} = \mathbf{n} | X = x, G = g) = \sum_{j=1}^K \ln P(N_j = n_j | R_j = r_j(x), G = g). \quad (20)$$

The probability in each term in Equation (20) is evaluated using Equation (10), and the decoder uses the simplex search method (Nelder & Mead, 1965) to find the stimulus value, x , that maximizes this sum.

The “Univariate” (Unknown Gain) decoder

Without knowledge of the gain signal, the spiking distributions are negative binomial distributions with the form given in Equation (11). The “Univariate” decoder decodes the spike counts as if they were independent. Thus, we found the stimulus value, x , that maximized the sum of log probabilities, $\sum_{j=1}^K \ln P(N_j = n_j | R_j = r_j(x))$, where the probability in each term is evaluated using Equation (11).

The “Univariate” decoder is not as good as the “Bivariate” decoder (described next), and our principal motivation for including it was to confirm that the “Bivariate” decoder did indeed perform better.

The “Bivariate” (Unknown Gain) decoder

Arbous & Kerrich (1951) derived an expression for the bivariate gamma-Poisson mixture distribution, which gives the joint probability of the spike counts of any pair of neurons in our restricted parameterizations of Goris et al’s (2014) spiking model. Substituting our variables for the corresponding variables or expressions in Arbous & Kerrich’s Equation (5.11), we obtain

$$P(N_i = n_i, N_j = n_j | X = x) = P(N_i = n_i, N_j = n_j | R_i = r_i(x), R_j = r_j(x)) \quad (21)$$

$$= \frac{\Gamma(n_i + n_j + 1/\sigma_G^2)}{n_i! n_j! \Gamma(1/\sigma_G^2)} \times$$

$$\frac{(\sigma_G^2)^{n_i + n_j} r_i(x)^{n_i} r_j(x)^{n_j}}{(\sigma_G^2 (r_i(x) + r_j(x)) + 1)^{n_i + n_j + 1/\sigma_G^2}}. \quad (22)$$

Here, we have deliberately expressed Arbous & Kerrich’s equation in a form that makes explicit the many parallels between the expressions for the bivariate and univariate distributions (Equations (22) and (11), respectively).

To decode the neuronal population, we took each pairwise combination of neurons, and calculated the likelihood for that pair, using Equation (22). The estimated stimulus value, x , was that which maximized the sum of log-likelihoods across all pairs of neurons. Thus, we decoded the population as if each pair of neurons was statistically independent from each other pair. As noted earlier, this decoding method takes account of pairwise statistical dependencies, but not higher-order dependencies.

Parameterizing the population

In principle, every parameter of every neuron in the population could vary independently. However, to maximize the simplicity of the model, we consider two simpler, more restricted classes of parameterization.

The “Constant” parameterization

To begin with, we consider a very simple class of parameterizations: r_{\max} , r_0 , q and σ_G are each constant across different neurons, and the tuning curve centres, z , are equally spaced along the log stimulus axis, with constant spacing, δz , between values of z_{\min} and z_{\max} . We define a *density* parameter, h , equal to $1/\delta z$. The model thus has seven parameters: r_{\max} , r_0 , q , σ_G , h , z_{\min} and

z_{\max} . Since each of these parameters is assumed to be constant across different neurons, we call this the “Constant” parameterization.

One important property of the “Constant” parameterization is that, if the stimulus value, x , is the logarithm of the physical stimulus level, ξ , then the model will show Weber’s Law behaviour. The reason for this is that, with all the parameters constant, if we move from a low stimulus value to a high stimulus value, we are faced with an identical decoding situation, but just shifted along the x -axis: The tuning functions have the same shape and density, and the noise properties are the same. Thus, the standard deviation of the decoded value of x will be constant across stimulus levels. This means that Δx_θ , the just-noticeable difference in x , will be constant. If ξ_p is the “pedestal” stimulus (i.e. the lower of the two stimuli to be discriminated) expressed in physical units, and $\Delta \xi_\theta$ is the just-noticeable difference in physical units for a pedestal of ξ_p , then $\Delta x_\theta = \log_b(\xi_p + \Delta \xi_\theta) - \log_b(\xi_p)$. This can be rearranged to give $\Delta \xi_\theta / \xi_p = b^{\Delta x_\theta} - 1$, which is constant for constant Δx_θ . Thus, the Weber fraction, $\Delta \xi_\theta / \xi_p$, is constant, which is the definition of Weber’s Law.

The “Exponential” parameterization

Although Weber’s Law has often been found in suprathreshold discrimination experiments, the Weber fraction for contrast discrimination generally decreases with increasing contrast, to give a slope of around 0.6-0.7 when the threshold difference of Michelson contrasts is plotted as a function of the pedestal Michelson contrast on log-log axes (Legge, 1981; Meese, Georgeson, & Baker, 2006). This is often referred to as the “near-miss to Weber’s Law”. To capture this behaviour, we used a parameterization in which h , r_{\max} , and q can increase with the tuning function’s position along the log stimulus axis. Specifically, we let h , r_{\max} , and q vary as exponential functions of tuning function position, z :

$$h = k_h \exp(m_h z) \quad (23)$$

$$r_{\max} = k_{r_{\max}} \exp(m_{r_{\max}} z) \quad (24)$$

$$q = k_q \exp(m_q z). \quad (25)$$

k_h , $k_{r_{\max}}$ and k_q are parameters that give the values of h , r_{\max} , and q when $z = 0$; m_h , $m_{r_{\max}}$ and m_q are parameters that determine how rapidly h , r_{\max} and q change as a

function of z . We call this the “Exponential” parameterization. It is a generalization of the “Constant” parameterization: The “Constant” parameterization is the “Exponential” parameterization with $m_h = m_{r_{\max}} = m_q = 0$. For simplicity, we will assume that r_0 / r_{\max} is constant in the “Exponential” parameterization. As before, z falls between z_{\min} and z_{\max} . Supplementary Appendix C shows how to generate a set of z values when h varies exponentially across the stimulus axis.

Predicting model performance

As noted in the Introduction, the decoding precision (i.e. the reciprocal of the variance of the decoded stimulus value, \hat{X}) is closely approximated by the Fisher information. In this section, we derive expressions for the Fisher information for decoding the neurons using the “known gain” decoder. On each trial, the tuning curve of each neuron in our model is multiplied by the gain signal, g , so the effective tuning curve for neuron j on that trial is $gr_j(x)$. If the decoder knows the gain signal, then it knows the effective tuning curve, $gr_j(x)$, and it can express the spiking distributions as independent Poisson distributions. For a set of independent Poisson-spiking neurons with tuning curves $gr_j(x)$, the Fisher information is given by

$$J(x, g) = g \sum_{j=1}^K \frac{r_j'(x)^2}{r_j(x)}, \quad (26)$$

where $r_j'(x)$ is the first derivative of neuron j ’s tuning curve, with respect to x (see Dayan & Abbott, 2001, chapter 3). Thus, the variance of the estimated stimulus value, \hat{X} , on trials with stimulus x and gain g will be approximated by

$$\text{var}[\hat{X} | G = g] \approx \frac{1}{J(x, g)} = \frac{1}{gJ(x, 1)}. \quad (27)$$

Over all trials with stimulus x , the variance will be given by

$$\text{var}[\hat{X}] \approx \text{mean} \left[\frac{1}{GJ(x, 1)} \right] \quad (28)$$

$$= \frac{\text{mean}[1/G]}{J(x, 1)}. \quad (29)$$

Thus, the precision, $\tau(x)$, for decoding a stimulus with value x , is given by

$$\tau(x) \approx \frac{J(x,1)}{\text{mean}[1/G]}. \quad (30)$$

Supplementary Appendix D shows that, for the gamma distributed gain signal in Goris et al.'s (2014) model, $\text{mean}[1/G] = 1/(1 - \sigma_G^2)$, giving

$$\tau(x) \approx (1 - \sigma_G^2)J(x,1). \quad (31)$$

From Relation (31), we have $\tau(x) \approx J(x,1 - \sigma_G^2)$. Interestingly, $1 - \sigma_G^2$ is the value of the gain with the highest probability density. Thus, the decoding precision is the *modal* value of the Fisher information across trials, not the mean.

In this paper, we consider 2-alternative forced-choice discrimination experiments, in which observers have to discriminate a ‘‘pedestal’’ stimulus value from a slightly higher stimulus value. If x is the log to base b of the physical stimulus value, ξ , then, if the pedestal stimulus value is ξ_p , and the difference in physical stimulus values at threshold is $\Delta\xi_\theta$, then the Weber fraction, W , is defined as

$$W = \Delta\xi_\theta / \xi_p. \quad (32)$$

Appendix E shows that, in this case, the Weber fraction is approximated by

$$W \approx b^{\sqrt{2/\tau(x_p)}\Phi^{-1}(P_\theta)} - 1, \quad (33)$$

where Φ is the integral of a Gaussian with unit area and variance and zero mean, and P_θ is the proportion of correct responses that defines the threshold level. We can then calculate the Weber fraction by using Equation (31) to substitute for $\tau(x)$ in Equation (33).

In summary, to predict the model's performance, we first calculate $J(x,1)$, the Fisher information that would occur when the gain is 1. Then we multiply this by $1 - \sigma_G^2$ to give $\tau(x)$, the overall precision when the gain varies with standard deviation σ_G (Equation (31)). We can then calculate the Weber fraction from the precision using Relation (33). The next two sections derive various expressions for $J(x,1)$ for different model parameterizations. The first of these two sections derives exact expressions; the second section derives compact approximations of these expressions, which can often give a better insight into the way that the different neuronal parameters are

related to perceptual performance in a population coding model.

Exact expressions for the Fisher information

Using the Naka-Rushton function (Equation (4)) to expand $r_j(x)$ in Equation (26), we get

$$\begin{aligned} \text{N-R } J(x,1) = & \\ & \sum_{j=1}^K \frac{r_{\max} (q \ln b)^2 b^{2q(z+x)}}{(b^{qz} + b^{qx})^3 (b^{qx} + (r_0/r_{\max})(b^{qz} + b^{qx}))}, \end{aligned} \quad (34)$$

where the parameters on the right hand side are the neuronal parameters of the Naka-Rushton function from Equation (4), and can vary from neuron to neuron (strictly speaking, each parameter should be indexed by the neuron number, j , but we omit these indices for clarity). The left subscript on $\text{N-R } J(x,1)$ indicates the form of the tuning function, in this case ‘‘N-R’’ for Naka-Rushton. When $r_0 = 0$, Equation (34) reduces to

$$\text{N-R } J(x,1) = \sum_{j=1}^K \frac{r_{\max} (q \ln b)^2 b^{q(2z+x)}}{(b^{qz} + b^{qx})^3} \quad \text{if } r_0 = 0. \quad (35)$$

Going through a similar process for the Gaussian tuning function, we obtain

$$\begin{aligned} \text{Gauss } J(x,1) = & \\ & 4 \sum_{j=1}^K \frac{r_{\max} \left\{ q^2 (x-z) \exp(-[q(x-z)]^2) \right\}^2}{\exp(-[q(x-z)]^2) + r_0/r_{\max}}, \end{aligned} \quad (36)$$

which reduces to

$$\begin{aligned} \text{Gauss } J(x,1) = & 4 \sum_{j=1}^K r_{\max} q^4 (x-z)^2 \exp(-[q(x-z)]^2) \\ & \text{if } r_0 = 0. \end{aligned} \quad (37)$$

Equations (34) to (37) are good for fitting the model to psychophysical data, but each of these equations is a

sum with one term for each neuron, and this complexity can make it difficult to see how the different neuronal parameters are related to perceptual performance in a population coding model. We therefore derived approximations to these equations by approximating the sum using an integral. This allowed us to write down the Fisher information for the whole population in a single compact expression. We call these expressions “integral approximations”. To distinguish these from the true Fisher information expressions, we use the letter I to represent them, rather than J (think of I for *I*ntegral approximation).

These approximations are described in the next section. We consider three classes of parameterization: The “Constant” parameterization with either the Naka-Rushton or the Gaussian tuning function, and the “Exponential” parameterization with the Naka-Rushton tuning function. For each of these, we derive an integral approximation of the Fisher information. Any of these expressions for Fisher information can be used in place of $J(x,1)$ in Equation (31) to predict the decoding precision.

Integral approximations of the Fisher information

The “Constant Naka-Rushton” parameterization

The “Constant Naka-Rushton” parameterization is the “Constant” parameterization that uses the Naka-Rushton tuning function. Since all the parameters except z are constant, and $\delta z = 1/h$, we can rearrange the right hand side of Equation (34) to give

$$\begin{aligned} \text{N-R } J(x,1) &= r_{\max} (q \ln b)^2 b^{2qx} h \times \\ &\sum_z \frac{b^{2qz}}{(b^{qz} + b^{qx})^3 (b^{qx} + (r_0/r_{\max})(b^{qz} + b^{qx}))} \delta z. \end{aligned} \quad (38)$$

As $\delta z \rightarrow 0$, the right hand side of Equation (38) can be approximated by an integral, which we call $\text{Const}_{\text{N-R}} I(x,1)$. As long as the stimulus value, x , is sufficiently far from the edges of the range of z -values, we can take the limits of z to be $\pm\infty$, giving

$$\text{Const}_{\text{N-R}} I(x,1) = r_{\max} (q \ln b)^2 b^{2qx} h \times$$

$$\int_{-\infty}^{\infty} \frac{b^{2qz}}{(b^{qz} + b^{qx})^3 (b^{qx} + (r_0/r_{\max})(b^{qz} + b^{qx}))} dz. \quad (39)$$

We used Mathematica (Wolfram Research) to solve the integral, and obtained

$$\text{Const}_{\text{N-R}} I(x,1) = \frac{\ln(b)}{2} \times r_{\max} qh \times Q(r_0/r_{\max}), \quad (40)$$

where

$$Q(y) = \begin{cases} 1 + 2y[1 - (1+y)\ln(1+1/y)] & y > 0 \\ 1 & y = 0 \end{cases}. \quad (41)$$

Figure 2 plots $Q(r_0/r_{\max})$ for $0 \leq r_0/r_{\max} \leq 1$, and it can be seen that $Q(r_0/r_{\max})$ smoothly decreases with increasing r_0/r_{\max} . Thus, as r_0 increases from 0, $\text{Const}_{\text{N-R}} I(x,1)$ undergoes a multiplicative attenuation that is a function only of the ratio, r_0/r_{\max} . Because the attenuation is a function of the ratio, r_0/r_{\max} , rather than r_0 alone, we will often take this ratio to be a model parameter, rather than r_0 . We refer to this ratio as the “relative spontaneous firing rate”. When $r_0 = 0$, $Q(r_0/r_{\max}) = 1$, and Equation (40) reduces to

$$\text{Const}_{\text{N-R}} I(x,1) = \frac{\ln(b)}{2} \times r_{\max} qh \quad \text{if } r_0 = 0 \quad (42)$$

The $\ln(b)/2$ part of this expression is just a constant that depends on the arbitrary choice of base of logarithm that we use to represent contrast (and reduces to 1 for $b = e^2$). The interesting part is $r_{\max} qh$: This is the simplest expression that we could possibly imagine, given that the Fisher information has to increase with increasing r_{\max} , q and h . Equation (42) therefore reveals a remarkably straightforward relationship between the Fisher information and the neuronal parameters.

Note that both expressions for $\text{Const}_{\text{N-R}} I(x,1)$ (Equations (40) and (42)) are independent of the stimulus value, x . From Equation (33), this leads to a constant Weber fraction, i.e. Weber’s Law.

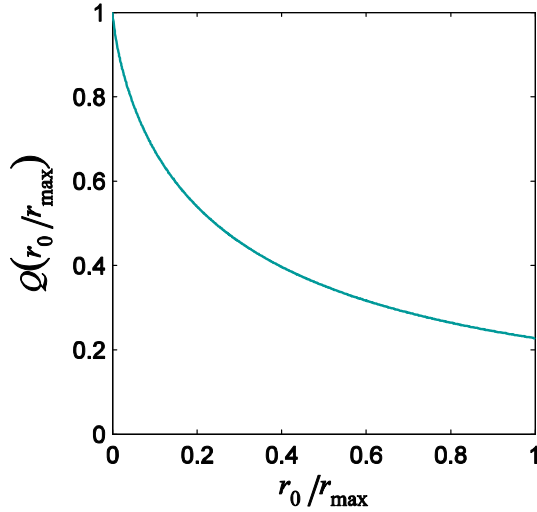


Figure 2. Attenuation of Fisher information for non-zero r_0 . The curve plots $Q(r_0/r_{\max})$ as defined in Equation (41).

The “Constant Gaussian” parameterization

The “Constant Gaussian” parameterization is the “Constant” parameterization that uses the Gaussian tuning function. We begin by assuming that $r_0 = 0$. Since all the parameters except z are constant, and $\delta z = 1/h$, we can rearrange the right hand side of Equation (37) to give

$$\begin{aligned} J_{\text{Gauss}}(x, 1) &= \\ &4r_{\max}q^4h \sum_z (x-z)^2 \exp\{-[q(x-z)]^2\} \delta z. \end{aligned} \quad (43)$$

When $\delta z \rightarrow 0$, the right hand side of Equation (43) can be approximated by an integral, which we call $I_{\text{Gauss}}^{\text{Const}}(x, 1)$:

$$\begin{aligned} I_{\text{Gauss}}^{\text{Const}}(x, 1) &= \\ &4r_{\max}q^4h \int_{-\infty}^{\infty} (x-z)^2 \exp\{-[q(x-z)]^2\} dz \\ &\text{if } r_0 = 0. \end{aligned} \quad (44)$$

The integral in Equation (44) is a standard definite integral, and Equation (44) reduces to

$$I_{\text{Gauss}}^{\text{Const}}(x, 1) = 2\sqrt{\pi} \times r_{\max} h q \quad \text{if } r_0 = 0. \quad (45)$$

Equation (45) applies only to the case of $r_0 = 0$ because we started with Equation (37). If instead we start

with Equation (36), we obtain an expression that applies to all r_0 :

$$\begin{aligned} J_{\text{Gauss}}(x, 1) &= \\ &4r_{\max}q^4h \sum_z \frac{(x-z)^2 \exp\{-2[q(x-z)]^2\}}{\exp\{-[q(x-z)]^2\} + r_0/r_{\max}} \delta z. \end{aligned} \quad (46)$$

When $\delta z \rightarrow 0$, the right hand side of Equation (46) can be approximated by an integral:

$$J_{\text{Gauss}}(x, 1) \approx 4r_{\max}q^4h \times S(q), \quad (47)$$

where

$$S(q) = \int_{-\infty}^{\infty} \frac{(x-z)^2 \exp\{-2[q(x-z)]^2\}}{\exp\{-[q(x-z)]^2\} + r_0/r_{\max}} dz. \quad (48)$$

In Supplementary Appendix F, we show that

$$S(q) = \frac{S(1)}{q^3}. \quad (49)$$

Using Equation (49) to substitute for $S(q)$ in Equation (47), we obtain

$$J_{\text{Gauss}}(x, 1) \approx 4r_{\max}qh \times S(1). \quad (50)$$

We can write $S(1)$ as

$$S(1) = \int_{-\infty}^{\infty} \frac{z^2 \exp(-2z^2)}{\exp(-z^2) + r_0/r_{\max}} dz. \quad (51)$$

Note that we were able to drop the x that appears in the function being integrated, because this just shifts the function horizontally by a finite amount, x , along the z -axis but does not change its integral between infinite limits; so $S(1)$ is a function of r_0/r_{\max} only. Unfortunately, for $r_0 > 0$, we cannot find a closed form expression for $S(1)$. However, for the range of relative spontaneous firing rates likely to occur, we have found that it can be very closely approximated by

$$S(1) \approx \frac{\sqrt{\pi}Q(r_0/r_{\max})}{2}, \quad (52)$$

where the function, Q , is defined in Equation (41). Supplementary Appendix F shows that, for $0 < r_0/r_{\max} < 0.119$, the approximation on the right hand

side of Relation (52) slightly overestimates $S(1)$ by a factor that never exceeds 0.7% of the true value; for $r_0/r_{\max} > 0.120$, the right hand side of Relation (52) underestimates $S(1)$, but not by much: Even for r_0/r_{\max} as high as 1, the underestimation is only about 3%, and the underestimation is always less than 6% of $S(1)$. Using Relation (52) to substitute for $S(1)$ in Relation (50), we obtain our integral approximation, $I_{\text{Gauss}}^{\text{Const}}(x,1)$ for any r_0 :

$$I_{\text{Gauss}}^{\text{Const}}(x,1) = 2\sqrt{\pi} \times r_{\max} q h \times Q(r_0/r_{\max}). \quad (53)$$

Note that, apart from having a different multiplicative constant ($2\sqrt{\pi}$ instead of $\ln(b)/2$), Equation (53) is identical to Equation (40), which gives the corresponding expression for the Naka-Rushton tuning function. Similarly, $I_{\text{Gauss}}^{\text{Const}}(x,1)$ is independent of x , which leads to Weber's Law.

The "Exponential Naka-Rushton" parameterization

The "Exponential Naka-Rushton" parameterization is the "Exponential" parameterization that uses the Naka-Rushton tuning function. We begin by assuming that $r_0 = 0$, so the Fisher information is given by Equation (35). For now, let us also assume that q is constant, while h and r_{\max} vary with z , according to Equations (23) and (24). Then we can rearrange the right hand side of Equation (35) to give

$$J_{\text{N-R}}(x,1) = (q \ln b)^2 b^{qx} \sum_z \frac{r_{\max} b^{2qz}}{(b^{qz} + b^{qx})^3}. \quad (54)$$

Using Equation (24) to substitute for r_{\max} in Equation (54), we have

$$J_{\text{N-R}}(x,1) = k_{r_{\max}} (q \ln b)^2 b^{qx} \sum_z \frac{\exp(m_{r_{\max}} z) b^{2qz}}{(b^{qz} + b^{qx})^3}. \quad (55)$$

To convert to an integral, we need to transform the z values so that the transformed values, ζ , are equally spaced. Supplementary Appendix C shows that an appropriate transformation is given by

$$\zeta = \frac{e^{m_h z}}{m_h}, \quad (56)$$

giving

$$z = \ln(m_h \zeta) / m_h. \quad (57)$$

Using Equation (57) to substitute for z in Equation (55), we obtain

$$J_{\text{N-R}}(x,1) = k_{r_{\max}} (q \ln b)^2 b^{qx} \times \sum_{\zeta} \frac{(m_h \zeta)^{m_{r_{\max}}/m_h} b^{2q \ln(m_h \zeta)/m_h}}{(b^{q \ln(m_h \zeta)/m_h} + b^{qx})^3}. \quad (58)$$

As shown in Supplementary Appendix C, the definition of ζ in Equation (56) causes the neurons to be separated in equal steps of size $\delta\zeta = 1/k_h$ along the ζ axis when h varies exponentially with z according to Equation (23). Thus, we have

$$J_{\text{N-R}}(x,1) = k_{r_{\max}} k_h (q \ln b)^2 b^{qx} \times \sum_{\zeta} \frac{(m_h \zeta)^{m_{r_{\max}}/m_h} b^{2q \ln(m_h \zeta)/m_h}}{(b^{q \ln(m_h \zeta)/m_h} + b^{qx})^3} \delta\zeta. \quad (59)$$

As $\delta\zeta \rightarrow 0$, the right hand side of Relation (59) can be approximated by an integral, which we call $I_{\text{N-R}}^{\text{Exp}}(x,1)$:

$$I_{\text{N-R}}^{\text{Exp}}(x,1) = k_{r_{\max}} k_h (q \ln b)^2 b^{qx} \times \int_0^{\infty} \frac{(m_h \zeta)^{m_{r_{\max}}/m_h} b^{2q \ln(m_h \zeta)/m_h}}{(b^{q \ln(m_h \zeta)/m_h} + b^{qx})^3} d\zeta. \quad (60)$$

The limits of 0 and ∞ on the integral arise because, as before, we assume that the stimulus value, x , is far from the ends of the range of z -values, and so the limits of z are effectively $\pm\infty$; from Equation (56), as $z \rightarrow -\infty$, $\zeta \rightarrow 0$, and, as $z \rightarrow \infty$, $\zeta \rightarrow \infty$. In Supplementary Appendix G, we derive an expression for the integral in Equation (60). This integral has a finite solution if $q \ln b > m_h + m_{r_{\max}}$, in which case the solution is given by Equation (G.18). Using Equation (G.18) to substitute for the integral in Equation (60), and simplifying, we obtain

$$I_{\text{N-R}}^{\text{Exp}}(x,1) = \frac{k_{r_{\max}} k_h \pi m (m/\gamma + 1) e^{mx}}{2 \sin(\pi m/\gamma)}. \quad (61)$$

where

$$m = m_h + m_{r_{\max}} \quad (62)$$

and

$$\gamma = q \ln b, \quad (63)$$

subject to the restriction that

$$\gamma > m. \quad (64)$$

It can be shown that $\text{Exp}_{\text{N-R}} I(x,1) \rightarrow \text{Const}_{\text{N-R}} I(x,1)$ as $m \rightarrow 0$, which is essential because, if the “ m ” parameters are all zero, then the “Exponential” parameterization reduces to the “Constant” parameterization.

Equation (61) shows two notable features. Firstly, the Fisher information is an exponential function of x : Specifically, when $h \propto \exp(m_h z)$ and $r_{\max} \propto \exp(m_{r_{\max}} z)$, the Fisher information is proportional to $\exp[(m_h + m_{r_{\max}})x]$. Secondly, $\text{Exp}_{\text{N-R}} I(x,1)$ is a function of the sum of m_h and $m_{r_{\max}}$, regardless of their individual values. Thus, m_h and $m_{r_{\max}}$ can be exactly traded off against each other, and the Fisher information will not change, as long as $m_h + m_{r_{\max}}$ remains constant.

Equation (61) assumes that q is constant (i.e., $m_q = 0$). Allowing q to vary with z greatly complicates the integral, and we were unable to solve it, so instead we used an approximation. First, we extend the definition of γ (Equation (63)) as follows:

$$\gamma = k_q \exp(m_q x) \ln b. \quad (65)$$

Here, q is treated as a function of x , but using the parameters that define how it varies with z (Equation (25)); however, the approximation is good enough because the performance for a log contrast of x will be dominated by the neurons with z close to x . If we then use Equation (65) instead of Equation (63) to substitute for γ in Equation (61), we obtain a very good approximation of the Fisher information when q varies as an exponential function of z .

Equation (61) also assumes $r_0 = 0$. We found that increasing r_0 causes an approximately multiplicative attenuation that is close to the factor $Q(r_0/r_{\max})$ (Equation (41)), derived for the “Constant” parameterization. Thus, even though the Q function was not derived for the “Exponential” parameterization, we can borrow it to approximate the effect of nonzero r_0 for this parameterization:

$$\text{Exp}_{\text{N-R}} I(x,1) =$$

$$\frac{k_{r_{\max}} k_h \pi m (m/\gamma + 1) \exp(mx)}{2 \sin(\pi m/\gamma)} \times Q(r_0/r_{\max}), \quad (66)$$

where m is given by Equation (62), and γ is given by Equation (65).

Fitting the model to psychophysical data

In this section, we test the accuracy of our equations by comparing their predictions against the true performance of the model, determined from Monte Carlo simulations. In principle, we could make this comparison for any set of model parameters. However, we are most interested in testing the accuracy of our equations when the model’s performance is close to human levels. The Cramér-Rao bound given by the Fisher information can substantially overestimate the decoding precision when the performance level or number of neurons is low (Xie, 2002; May & Solomon, 2014); if the Fisher information deviated substantially from the model’s decoding precision at performance levels shown by humans, then our equations would be of little value. It is therefore important to verify that our equations closely predict the model’s decoding precision for the performance levels shown by observers in psychophysical tasks. The best way to be sure of this is to compare the simulations and equations at human performance levels, i.e. to fit the model to psychophysical data.

Because the model performance can be estimated so quickly from the equations, we used the equations to fit the model to psychophysical data, and then we carried out Monte Carlo simulations using the fitted parameter values. This section therefore serves two purposes: As well as validating our equations for relevant performance levels, it provides a demonstration of how our equations can be used to fit the generic sensory coding model to psychophysical data.

The parameters were fitted to the psychophysical data using the simplex algorithm to minimize the sum of squared differences between predicted and actual log discrimination thresholds. On each iteration of the fitting procedure, the population of neurons was set up as described in Supplementary Appendix C; then the exact Fisher information for each pedestal level was found, using Equation (34) or (36) as appropriate, and this was used to calculate the precision, $\tau(x)$, using Equation (31); the precision was then used to calculate the Weber fraction, W , using Equation (33) with P_θ set to the threshold performance level that had been used in the psychophysical

study; the predicted Weber fraction was multiplied by the pedestal level to give the predicted threshold (Δf_θ in Figure 3, or Δc_θ in Figures 5 and 6). We took the logarithms of these predicted thresholds and found the sum of squared differences between the predicted log thresholds and the log thresholds from the psychophysical data. The simplex algorithm adjusted the model parameters to minimize this sum.

Modelling Weber's Law with Gaussian-tuned neurons: The "Constant Gaussian" parameterization

As noted above, the "Constant" parameterizations give rise to Weber's law. Mayer and Kim's (1986) data on spatial frequency discrimination conform to Weber's Law, and spatial frequency tuning functions in primary visual cortex are approximately Gaussian functions of log spatial frequency (De Valois, Albrecht, & Thorell, 1982), so it is appropriate to fit the "Constant Gaussian" parameterization to Mayer & Kim's data. These data are plotted as black triangles in Figure 3.

In the case of Weber's Law for spatial frequency discrimination, if the pedestal frequency is f_p and the frequency difference at threshold is Δf_θ , then the plot of Δf_θ against f_p on log-log axes will be a straight line of gradient 1. There is only one degree of freedom in this plot – the vertical position – so we needed only one free parameter to fit the data. Our approach was to hold all the model parameters constant except for the density, h , and fit h to the psychophysical data.

For the modelling shown in Figure 3, we set the tuning bandwidth, ω , to 1.5 octaves, which is close to the median value found physiologically (De Valois et al., 1982); q was then calculated from ω using Equation (9). We set r_0/r_{\max} to 0.03 for all the modelling; the rationale for this choice was that Geisler and Albrecht (1997) found that the median r_0 for a 200 ms stimulus was 0.17 for monkey V1 neurons, which is 0.03 when expressed as a proportion of median $r_{\max} = 5.7$ spikes for neurons tuned to the stimulus. Different columns of panels in Figure 3 show results for different combinations of σ_G and r_{\max} (as indicated above each panel in the top row). The values of $\sigma_G = 0.2$ or 0.4 are close to the mean values obtained by Goris et al. (2014) for awake and anaesthetized monkeys, respectively. The lower value of $r_{\max} = 4$ was close to the median value (5.7 spikes) reported by Geisler and Albrecht (1997) for a 200 ms stimulus, and the purpose of

the higher value ($r_{\max} = 16$) was to show how this affects the size of the neuronal correlations and Fano factors.

Having fitted h , we constructed a set of model neurons with z equally spaced along the log spatial frequency axis between $z_{\min} = -0.3$ and $z_{\max} = 1.7$ with spacing

$\delta z = 1/h$. Then we performed Monte Carlo Simulations.

The full details of the Monte Carlo simulations are given in Supplementary Appendix H, but, briefly, they were carried out as follows. First, we sampled a large number of points along the stimulus (x) axis. For each stimulus value, x , we used the stochastic spiking model to generate 10,000 sets of spike counts, and decoded each set of spike counts to give 10,000 estimated stimulus levels. The decoding precision was then calculated as the reciprocal of the variance of the stimulus estimates (decoding precision for the "known gain" decoder is plotted as blue circles in the top row of panels in Figure 3). The stimulus estimates were also used to simulate a 2AFC discrimination task (described in detail in Supplementary Appendix H). 2AFC trials consisted of two stimulus presentations, each with a different randomly generated gain value, g . The lower stimulus value was the pedestal, and the higher value was the target. On each 2AFC trial, the model selected as the target the stimulus with the highest estimated value, and we found the proportion of correct responses for each combination of pedestal and target. For each pedestal, we fitted a Weibull psychometric function (May & Solomon, 2013) to the model's proportion-correct data and obtained a discrimination threshold from the fitted function as described in Supplementary Appendix H. The discrimination thresholds for the "known gain" decoder are plotted as blue circles in the bottom row of panels in Figure 3.

The red lines in the top row of panels give the predicted decoding precision, calculated from the Fisher information, $_{\text{Gauss}}J(x,1)$, using Equations (31) and (36). Table 1 shows that the true decoding precision obtained from the Monte Carlo simulations with known gain differs from the predicted value by less than 0.5%. This close match confirms that the Fisher information gives a sufficiently close approximation of decoding precision to allow us to calculate model performance and gain insights into the relationships between physiology and behaviour.

The red lines in the bottom row of panels show the discrimination thresholds predicted from $_{\text{Gauss}}J(x,1)$ using Relation (33) with $P_\theta = 0.75$. It was these predicted thresholds that were used to fit the model to Mayer & Kim's data in the first place, so it is no surprise that they fit well to Mayer & Kim's data. What is more important is how accurately the thresholds predicted from the Fisher information match those obtained from the Monte Carlo simulations with the "Known Gain" decoder (compare the

red lines against the blue circles in the bottom row of Figure 6).

The thick, grey lines in the top row of Figure 3 plot the precision predicted from the integral approximation of the Fisher information, $\frac{\text{Const}}{\text{Gauss}} I(x,1)$, as defined in Equation (53). The thick, grey lines in the bottom row plot the thresholds predicted from $\frac{\text{Const}}{\text{Gauss}} I(x,1)$. The integral approximation of the Fisher information differs from the true Fisher information by less than 1% (see Table 1), and provides a better insight into the relationship between psychophysical performance and the neuronal parameters than we get from $\frac{\text{Const}}{\text{Gauss}} J(x,1)$, which is a sum with one term for each neuron.

In the Monte Carlo simulations described so far, the decoder knew the gain on each stimulus presentation. We also carried out analogous simulations using the “Univariate” and “Bivariate” decoders, which did not know the gain signal. These decoders were applied to the same simulated spike data as the “Known Gain” decoder – it was just the decoding algorithm that differed. Table 1 shows that, for three of the four conditions, the precision of the “Bivariate” decoder was within 2% of the value predicted from the Fisher information. For the most physiologically plausible condition ($\sigma_G = 0.2$, $r_{\max} = 4$), the “Bivariate” decoder’s performance was not reliably distinguishable from the predicted value with the number of trials that we used; even on the “Bivariate” decoder’s worst condition ($\sigma_G = 0.4$, $r_{\max} = 16$), its precision was within 6% of the value predicted from the Fisher information. This is remarkably close, considering that the Fisher-information-based estimate assumes that the decoder knows the gain signal on each trial and is performing maximum-likelihood estimation; in reality, the “Bivariate” decoder does not know the gain signal, and is not a maximum-likelihood decoder, as it can only take account of pairwise statistical dependencies. The true maximum-likelihood decoder for the “Unknown Gain”

situation would almost certainly yield a precision even closer to the predicted value.

Why should knowledge of the gain signal yield so little benefit? For a Gaussian-tuned neuron, increasing the gain has a similar effect to moving the stimulus value towards the peak of the neuron’s tuning curve. In response to an unknown change of gain, neurons with tuning peaks either side of the true stimulus value will tend to pull the decoded stimulus value in opposite directions, so the effects largely cancel out, and gain fluctuations have little effect on the decoded stimulus values, even when the gain value is unknown to the decoder. Because the gain signal conveys so little extra information, we can relax the assumption that the gain signal is known, and our equations still predict performance closely.

The closeness of the “Bivariate” and “Univariate” decoders’ precision values shows that it is not even particularly important for the decoder to take into account pairwise statistical dependencies. The “Univariate” decoder decodes the neuronal responses as if they were entirely independent of one another, yet, for three of the four conditions, the precision of the “Univariate” decoder was within 1 or 2% of the value predicted from the Fisher information.

We also analyzed the spike counts generated by the model to check that they showed the expected Fano factors and spike count correlations. As already noted, the Fano factor of these model neurons depends on the mean spike rate and the standard deviation of the gain: Equation (14) shows that the plot of Fano factor against mean spike rate should be a straight line, with gradient σ_G^2 , passing through the point (0,1). Figure 4 confirms that our model neurons do show this relationship. Figure 5 confirms that the spike count correlations between pairs of model neurons follow the pattern predicted by Equation (17). The ranges covered by the thick, pink lines in Figures 4 and 5 approximately indicate the ranges of Fano factors and correlations that occur in each parameterization

	$\sigma_G = 0.2, r_{\max} = 4$	$\sigma_G = 0.2, r_{\max} = 16$	$\sigma_G = 0.4, r_{\max} = 4$	$\sigma_G = 0.4, r_{\max} = 16$
$(1 - \sigma_G^2) \times \frac{\text{Const}}{\text{Gauss}} I(x,1)$	1.0067	1.0067	1.0067	1.0067
“Known gain”	0.9957	0.9983	0.9983	0.9996
“Bivariate”	0.9973	0.9863	0.9883	0.9447
“Univariate”	0.9944	0.9833	0.9833	0.9100

Table 1. Precision scores expressed as a proportion of the precision predicted from $_{\text{Gauss}}J(x,1)$ for the “Constant Gaussian” parameterizations of Figure 3. For each of the 104 stimulus levels, x , between 0.6 and 0.8, we expressed the precision for each decoder as a proportion of that predicted from $_{\text{Gauss}}J(x,1)$. The table shows the mean value of this proportion for each condition and decoder. The top row shows the precision predicted from $_{\text{Gauss}}^{\text{Const}}I(x,1)$, expressed as a proportion in the same way. Each value in the table is the mean of 104 ratios. For each ratio, the precisions were calculated from 10,000 trials, except for the “Bivariate” decoder on the two conditions with $r_{\text{max}} = 4$. These conditions had a very large number of neurons, so the “Bivariate” decoding algorithm, which calculated a likelihood for every pair of neurons, was very slow. Due to time constraints, we were only able to decode 4,200 of the 10,000 trials for $\sigma_G = 0.2$, $r_{\text{max}} = 4$, and 3,500 of the 10,000 trials for $\sigma_G = 0.4$, $r_{\text{max}} = 4$. The slightly higher precision score for the “Bivariate” than “Known Gain” decoder for $\sigma_G = 0.2$, $r_{\text{max}} = 4$ is a result of sampling error due to the smaller number of trials.

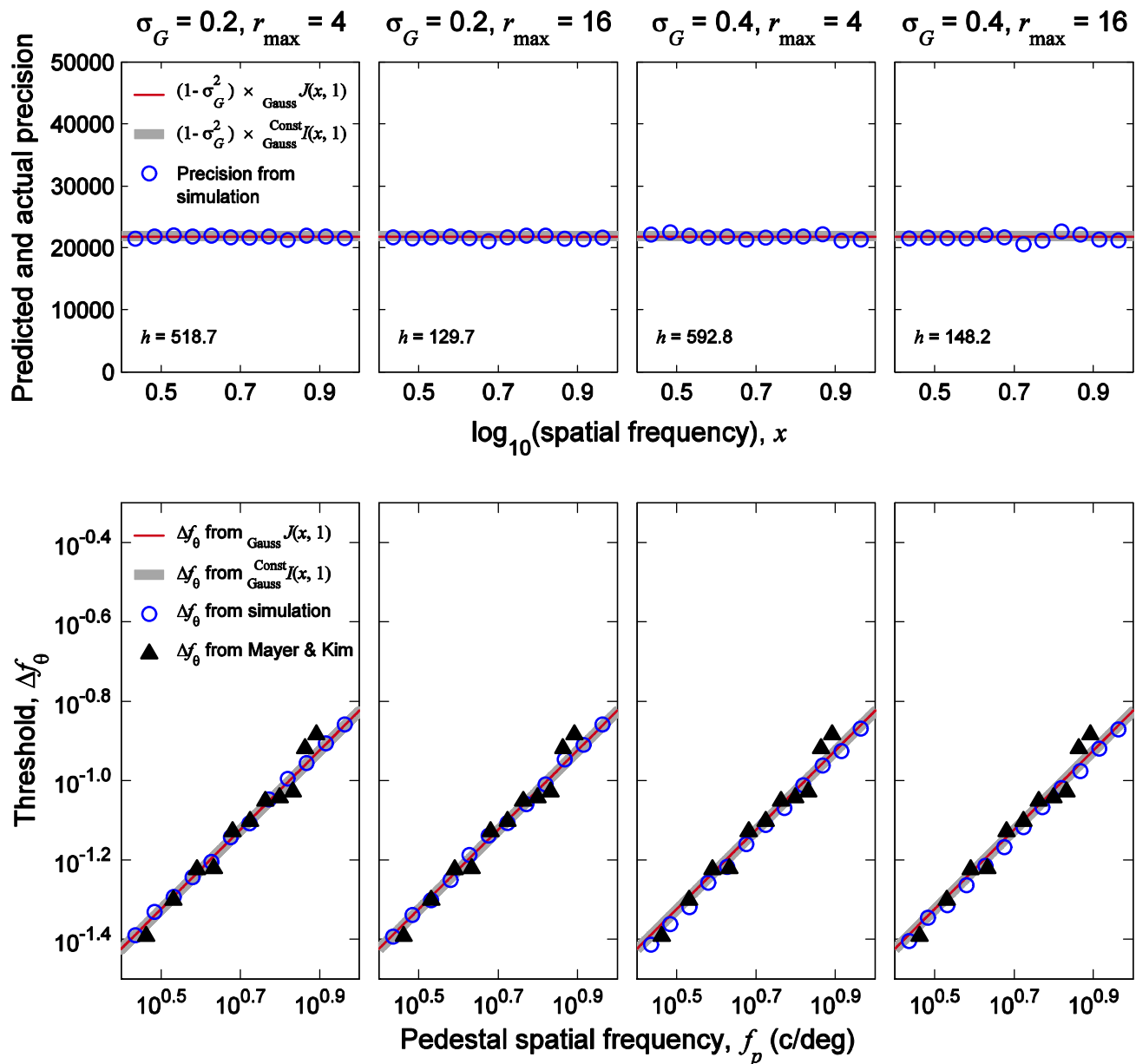


Figure 3. Fitting the “Constant Gaussian-tuned” parameterization to data from Mayer and Kim (1986). The black triangles in the bottom panels show Mayer and Kim’s (1986) spatial frequency discrimination data for their subject MJ, condition R 2PFC, transcribed from their Figure 7. Mayer and Kim’s data actually show the difference in spatial frequency between the 0.25 point and 0.75 point of the psychometric function. The “pedestal” frequency should fall halfway between these points, so we halved their thresholds to obtain Δf_θ , the difference between target and pedestal at a threshold performance level of $P_\theta = 0.75$. Four of the model’s parameters were fixed as follows: $z_{\min} = -0.3$, $z_{\max} = 1.7$, $r_0/r_{\max} = 0.03$, $\omega = 1.5$ octaves. r_{\max} and σ_G were set as indicated above the top of each panel in the top row. The remaining parameter, h , was fitted to Mayer and Kim’s data; the fitted values are shown in the top panels. Having fitted this parameter, we carried out Monte Carlo simulations as described in the text and in Supplementary Appendix H. The blue circles in the top row of panels plot the decoding precision from the Monte Carlo simulations using the “Known Gain” decoder. These points show decoding precision for every 30th value along the x -axis that we calculated. The rest are omitted from the figure for clarity, although the stimulus estimates for these x -values were used in the 2AFC simulations, where we needed a fairly fine sampling of the x -axis to fit the psychometric function, and hence find the discrimination threshold. The red lines in the top row of panels plot the precision predicted from the Fisher information, ${}_{\text{Gauss}}J(x,1)$ (Equation (36)), while the thick, grey lines plot the precision predicted from the integral approximation of the Fisher information, ${}_{\text{Gauss}}^{\text{Const}}I(x,1)$ (Equation (53)); in both cases, we converted ω to q , using Equation (9), before applying equation (36) or (53). The predicted precision is found by multiplying the Fisher information by $1 - \sigma_G^2$ (see Equation (31)). In the bottom row, the blue circles plot the thresholds, Δf_θ , obtained from the Monte Carlo simulations of the 2AFC discrimination task with the “Known Gain” decoder. The red lines in the bottom row plot the thresholds predicted from ${}_{\text{Gauss}}J(x,1)$ using Relation (33) with $P_\theta = 0.75$. The thick, grey lines plot the thresholds predicted in the same way, except using ${}_{\text{Gauss}}^{\text{Const}}I(x,1)$ to approximate the Fisher information. Note that the abscissas in the top row are identical to those on the bottom row, i.e. each position on the abscissa in the top row represents the same stimulus as the same position on the abscissa in the bottom row. On the top row, we have marked the abscissas with log units since the precision is calculated from the values in these units; in the bottom row we have marked the abscissas with linear units, to be compatible with the threshold, which is defined as the difference of spatial frequencies.

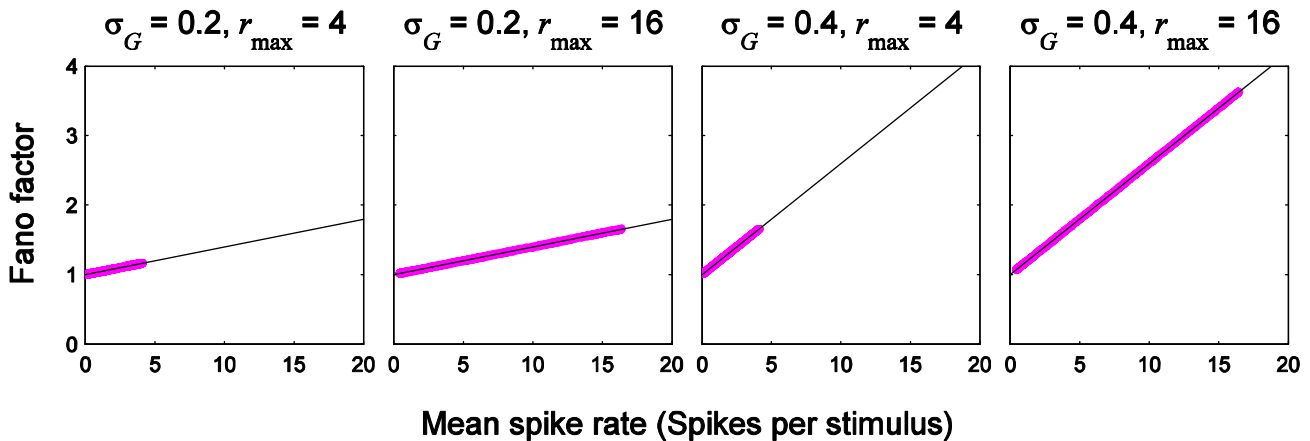


Figure 4. Fano factors of the neurons in the modelling of Figure 3. Each panel in this figure shows the Fano factors for the parameterization in the corresponding column of panels in Figure 3. For each model neuron, and each stimulus level, we found the mean and variance of the spike count across the 10,000 repetitions. We divided the variance by the mean to give the Fano factor (we had to exclude from this analysis combinations of neuron and stimulus level for which the neuron never fired). Because of the large number of different combinations of neuron and stimulus level ($> 10^5$ in each panel), there were too many points to plot as individual dots, so we sorted the points in order of mean spike rate, and then chunked them into groups of 1,000. The thick, pink lines plot the average mean spike rate against the average Fano factor for each group. The black line in each plot is the predicted relationship between mean and Fano factor given by Equation (14).

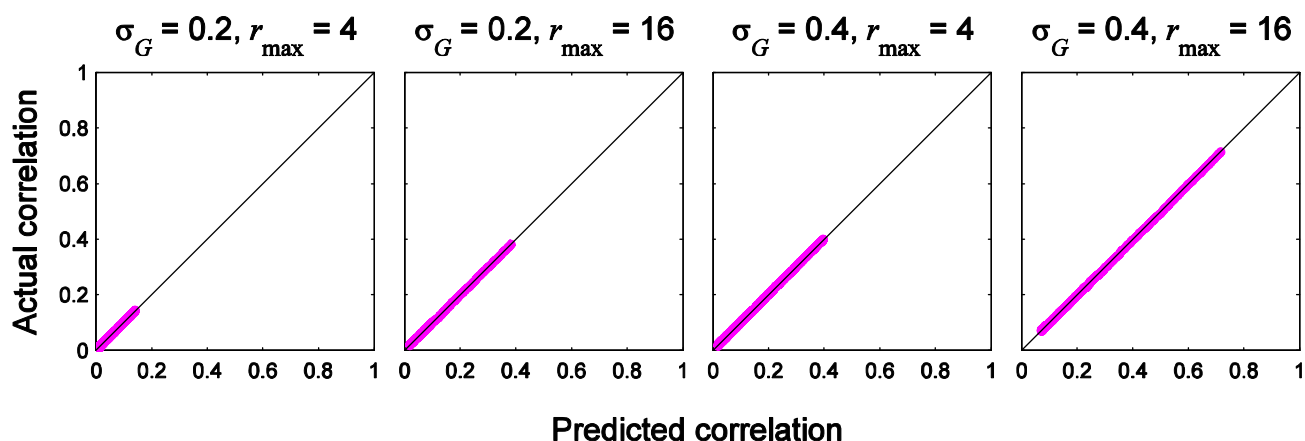


Figure 5. Spike count correlations between pairs of neurons in the modelling of Figure 3. Each panel in this figure shows the correlations for the parameterization in the corresponding column of panels in Figure 3. Because of the large number of pairs of neurons, we just analyzed the responses to the stimulus level in the middle of the range of stimuli used in the modelling. For each pair of model neurons, we found the Pearson correlation between the spike counts of the two neurons across the 10,000 repetitions (we omitted from this analysis all pairs of neurons for which either neuron gave no spikes across all 10,000 repetitions). We then sorted the data points in order of the predicted correlation given by Equation (17), and chunked them into groups of 1,000. The thick, pink lines plot the mean predicted correlation against the mean actual correlation for each group. The black lines plot the outcome that would occur if the actual correlation were always exactly equal to the predicted correlation.

Modelling Weber’s Law for suprathreshold contrast discrimination: The “Constant Naka-Rushton” parameterization

The “Constant” parameterization yields Weber’s Law, and the Naka-Rushton function is the tuning function for contrast. Thus, it is appropriate to fit the “Constant Naka-Rushton” parameterization to the data from a contrast discrimination experiment that gave rise to Weber’s Law. Although contrast discrimination generally shows a near-miss to Weber’s Law (Legge, 1981; Meese et al., 2006), there have been some reports of Weber’s Law for contrast discrimination (Swift & Smith, 1983; Bird, Henning, & Wichmann, 2002). We fitted the model to some of Bird et al.’s (2002) data (which are plotted as black triangles in Figure 6), and carried out simulations analogous to those in Figure 3, but using the Naka-Rushton tuning function instead of the Gaussian. The model parameters are given in the caption of Figure 6. In this simulation, we only considered psychophysical data from pedestals that were clearly suprathreshold: Very low contrasts would fall below the range of semi-saturation contrasts found physiologically, and would not stimulate many neurons, and, in these circumstances, the Fisher information is unlikely to

be close to the decoding precision, and cannot be used for estimating the model’s psychophysical performance.

The blue circles in Figure 6 show the “Known Gain” decoder’s decoding precision and discrimination thresholds; these are very well predicted from both the true Fisher information expressions (thin, red lines in Figure 6) and the integral approximations (thick, grey lines). Table 2 shows that, in each case, the true decoding precision is within 2 or 3% of that predicted from the Fisher information. The integral approximation of the Fisher information is exceptionally close to the true Fisher information.

Table 2 shows that, unlike with Gaussian tuning functions, both “Unknown Gain” decoders were much worse than the “Known Gain” decoder. The “Bivariate” decoder does better than the “Univariate” decoder but, in both cases, the gain fluctuations have a catastrophic effect on decoding precision. It is easy to see why this happens. For any Naka-Rushton-tuned neuron, an increase in gain has the same effect as an increase in contrast, so, if the gain is unknown, then random gain fluctuations will be interpreted as fluctuations in contrast, and, on each stimulus presentation, the decoded contrast will be biased in the direction of the gain value on that stimulus presentation, leading to inaccurate estimation.

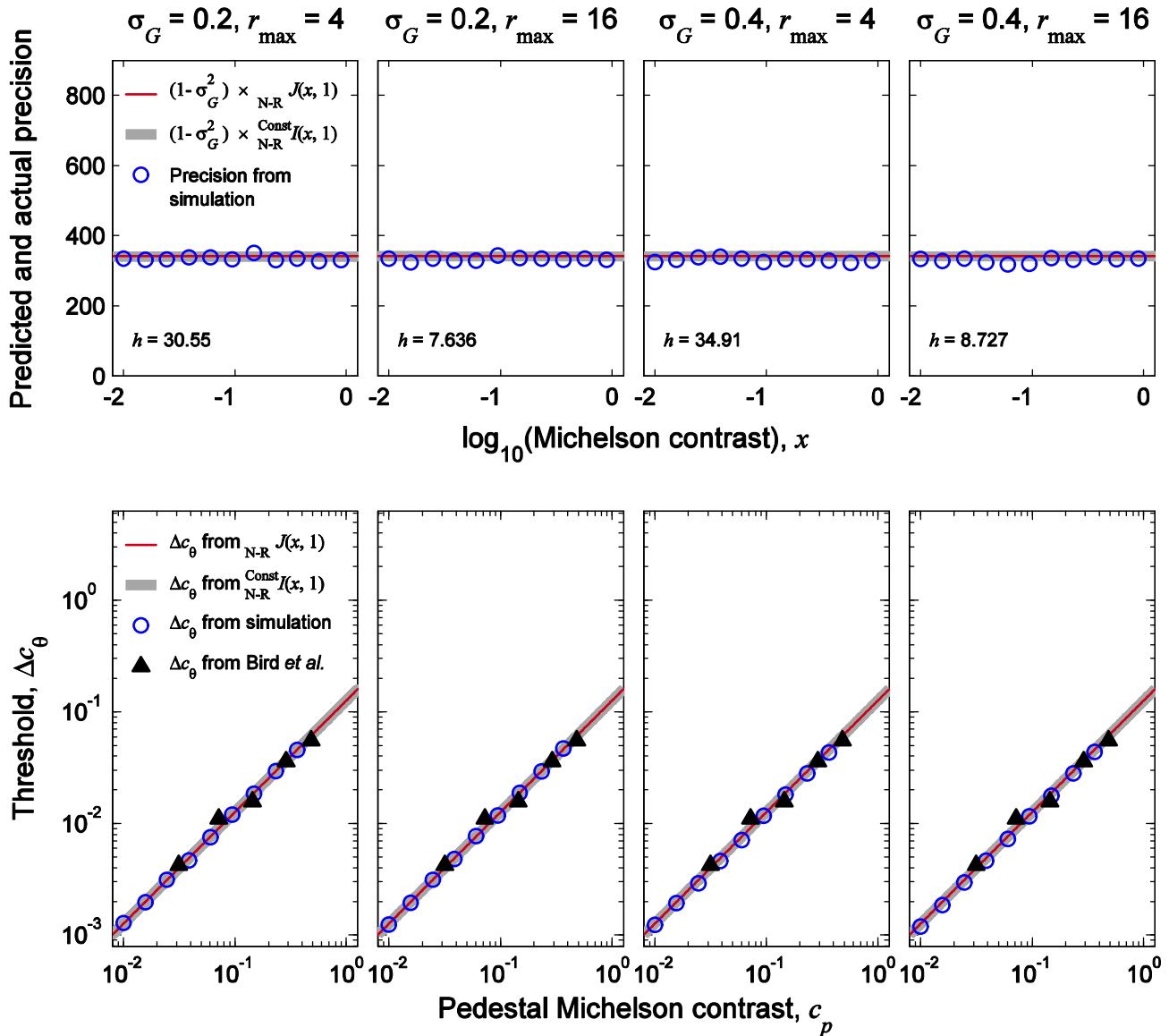


Figure 6. Fitting the “Constant Naka-Rushton” parameterization to data from Bird et al. (2002). The black triangles in the bottom panels show Bird et al.’s (2002) contrast discrimination data for their subject GBH, for 4.19 c/deg sine wave gratings, transcribed from their Figure 3b. Four of the model’s parameters were fixed as follows: $q = 3$, $r_0/r_{\max} = 0.03$, $z_{\min} = -3$, $z_{\max} = 1$ ($q = 3$ is close to the mean value found physiologically – see May & Solomon, 2014, Table 1). r_{\max} and σ_G were set as indicated above the top of each panel in the top row. The remaining parameter, h , was fitted to Bird et al.’s data; the fitted values are shown in the top panels. Monte Carlo simulations were conducted in a similar way to those in Figure 3, except using the Naka-Rushton tuning function. All plotting conventions are the same as or analogous to those in Figure 3. The blue circles plot Monte Carlo simulation results for every 18th value along the x -axis that we calculated (the rest are omitted from the figure for clarity, although the stimulus estimates for these x -values were used for fitting psychometric functions in the 2AFC simulations). The red lines plot performance predicted from the true Fisher information, ${}_{\text{N-R}}J(x, 1)$ (Equation (34)), while the thick, grey lines plot performance predicted from the integral approximation of the Fisher information, ${}_{\text{N-R}}^{\text{Const}}I(x, 1)$ (Equation (40)). Precision was predicted by multiplying the Fisher information by $1 - \sigma_G^2$ (see Equation (31)). Predicted discrimination threshold was derived from the predicted precision using Relation (33) with $P_{\theta} = 0.75$.

	$\sigma_G = 0.2, r_{\max} = 4$	$\sigma_G = 0.2, r_{\max} = 16$	$\sigma_G = 0.4, r_{\max} = 4$	$\sigma_G = 0.4, r_{\max} = 16$
$(1 - \sigma_G^2) \times \frac{\text{Const}}{\text{N-R}} I(x, 1)$	1.000002	1.000002	1.000002	1.000003
“Known gain”	0.9804	0.9782	0.9738	0.9736
“Bivariate”	0.3897	0.4852	0.09312	0.2162
“Univariate”	0.3574	0.3776	0.06974	0.09887

Table 2. Precision scores expressed as a proportion of the precision predicted from $\frac{\text{N-R}}{\text{N-R}} J(x, 1)$ for the “Constant Naka-Rushton” parameterizations of Figure 6. For each of the 139 stimulus levels, x , between -1.5 and -0.5 , we expressed the precision for each decoder as a proportion of that predicted from $\frac{\text{N-R}}{\text{N-R}} J(x, 1)$. The table shows the mean value of this proportion for each condition and decoder. The top row shows the precision predicted from $\frac{\text{Const}}{\text{N-R}} I(x, 1)$, expressed as a proportion in the same way. Each value in the table is the mean of 139 ratios. For each ratio, the precisions were calculated from 10,000 trials.

Modelling the near-miss to Weber’s Law for contrast discrimination: The “Exponential Naka-Rushton” parameterization

The black triangles in the bottom row of Figure 7 show human contrast discrimination data from Meese et al. (2006). The log-log slope is clearly shallower than that obtained by Bird et al., plotted in Figure 6. We fitted the “Exponential Naka-Rushton” parameterization to Meese et al.’s data by using the Fisher information to predict the model’s thresholds, as described earlier, and performed similar Monte Carlo simulations (see Supplementary Appendix H). The psychophysical data have two degrees of freedom: the height and slope of the log-log plot of threshold against pedestal. Thus, we now needed two degrees of freedom in our model fit. In all the parameterizations in Figure 7, we fitted the parameter k_h , which gives the density for a stimulus value of $x = 0$. Each column of Figure 7 uses a different choice of which other parameter(s) to fit. In column (A), we fitted m_h , so that h varied with z , and set the other m -parameters ($m_{r_{\max}}$ and m_q) to zero, so that r_{\max} and q were constant across the neurons (equal to $k_{r_{\max}}$ and k_q , respectively). In column (B), we fitted $m_{r_{\max}}$, so that r_{\max} varied with z , and set the other m -parameters (m_h and m_q) to zero. In column (C), we fitted m_q , so that q varied with z , and set the other m -parameters (m_h and $m_{r_{\max}}$) to zero. In column (D), we fitted all three m -parameters, subject to the constraint that $m_h = m_{r_{\max}} = m_q$; thus, there were still only two degrees of freedom in the fit, but all three parameters, h , r_{\max} and q , varied with z . In each parameterization, all the parameters that were not fitted were set to reasonable values (see the caption of Figure 7 for these values).

Note that the individual fitted m -parameter values in column (D) are approximately one third of the values ob-

tained when only one of the three parameters was fitted, indicating that the contribution of all three of these parameters to the Fisher information is additive. The precise additivity of m_h and $m_{r_{\max}}$ was proved analytically earlier, and the fitting results in column (D) suggest that this additivity also applies to m_q , at least approximately.

The plotting conventions in Figure 7 are the same as in Figure 6. The top row shows predicted and actual precision, while the bottom row shows 2AFC discrimination thresholds. Blue circles show the precision and thresholds from the Monte Carlo simulations with the “Known Gain” decoder; thin, red curves show the precision and thresholds predicted from the true Fisher information, $\frac{\text{N-R}}{\text{N-R}} J(x, 1)$ (Equation (34)); thick, grey curves show the precision and thresholds predicted from the integral approximation of the Fisher information, $\frac{\text{Exp}}{\text{N-R}} I(x, 1)$ (Equation (66)). Thresholds were predicted from the Fisher information using Relation (33) with $P_\theta = 1 - 0.5/e = 0.816\dots$, which was the performance level that defined the threshold in Meese et al.’s study. These threshold predictions map out almost straight lines on the log-log plots, which fit well to Meese et al.’s data.

Recall that the derivation of $\frac{\text{Exp}}{\text{N-R}} I(x, 1)$ was exactly correct only when $r_0 = m_q = 0$ (i.e. zero spontaneous firing rate, and q constant with respect to z). When either $r_0 \neq 0$ or $m_q \neq 0$ (as is the case in each panel of Figure 7), the integral was intractable, so we used workarounds to give an approximate expression. Nevertheless, Table 3 shows that the precision predicted from the integral approximations of the Fisher information never differed by more than about 6% from that predicted from the true Fisher information. Meanwhile, the actual precision from the “Known Gain” decoder was never more than 6% lower than that predicted from the Fisher information. As with our other simulations of contrast discrimination, the performance of the “Unknown Gain” decoders was much worse than that of the “Known Gain” decoder.

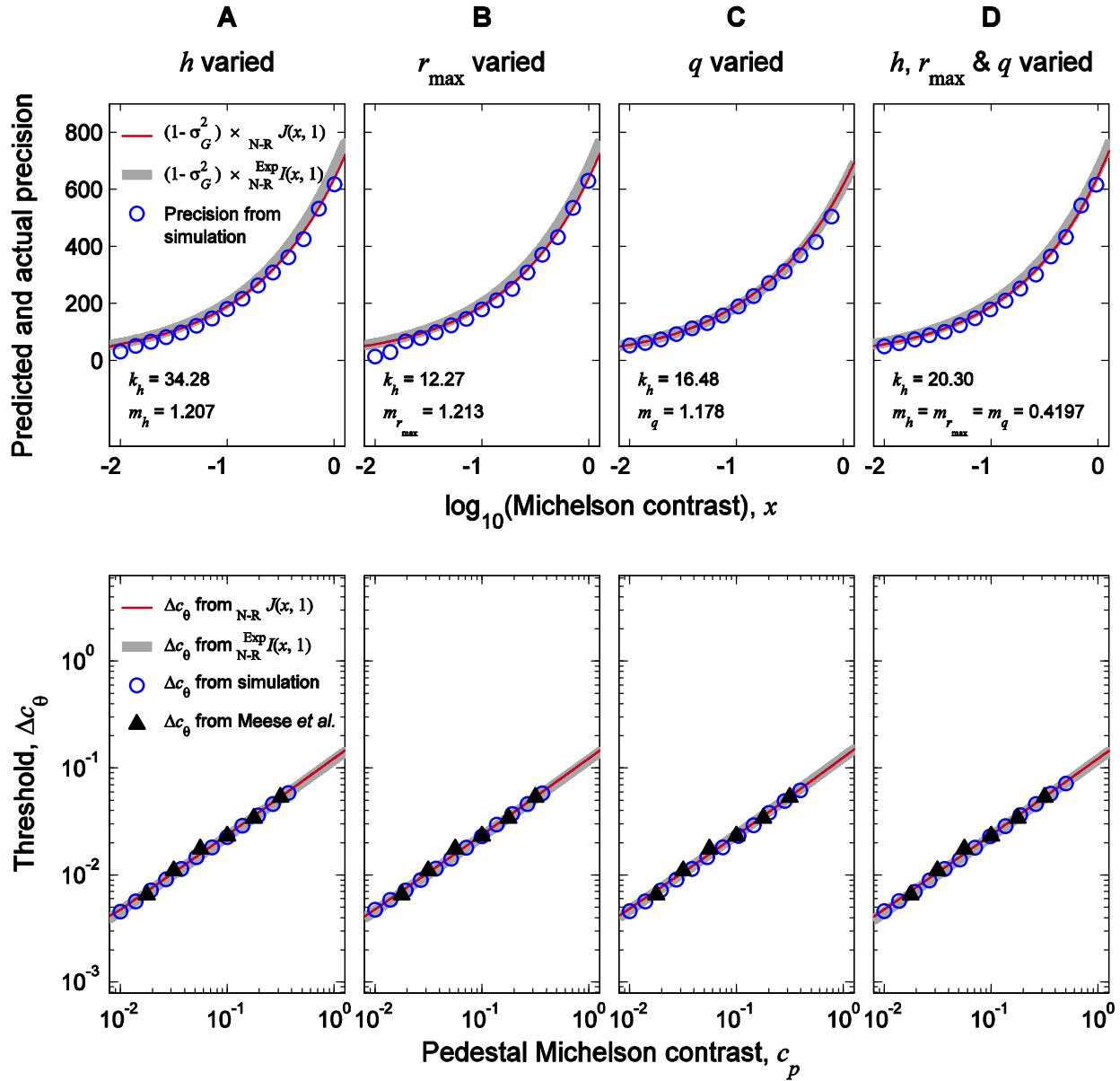


Figure 7. Fitting the “Exponential Naka-Rushton” parameterization to data from Meese et al. (2006). Meese et al.’s data are plotted as black triangles in the bottom row of this figure (these data are from Meese et al.’s Binocular condition, plotted as squares in their Figure 5, and kindly supplied by Tim Meese). Each column of panels gives the data for one parameterization. In each case, $\sigma_G = 0.2$, $r_0/r_{\max} = 0.03$, $z_{\min} = -3$, and $z_{\max} = 1$. We fitted two parameter values to the data (fitted values are given in the top panel), and chose reasonable values for the others. One of the fitted parameters was always k_h . The other fitted and fixed parameters in the different parameterizations were as follows. (A) m_h was fitted, so that h varied with z , and we set $m_{r_{\max}} = m_q = 0$, $k_{r_{\max}} = 5.7$, and $k_q = 3$. (B) $m_{r_{\max}}$ was fitted, so that r_{\max} varied with z , and we set $m_h = m_q = 0$, $k_{r_{\max}} = 16$, and $k_q = 3$. (C) m_q was fitted, so that q varied with z , and we set $m_h = m_{r_{\max}} = 0$, $k_{r_{\max}} = 5.7$, and $k_q = 7$. (D) m_h , $m_{r_{\max}}$ and m_q were fitted, subject to the constraint that $m_h = m_{r_{\max}} = m_q$, so that h , r_{\max} and q varied with z , and we set $k_{r_{\max}} = 8$, and $k_q = 4$. Plotting conventions are the same as in Figure 6. $\text{N-R } J(x, 1)$ was calculated using Equation (34), and $\text{N-R } \text{Exp } I(x, 1)$ was calculated using Equation (66). Predicted discrimination threshold was predicted from each Fisher information expression using Relation (33) with $P_\theta = 1 - 0.5/e = 0.816\dots$. Blue circles plot performance for every 18th stimulus level that we evaluated.

	h varied	r_{\max} varied	q varied	h, r_{\max} & q varied
$(1 - \sigma_G^2) \times \text{Exp}_{\text{N-R}} I(x, 1)$	1.0602	1.0606	0.9848	1.0558
“Known gain”	0.9407	0.9416	0.9544	0.9509
“Bivariate”	0.4605	0.4575	0.3623	0.4538
“Univariate”	0.4510	0.4385	0.3302	0.4334

Table 3. Precision scores expressed as a proportion of the precision predicted from $\text{N-R } J(x, 1)$ for the “Exponential Naka-Rushton” parameterizations of Figure 7. For each of the 191 stimulus levels, x , between -1.5 and -0.5 , we expressed the precision for each decoder as a proportion of that predicted from $\text{N-R } J(x, 1)$. The table shows the mean value of this proportion for each condition and decoder. The top row shows the precision predicted from $\text{Exp}_{\text{N-R}} I(x, 1)$, expressed as a proportion in the same way. Each value in the table is the mean of 191 ratios. For each ratio, the precisions were calculated from 10,000 trials.

Discussion

The purpose of this study was to construct psychophysical models whose performance could easily be connected to the parameters of the neurons that encode the stimulus. We used a parameterization of Goris et al.’s (2014) model of neuronal spiking, in which each neuron has an independent Poisson spiking process, and the spike rate of each neuron is modulated by a multiplicative, gamma-distributed gain signal that is shared between all the neurons. The individual neurons were characterized by the parameters of their tuning function: r_0 (spontaneous spike rate), r_{\max} (maximum increment in spike rate from r_0), q (tuning sharpness), and z (position along the stimulus axis). Tuning functions could be sigmoidal (Naka-Rushton) or Gaussian. The population of neurons was characterized by the density, h , of neurons along the stimulus axis. The gain fluctuations (parameterized by the standard deviation, σ_G) had both neuron-specific and population-wide effects. The neuron-specific effects of the gain fluctuations were the Fano factors (Figure 4), and the population-wide effects were the spike count correlations that resulted from having the neurons share the same gain signal (Figure 5); both effects arise from Equation (12), which describes the covariance matrix for a population of model neurons of this kind.

Model performance was calculated analytically by using the neuronal parameters to calculate the Fisher information, from which we can estimate the decoding precision. Although one can always find the performance level by setting up a Monte Carlo simulation (Clatworthy, Chirimuuta, Lauritzen, & Tolhurst, 2003; Chirimuuta, Clatworthy, & Tolhurst, 2003; Chirimuuta & Tolhurst, 2005), the long time required to complete the simulations

for a single parameter set makes it difficult to fit the model to data (Chirimuuta & Tolhurst, 2005) and this method provides little insight into why the observed patterns of results were obtained, or whether the results generalize to other parameterizations of the model. Our work solves all of these problems by providing fairly simple equations that can be evaluated to give a close approximation of the decoding precision, and thus discrimination threshold.

Our expressions for the decoding precision were derived assuming that the decoder knows the gain signal, g . In this case, we can derive the Fisher information as if the neurons were statistically independent; we then take the Fisher information for the case of $g = 1$ and multiply it by $1 - \sigma_G^2$ to obtain the predicted decoding precision (Equation (31)). This predicted decoding precision is the modal value of the Fisher information across all stimulus presentations. As discussed later, with Gaussian tuning curves, we can relax the assumption that the decoder knows the gain signal, and our equations still provide an accurate prediction of the model’s performance.

We derived two kinds of expression for the Fisher information. One was an exact expression (represented by the letter J), which consists of a sum with one term for each neuron. The other kind of expression (represented by the letter I) approximates this sum using an integral. These integral approximations are much more compact, and can help to shed light on the relationships between psychophysical performance and the neuronal parameters.

We outlined two basic types of parameterization of the generic sensory coding model: the “Constant” and “Exponential” parameterizations. In the “Constant” parameterization, all the neuronal parameters are constant with respect to z , and z is distributed with constant density along the stimulus axis. Our integral approximations revealed some particularly simple relationships between Fisher information and the neuronal parameters for the “Constant”

parameterization. For both Naka-Rushton and Gaussian tuning functions, if r_0/r_{\max} is held constant, the Fisher information is proportional to $r_{\max}qh$ (see Equation (40) for the Naka-Rushton function and Relation (50) for the Gaussian tuning function). For the Naka-Rushton tuning function, the Fisher information is proportional to a decreasing function of the relative spontaneous firing rate, r_0/r_{\max} , which we call Q (see Equation (41) and Figure 2); as r_0 increases, the Fisher information undergoes a multiplicative attenuation that is a function *only* of the ratio r_0/r_{\max} . The same effect holds to a very close approximation for the Gaussian tuning function (see Supplementary Appendix F). Another feature of the “Constant” parameterization is that the Fisher information is constant across the stimulus axis (i.e. it is independent of x). Because of this, if x is the logarithm of the physical stimulus value, then performance of the “Constant” parameterization will obey Weber’s Law (discrimination threshold proportional to pedestal). We used the exact Fisher information expressions to fit the “Constant” parameterizations of the model to real psychophysical data that conformed to Weber’s Law: Mayer and Kim’s (1986) spatial frequency discrimination data (Figure 3) and Bird et al.’s (2002) suprathreshold contrast discrimination data (Figure 6). In all cases, the thresholds predicted from the Fisher information expressions gave excellent matches to the thresholds obtained from Monte Carlo simulations.

Many studies of suprathreshold contrast discrimination have found a near-miss to Weber’s Law, where the plot of threshold against pedestal is a straight line with a slope of about 0.6-0.7 on a log-log plot. We showed that this could be accounted for by allowing any of the r_{\max} , q , or h parameters to vary exponentially with z . The rates of increase were determined by parameters $m_{r_{\max}}$, m_q and m_h , respectively. We call this the “Exponential” parameterization, and it is a generalization of the “Constant” parameterization (The “Constant” parameterization is the “Exponential” parameterization with $m_{r_{\max}} = m_q = m_h = 0$). Again, the integral approximation of the Fisher information (Equation (66)) revealed two features that were not explicit in the exact expressions. Firstly, the Fisher information for the “Exponential” parameterization is an exponential function of the stimulus level: It is proportional to $\exp(mx)$, where $m = m_h + m_{r_{\max}}$. Secondly, the Fisher information is a function of the sum of $m_{r_{\max}}$ and m_h , regardless of their individual values. We also found that, as with the “Constant” parameterizations, we could closely model the effect of r_0 by multiplying the Fisher information by

$Q(r_0/r_{\max})$ as defined in Equation (41). We used the exact Fisher information expressions to fit the “Exponential Naka-Rushton” parameterization to Meese et al.’s (2006) contrast discrimination data. The thresholds predicted from the Fisher information expressions gave a good match to the thresholds obtained from Monte Carlo simulations.

The fact that we needed an exponential increase in at least one neuronal parameter with increasing semi-saturation contrast shows that the physiological constraints imposed by the near-miss to Weber’s law are quite different from those imposed by the true Weber’s Law, which results if all the neuronal parameters are constant. This fact seems not to have been widely appreciated, because contrast discrimination performance is usually modelled using transducer models with additive noise (e.g., Wilson, 1980; Legge & Foley, 1980; Meese et al., 2006), and the transducer only needs a slight tweak to alter its predictions from Weber’s Law to the near-miss.

“Implicit” decorrelation when the gain is known

In our model, if the decoder knows the gain signal, it can express the neuronal spike distributions as independent Poisson distributions. This is a form of decorrelation, but the neurons’ responses are not changed, and are therefore not explicitly decorrelated. The neuronal responses are correlated across all trials, but are uncorrelated within each subset of trials that share the same gain signal. The decoder’s knowledge of the gain signal allows it to identify which uncorrelated subset of trials the current trial belongs to, so the neuronal spiking distributions, conditioned on this knowledge, are statistically independent. The neurons have thus been decorrelated “implicitly” by virtue of the decoder’s knowledge of what caused the correlations. Given the finding that a large proportion of neuronal variability is explained by fluctuating internal gain signals that can be shared between neurons (Ecker et al., 2014; Goris, Movshon, & Simoncelli, 2014), it could be that many apparently correlated populations of neurons are implicitly uncorrelated to a large extent.

Effect of correlations when the gain is unknown

Although knowledge of the gain signal can result in an implicit decorrelation of the neuronal responses, our simulations with the Gaussian tuning curve show that it is not always necessary to know the gain in order to achieve

near-optimal performance. The “Bivariate” decoder does not know the gain signal, and expresses each pair of neuronal responses as a correlated gamma-Poisson mixture distribution. Nevertheless, its decoding performance is nearly as good as that of the “Known Gain” decoder. Furthermore, performance was not much worse with the “Univariate” decoder, which knows nothing whatever about the correlation structure of the population. This is highly convenient, as it means that, for Gaussian tuning curves, our analytical expressions for the model’s performance are not restricted to the class of models for which the decoder knows the gain signal or even knows about the pairwise correlations.

With the Naka-Rushton tuning function, a decoder that does not know the gain signal cannot distinguish changes in gain from changes in stimulus contrast, so the gain fluctuations greatly impair performance. Because of this, for Naka-Rushton tuning curves, our analytical expressions for the model’s decoding performance only apply to the “Known Gain” decoder.

So far, we have explained this difference between Gaussian and Naka-Rushton tuning curves intuitively: For the Naka-Rushton function, a change of gain will bias all the neurons’ likelihood functions in the same direction, leading to inaccurate decoding, whereas, for the Gaussian tuning function, neurons with tuning peaks either side of the true stimulus value will have their likelihood functions biased in opposite directions, so the effect of a gain fluctuation cancels out, and the stimulus estimate is largely unaffected. We can gain a more formal insight into this difference between the Gaussian and Naka-Rushton tuning functions by considering the Fisher information for decoding a pair of our model neurons that share a gain signal unknown to the decoder. In this two-neuron system, the likelihood function is given by Equation (22). Supplementary Appendix I shows that the Fisher information for decoding a stimulus of value x in this case is given by

$$J = \frac{(r_1'(x))^2}{r_1(x)} + \frac{(r_2'(x))^2}{r_2(x)} - \frac{(r_1'(x) + r_2'(x))^2}{r_1(x) + r_2(x) + 1/\sigma_G^2}, \quad (67)$$

where $r_1(x)$ and $r_2(x)$ are the tuning functions of the two neurons, and $r_1'(x)$ and $r_2'(x)$ are their first derivatives. If the tuning functions are Gaussian, then the slopes, $r_1'(x)$ and $r_2'(x)$, will often be opposite in sign, and will partially or completely cancel out in the third (subtractive) term of Equation (67). When x falls exactly midway between the peaks of two identically shaped Gaussian tuning functions, the third term is zero, and the Fisher information is the same as for two independent Poisson-spiking neurons (i.e. equivalent to $\sigma_G = 0$); this gives an insight into why gain fluctuations have so little effect on decoding

Gaussian-tuned neurons, even when the gain signal is unknown. On the other hand, the slopes of the Naka-Rushton functions are always positive, and so they never cancel out in this way, so the third term of Equation (67) generally subtracts more from the Fisher information for Naka-Rushton-tuned neurons than it does for Gaussian-tuned neurons.

It is important not to read too much into this analysis, because the Fisher information can be a poor estimator of decoding precision when the number of neurons is small (Xie, 2002; May & Solomon, 2014; also see the discussion of this issue in Supplementary Appendix I). The key point is that, for Gaussian tuning functions, the Cramér-Rao upper bound on the decoding precision can be independent of the gain variance, whereas, for Naka-Rushton tuning functions, the upper bound on the precision will always decrease with increasing gain variance.

Comparisons with other studies

Sanborn and Dayan (2011) argued that the central puzzle for contrast discrimination at high contrast is that the near-miss to Weber’s Law for contrast discrimination is difficult to reconcile with neuronal noise. They explained the near-miss to Weber’s Law using a model in which the neuronal noise was Gaussian with response variance that was nearly constant for low mean response levels, and nearly proportional to the mean response for high mean response levels; they called this the “hinge noise” model. They set up a model in which the stimulus was encoded by a population of neurons with this form of noise. Each neuron had a linear contrast-response function, and the neurons differed from each other in their sensitivity to the stimulus orientation. Sanborn and Dayan showed that optimal decoding of this population gave rise to a contrast discrimination function with a log-log slope of 0.84 at high contrasts, i.e. a near-miss to Weber’s Law.

A weakness of Sanborn and Dayan’s model is that, although it attempts to reconcile psychophysical and physiological findings, none of the key elements of their model are physiologically plausible. Firstly, the function mapping mean response to variance in their hinge noise model is quite unlike neuronal noise, which is proportional to the mean for low responses and proportional to the square of the mean for high responses, a relationship that is well captured by the gamma-Poisson mixture distribution that we used (see Goris et al., 2014). Secondly, Sanborn and Dayan’s linear contrast-response function is quite different from those of real neurons, which are well described by the Naka-Rushton function (Albrecht & Hamilton, 1982). Thirdly, although Sanborn and Dayan’s model had a range of orientation channels, it had only one contrast channel.

In the visual cortex, neurons tend to be sensitive to changes in contrast over a relatively narrow contrast range, which suggests that the contrast code is distributed across a population of neurons with different semi-saturation contrasts (Teo & Heeger, 1994; Clatworthy et al., 2003; Chirimuuta et al., 2003). Our work shows that, if we adopt this more physiologically plausible notion, then the “central puzzle” raised by Sanborn and Dayan disappears: The effect of the pedestal on threshold depends, not just on the form of noise, but also on the distribution of the contrast-response functions along the contrast axis.

Our approach of coding contrast across a range of model neurons with different semi-saturation contrasts was used by Chirimuuta and Tolhurst (2005) to model contrast discrimination, and indeed our work was originally inspired by theirs. However, Chirimuuta and Tolhurst lacked an analytical description of their model’s performance. This seriously hampered their attempts to fit their model to psychophysical data, as they were only able to try a small number of model parameter sets due to the slowness of the Monte Carlo simulations. In their Figure 9A, they used a model with 8 Naka-Rushton-tuned neurons. Each neuron had $q = 2$, and there was additionally a threshold applied to the output of the Naka-Rushton function. For each of the 8 neurons, r_{\max} and z were free parameters, and Chirimuuta and Tolhurst adjusted these 16 parameters by hand to fit their psychophysical data. Our analytical approach allows us to reduce the parameter set, and more importantly, to understand exactly what contribution each parameter makes to the decoding precision. We can then fix most parameters to physiologically plausible values, and adjust no more parameters than we need to fit the data (i.e. 1 parameter for fitting Weber’s Law and 2 parameters for fitting the near-miss to Weber’s Law).

Our general approach to modelling psychophysical performance also has much in common with that of Goris et al. (2013) and Itti, Koch and Braun (2000). In both cases, the stimulus was encoded over a population of model neurons, which were assumed to be decoded in a way that was optimal (or quasi-optimal) for the psychophysical task. The emphasis in both of those studies was to explain a large set of data using a single model. The emphasis in our study has been to derive mathematical relationships between the neuronal response properties and characteristics of performance.

The distribution of neurons along the stimulus axis

Although our “Constant” and “Exponential” parameterizations of the model with Naka-Rushton tuning functions are successful in giving rise to, respectively, Weber’s Law and the near-miss to it, one might object that these models cannot be right because the semi-saturation contrasts measured in physiological experiments have neither a constant, nor exponential, distribution on the log-contrast axis – the distribution is close to Gaussian (Clatworthy et al., 2003, their Figure 6; also see May & Solomon, 2014, Supplementary Appendix I). However, it is difficult to know which neurons are being used for a particular task; Clatworthy et al.’s distribution may include many neurons that do not contribute to contrast discrimination performance. In addition, it should be noted that Clatworthy et al.’s empirical distribution was compiled by pooling a large group of neurons, regardless of the neurons’ stimulus preferences. Psychophysical contrast sensitivity (the reciprocal of detection threshold) varies greatly across spatial and temporal frequency (Robson, 1966), suggesting that the lowest semi-saturation contrast (z_{\min}) would vary greatly between different subpopulations tuned to different spatiotemporal frequency combinations. Even if the distribution of z was flat, say, between z_{\min} and z_{\max} for each subpopulation, by pooling the subpopulations, we would be adding together distributions with different lower limits, creating the graded drop-off that we see in the full population. A similar argument could be made regarding the upper end of the distribution.

Different kinds of noise

In most psychophysical models, the noise is a random variable added to the model’s deterministic response, and so the variance of the noise is the variance of the model’s output signal. In our model, the output signal is the estimated stimulus value, and the variance on this output signal does not necessarily show the same characteristics as the noise on the neurons itself. For example, in our model, the variance of the noise on a single neuron increases with the mean response according to Equation (13): Variance is approximately proportional to the mean for low firing rates, and approximately proportional to the square of the mean for high firing rates. However, for the “Constant” parameterization, the variance of the estimated log contrast is constant with respect to the stimulus level. Similarly, Sanborn and Dayan showed that, in their model, with a linear transducer and Gaussian noise with variance proportional to contrast, the variance of the stimulus esti-

mate was proportional to contrast only at very high contrasts: At low contrasts, the variance of the stimulus estimate was proportional to the square of the contrast. In summary, with a population coding model, the characteristics of the noise on the model's output are not necessarily similar to the characteristics of the noise shown by individual neurons.

Physiological plausibility vs. mathematical tractability

In any attempt to model brain processes, there is a tradeoff between physiological plausibility and mathematical tractability. As we make the model more and more realistic from a biological point of view, it becomes harder and harder to understand the model's psychophysical behaviour. Thus, when modelling psychophysical behaviour, it is not always desirable for the mechanisms of the model to be as realistic and complex as when we are modelling the behaviour of individual neurons.

Traditionally, psychophysical performance has usually been modelled using very simple models, in which real-valued signals are sent through a deterministic transducer, and (usually Gaussian) noise is added to the transducer's output (e.g., Wilson, 1980; Legge & Foley, 1980; Meese et al., 2006). This kind of model is straightforward to understand mathematically, and can provide a useful functional description of the system, because the relationships between model characteristics and performance characteristics are well understood (May & Solomon, 2013). However, the transducer model is such a simplification of the biological reality that it may shed little light on the connection between psychophysical performance and the properties of the neurons.

In the work presented here, we have attempted to push our psychophysical modelling much closer towards physiological plausibility without losing the mathematical tractability that traditional models benefit from. To allow mathematical analysis of our model, we had to make some simplifying assumptions. Nevertheless, our model has realistic tuning functions or contrast-response functions, and responds with discrete, integer spike counts that are generated by a promising new model of neuronal variability (Goris et al., 2014), which gives rise to a realistic correlation structure in which correlations increase with spike rate and tuning similarity. We derived equations that accurately predict the model's performance, and reveal surprisingly simple relationships between psychophysical performance and the neuronal parameters.

Acknowledgments

This work was supported by EPSRC grant EP/H033955/1 to Joshua Solomon. The authors would like to thank Julia Förster for helpful discussions, and Tim Meese for providing the data from Meese et al. (2006), which we plotted in Figure 7. We also thank Wei Ji Ma and an anonymous reviewer for very helpful suggestions.

Commercial relationships: none.

Corresponding author: Keith A. May.

Email: keith@keithmay.org

Address: UCL Department of Computer Science, Gower Street, London, WC1E 6BT, UK.

Footnote

¹In this paper, we use the word "trial" in two ways. Firstly, we use it in the way a physiologist would, to mean a stimulus presentation. Secondly, we use it to mean a trial on a 2-alternative forced-choice psychophysical experiment, on which the observer is presented with two stimuli, and has to make a response. To distinguish these two meanings, we always refer to the latter type of trial as a "2AFC trial".

References

- Albrecht, D. G. & Hamilton, D. B. (1982). Striate cortex of monkey and cat: contrast response function. *Journal of Neurophysiology*, *48*, 217-237.
- Arbous, A. G. & Kerrich, J. E. (1951). Accident statistics and the concept of accident-proneness. *Biometrics*, *7*, 340-432.
- Atick, J. J. (1992). Could information theory provide an ecological theory of sensory processing? *Network: Computation in Neural Systems*, *3*, 213-251.
- Atick, J. J., Li, Z., & Redlich, A. N. (1992). Understanding retinal color coding from first principles. *Neural Computation*, *4*, 559-572.
- Atick, J. J., Li, Z., & Redlich, A. N. (1993). What does post-adaptation color appearance reveal about cortical color representation? *Vision Research*, *33*, 123-129.
- Atick, J. J. & Redlich, A. N. (1990). Towards a theory of early visual processing. *Neural Computation*, *2*, 308-320.
- Atick, J. J. & Redlich, A. N. (1992). What does the retina know about natural scenes? *Neural Computation*, *4*, 196-210.

- Averbeck, B. B., Latham, P. E., & Pouget, A. (2006). Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, *7*, 358-366.
- Banks, M. S., Geisler, W. S., & Bennett, P. J. (1987). The physical limits of grating visibility. *Vision Research*, *27*, 1915-1924.
- Beck, J. M., Ma, W. J., Pitkow, X., Latham, P. E., & Pouget, A. (2012). Not noisy, just wrong: The role of suboptimal inference in behavioral variability. *Neuron*, *74*, 30-39.
- Bird, C. M., Henning, G. B., & Wichmann, F. A. (2002). Contrast discrimination with sinusoidal gratings of different spatial frequency. *Journal of the Optical Society of America, A*, *19*, 1267-1273.
- Chirimuuta, M., Clatworthy, P. L., & Tolhurst, D. J. (2003). Coding of the contrasts in natural images by visual cortex (V1) neurons: a Bayesian approach. *Journal of the Optical Society of America, A*, *20*, 1253-1260.
- Chirimuuta, M. & Tolhurst, D. J. (2005). Does a Bayesian model of V1 contrast coding offer a neurophysiological account of human contrast discrimination? *Vision Research*, *45*, 2943-2959.
- Clatworthy, P. L., Chirimuuta, M., Lauritzen, J. S., & Tolhurst, D. J. (2003). Coding of the contrasts in natural images by populations of neurons in primary visual cortex (V1). *Vision Research*, *43*, 1983-2001.
- Cohen, M. R. & Kohn, A. (2011). Measuring and interpreting neuronal correlations. *Nature Neuroscience*, *14*, 811-819.
- Cramér, H. (1946). *Mathematical Methods of Statistics*. Princeton, NJ: Princeton University Press.
- Dakin, S. C. (2001). Information limit on the spatial integration of local orientation signals. *Journal of the Optical Society of America, A*, *18*, 1016-1026.
- Dayan, P. & Abbott, L. F. (2001). *Theoretical Neuroscience: Computational and mathematical modeling of neural systems*. Cambridge, MA: MIT Press.
- de la Rocha, J., Doiron, B., Shea-Brown, E., Josic, K., & Reyes, A. (2007). Correlation between neural spike trains increases with firing rate. *Nature*, *448*, 802-806.
- De Valois, R. L., Albrecht, D. G., & Thorell, L. G. (1982). Spatial frequency selectivity of cells in macaque visual cortex. *Vision Research*, *22*, 545-559.
- Ecker, A. S., Berens, P., Cotton, R. J., Subramanian, M., Denfield, G. H., Cadwell, C. R. et al. (2014). State dependence of noise correlations in macaque primary visual cortex. *Neuron*, *82*, 235-248.
- Ecker, A. S., Berens, P., Keliris, G. A., Bethge, M., Logothetis, N. K., & Tolias, A. S. (2010). Decorrelated neuronal firing in cortical microcircuits. *Science*, *327*, 584-587.
- Geisler, W. S. (1989). Sequential ideal-observer analysis of visual discriminations. *Psychological Review*, *96*, 267-314.
- Geisler, W. S. & Albrecht, D. G. (1997). Visual cortex neurons in monkeys and cats: Detection, discrimination, and identification. *Visual Neuroscience*, *14*, 897-919.
- Goris, R. L. T., Movshon, J. A., & Simoncelli, E. P. (2014). Partitioning neuronal variability. *Nature Neuroscience*, *17*, 858-854.
- Goris, R. L. T., Putzeys, T., Wagemans, J., & Wichmann, F. A. (2013). A neural population model for visual pattern detection. *Psychological Review*, *120*, 472-496.
- Graham, N. V. S. (1989). *Visual Pattern Analyzers*. New York: Oxford University Press.
- Itti, L., Koch, C., & Braun, J. (2000). Revisiting spatial vision: Toward a unifying model. *Journal of the Optical Society of America, A*, *17*, 1899-1917.
- Laughlin, S. (1981). A simple coding procedure enhances a neuron's information capacity. *Zeitung für Naturforschung C*, *36*, 910-912.
- Legge, G. E. (1981). A power law for contrast discrimination. *Vision Research*, *21*, 457-467.
- Legge, G. E. & Foley, J. M. (1980). Contrast masking in human vision. *Journal of the Optical Society of America*, *70*, 1458-1471.
- May, K. A. & Solomon, J. A. (2013). Four theorems on the psychometric function. *PLoS ONE*, *8*(10):e74815.
- May, K. A. & Solomon, J. A. (2014). Connecting psychophysical performance to neuronal response properties II: Contrast decoding and detection. Submitted.
- Mayer, M. J. & Kim, C. B. Y. (1986). Smooth frequency discrimination functions for foveal, high-contrast, mid spatial frequencies. *Journal of the Optical Society of America, A*, *3*, 1957-1969.
- Meese, T., Georgeson, M. A., & Baker, D. H. (2006). Binocular contrast vision at and above threshold. *Journal of Vision*, *6*, 1224-1243.
- Morgan, M., Chubb, C., & Solomon, J. A. (2008). A 'dipper' function for texture discrimination based on orientation variance. *Journal of Vision*, *8*(11):9, 1-8.
- Naka, K. I. & Rushton, W. A. H. (1966). S-potentials from colour units in the retina of fish (cyprinidae). *Journal of Physiology*, *185*, 536-555.
- Nelder, J. A. & Mead, R. (1965). A simplex method for function minimization. *The Computer Journal*, *7*, 308-313.
- Pelli, D. G. (1990). The quantum efficiency of vision. In C. Blakemore (Ed.), *Vision: Coding and Efficiency* (pp. 3-24). Cambridge: Cambridge University Press.
- Pelli, D. G., Burns, C. W., Farell, B., & Moore-Page, D. C. (2006). Feature detection and letter identification. *Vision Research*, *46*, 4646-4674.

- Pelli, D. G. & Farell, B. (1999). Why use noise? *Journal of the Optical Society of America, A*, 16, 647-653.
- Rao, C. R. (1945). Information and accuracy obtainable in the estimation of statistical parameters. *Bulletin of the Calcutta Mathematical Society*, 37, 81-89.
- Robson, J. G. (1966). Spatial and temporal contrast-sensitivity functions of the visual system. *Journal of the Optical Society of America*, 56, 1141-1142.
- Sanborn, A. N. & Dayan, P. (2011). Optimal decisions for contrast discrimination. *Journal of Vision*, 11(14):9, 1-13.
- Smith, M. A. & Kohn, A. (2008). Spatial and temporal scales of neuronal correlation in primary visual cortex. *Journal of Neuroscience*, 28, 12591-12603.
- Solomon, J. A. (2010). Visual discrimination of orientation statistics in crowded and uncrowded arrays. *Journal of Vision*, 10(14):19, 1-16.
- Srinivasan, M. V., Laughlin, S. B., & Dubs, A. (1982). Predictive coding: A fresh view of inhibition in the retina. *Proceedings of the Royal Society of London B*, 216, 427-459.
- Swift, D. J. & Smith, R. A. (1983). Spatial frequency masking and Weber's law. *Vision Research*, 23, 495-505.
- Tadmor, Y. & Tolhurst, D. J. (2000). Calculating the contrasts that retinal ganglion cells and LGN neurones encounter in natural scenes. *Vision Research*, 40, 3145-3157.
- Teo, P. C. & Heeger, D. J. (1994). Perceptual image distortion. *Proceedings of the SPIE*, 2179, 127-141.
- van Kan, P. L. E., Scobey, R. P., & Gabor, A. J. (1985). Response covariance in cat visual cortex. *Experimental Brain Research*, 60, 559-563.
- Wilson, H. R. (1980). A transducer function for threshold and suprathreshold human vision. *Biological Cybernetics*, 38, 171-178.
- Xie, X. (2002). Threshold behaviour of the maximum likelihood method in population decoding. *Network: Computation in Neural Systems*, 13, 447-456.
- Zhaoping, L., Geisler, W. S., & May, K. A. (2011). Human wavelength discrimination of monochromatic light explained by optimal wavelength decoding of light of unknown intensity. *PLoS ONE*, 6(5):e19248.
- Zohary, E., Shadlen, M. N., & Newsome, W. T. (1994). Correlated neuronal discharge rate and its implications for psychophysical performance. *Nature*, 370, 140-143.

Supplementary appendices for “Connecting psychophysical performance to neuronal response properties I: Discrimination of suprathreshold stimuli”

Keith A. May

Centre for Applied Vision Research,
City University London, London, UK.



Joshua A. Solomon

Centre for Applied Vision Research,
City University London, London, UK



Appendix A: Main symbols used in the text

b	Base of logarithm when x represents the log of the physical stimulus units	k_h	Parameter that sets the value of h when $z = 0$ in the “Exponential Naka-Rushton” parameterization
c	Michelson contrast	$k_{r_{\max}}$	Parameter that sets the value of r_{\max} when $z = 0$ in the “Exponential Naka-Rushton” parameterization
c_p	Pedestal Michelson contrast	k_q	Parameter that sets the value of q when $z = 0$ in the “Exponential Naka-Rushton” parameterization
Δc_θ	Threshold Michelson contrast difference between two stimuli on a 2AFC trial	m_h	Parameter that controls how quickly h increases as an exponential function of z in the “Exponential Naka-Rushton” parameterization
f_p	Pedestal spatial frequency	$m_{r_{\max}}$	Parameter that controls how quickly r_{\max} increases as an exponential function of z in the “Exponential Naka-Rushton” parameterization
Δf_θ	Threshold spatial frequency difference between two stimuli on a 2AFC trial	m_q	Parameter that controls how quickly q increases as an exponential function of z in the “Exponential Naka-Rushton” parameterization
G	Random variable representing the gain signal in Goris et al.’s neuronal spiking model	m	$m_h + m_{r_{\max}}$
g	The value of G on a particular stimulus presentation	N	Random variable representing the number of spikes produced by a neuron
h	Density of neuronal tuning functions along the stimulus (x) axis, equal to $1/\delta z$	n	The value of N on a particular stimulus presentation
I	Integral approximation of the Fisher information (found by approximating the sum of Fisher information across neurons using an integral)	\mathbf{N}	Vector of random variables representing the spikes produced by the population of neurons
i	Integer index of the neurons in a population	\mathbf{n}	The value of \mathbf{N} on a particular stimulus presentation
J	Exact expression for the Fisher information	P	Probability
j	Integer index of the neurons in a population		
K	The number of neurons being monitored by the observer		

P_θ	Probability of a correct response at threshold performance level	x_p	Pedestal value of x
p	Probability density	Δx	Difference in x between two stimuli on a 2AFC trial
Q	Function that describes how Fisher information declines with increasing relative spontaneous firing rate, r_0/r_{\max}	Δx_θ	Δx at threshold
q	Tuning function parameter: tuning sharpness	z	Tuning function parameter: stimulus value corresponding to the “middle” of the tuning function, i.e. the peak of the Gaussian function, or the log semi-saturation contrast for the Naka-Rushton function
R	Random variable representing the mean response of a neuron	δz	Spacing between neighbouring z -values in the neuronal population, equal to $1/h$
$r(x)$	The neuron’s tuning function, which gives the value of R on a particular stimulus presentation	γ	$q \ln b$
r_0	Tuning function parameter: spontaneous firing rate	ζ	Transformation of z such that, if ζ has a uniform distribution, z has an exponential distribution
r_{\max}	Tuning function parameter: maximum increment from r_0	ξ	Unspecified physical stimulus units when x is the log of the physical stimulus units
S	An intractable definite integral (defined in Equation (48)), which forms part of the integral approximation of the Fisher information for the “Constant Gaussian” parameterization (see Appendix F)	ξ_p	Pedestal value of ξ
		$\Delta \xi_\theta$	Threshold difference in ξ between two stimuli on a 2AFC trial
T	A definite integral that forms part of the integral approximation of the Fisher information for the “Exponential Naka-Rushton” parameterization (see Appendix G)	ρ_{ij}	Pearson correlation between the spike counts of neurons i and j
		$\rho_{P_{ij}}$	Pearson correlation between the Poisson spiking processes of neurons i and j in Goris, Movshon & Simoncelli’s (2014) neuronal spiking model
u	Parameter of the gamma distribution for the gain signal (used in Appendix D)	$\rho_{G_{ij}}$	Pearson correlation between the gain values of neurons i and j in Goris, Movshon & Simoncelli’s (2014) neuronal spiking model
v	Parameter of the gamma distribution for the gain signal (used in Appendix D)	σ	Standard deviation
W	Weber fraction	σ_G	Standard deviation of the gain signal in Goris et al.’s neuronal spiking model
w	Tuning bandwidth of the Gaussian tuning function (full width at half height), in the same units as x	σ_{tuning}	Standard deviation of the Gaussian tuning function
ω	Tuning bandwidth of the Gaussian tuning function (full width at half height) in octaves, assuming that x is the log of the physical stimulus value	Φ	The integral of a Gaussian with unit area and variance, and zero mean
X	Random variable representing the stimulus level		
x	The value of X on a particular stimulus presentation		
\hat{X}	Random variable representing the estimated stimulus level after decoding the spike counts		
\hat{x}	The value of \hat{X} for a particular stimulus presentation		

Appendix B: Effects of tuning functions on correlations

Effect of spike rate

In this section, we prove that the neuronal correlation, ρ_{ij} , given by Equation (17) of the main text, always increases with increases in mean firing rate of either neuron.

First, let us keep $r_i(x)$ constant, while varying $r_j(x)$. Then, from Equation (17),

$$\rho_{ij} = A\sqrt{y}, \quad (\text{B.1})$$

where A is a constant, given by

$$A = \sigma_G^2 \sqrt{\frac{r_i(x)}{1 + \sigma_G^2 r_i(x)}}, \quad (\text{B.2})$$

and

$$y = \frac{r_j(x)}{1 + \sigma_G^2 r_j(x)}, \quad (\text{B.3})$$

The derivative of y with respect to $r_j(x)$ is given by

$$\frac{dy}{dr_j(x)} = \frac{1}{(1 + \sigma_G^2 r_j(x))^2}, \quad (\text{B.4})$$

which is always positive. Then, since ρ_{ij} must always increase monotonically with increases in y , $d\rho_{ij}/dr_j(x)$ must also be positive. Therefore, increasing $r_j(x)$ while $r_i(x)$ remains constant always causes ρ_{ij} to increase. The same argument applies to increasing $r_i(x)$ while $r_j(x)$ remains constant. Therefore, increasing $r_i(x)$ or $r_j(x)$ or both always causes ρ_{ij} to increase.

Effect of tuning similarity

In this section, we prove that ρ_{ij} increases with tuning similarity between the two neurons. Since ρ_{ij} is also affected by the overall firing rate, we show the effect of increasing the difference in mean spike rate while keeping the sum of mean spike rates constant. Let us define S and D to be the sum and difference, respectively:

$$S = r_i(x) + r_j(x) \quad (\text{B.5})$$

$$D = r_i(x) - r_j(x). \quad (\text{B.6})$$

Then, by expressing $r_i(x)$ and $r_j(x)$ in terms of S and D , and substituting into Equation (17), we obtain

$$\rho_{ij} = \frac{\sigma_G^2}{2} \sqrt{Y}, \quad (\text{B.7})$$

where

$$Y = \frac{S^2 - D^2}{(1 + \sigma_G^2 S/2)^2 - \sigma_G^4 D^2/4}. \quad (\text{B.8})$$

ρ_{ij} and Y vary monotonically, and therefore both peak at the same D value, so the D that maximizes ρ_{ij} is that for which $dY/dD = 0$. It is easily shown that $dY/dD = 0 \Rightarrow D = 0$. Thus, for a given sum of spike rates, the correlation is highest when the two neurons have identical tuning. Figure B.1 plots ρ_{ij} as a function of D for $S = 10$ and $\sigma_G = 0.2$.

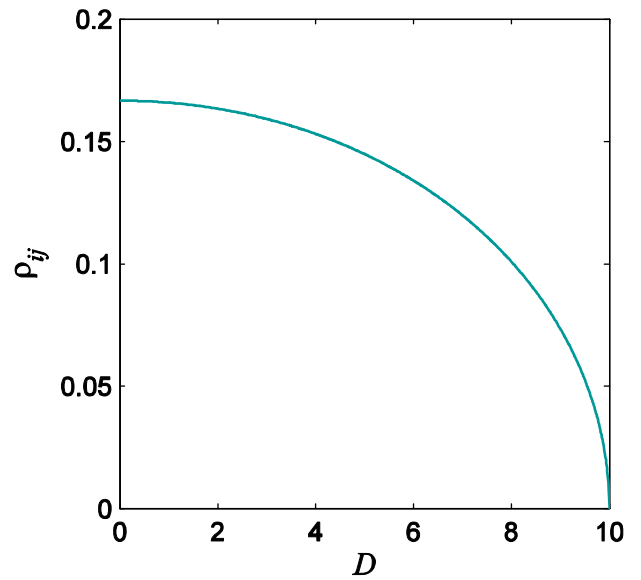


Figure B.1. Correlation, ρ_{ij} , plotted as a function of D for $S = 10$ and $\sigma_G = 0.2$, according to Equations (B.7) and (B.8).

Appendix C: Setting up a population of neurons

The first stage of setting up a population of neurons was to calculate the log semi-saturation contrast, z , for each neuron. In the ‘‘Constant’’ parameterizations, the other neuronal parameters (r_{\max} , q , and relative spontaneous firing rate, r_0/r_{\max}) were constant across the different

neurons. In the ‘‘Exponential’’ parameterizations, r_{\max} and q were allowed to vary as functions of z according to Equations (24) and (25), respectively.

For all the simulations of contrast discrimination that we performed for this paper, the pedestal values were clearly suprathreshold, and within the range of semi-saturation contrasts found physiologically. For these pedestals, the actual values of z_{\min} and z_{\max} in the model do not matter, as long as they are sufficiently far from the ends of the range of pedestals. For our contrast discrimination simulations, z_{\min} was -3 , and the z_{\max} was as high as possible without exceeding 1. This ensured that the stimulus values that we used (x from -2 to 0) were well away from the ends of the range of z -values, as assumed by our equations. These values of z_{\min} and z_{\max} correspond to Michelson contrasts of 0.001 and 10, respectively; although a Michelson contrast greater than 1 is physically impossible, fitted Michelson semi-saturation contrasts of up to 12 have been obtained in physiological studies (Chirimuuta, et al., 2003) – in this case, the physiological data would have been collected only over the lower portion of the contrast-response function.

For our simulations of spatial frequency discrimination, z_{\min} was set to -0.3 and z_{\max} was as high as possible without exceeding 1.7; these were sufficiently far from the edges of the range of log spatial frequency pedestals (0.4 and 1.2) for the performance to be negligibly affected by the values of z_{\min} and z_{\max} .

For the ‘‘Constant’’ parameterizations, the z -values were equally spaced along the x -axis, with a spacing of $1/h$, starting at $z = z_{\min}$. For the ‘‘Exponential’’ parameterizations, we had to place each z value along the x -axis so that the local density of neurons along the log contrast axis was given by $h = k_h \exp(m_h z)$. We defined a variable, ζ , that is related to z in such a way that a flat distribution of ζ maps onto the required distribution of z ; we then stepped through the ζ -values in equal steps, and obtained the corresponding z -value for each ζ .

First, we defined the (constant) step in ζ as $\delta\zeta = 1/k_h$.

Since $\delta z = 1/h = 1/(k_h e^{m_h z})$, we have

$$\delta\zeta = e^{m_h z} \delta z. \quad (\text{C.1})$$

As $\delta\zeta \rightarrow 0$, we have

$$\int d\zeta = \int e^{m_h z} dz \quad (\text{C.2})$$

and therefore

$$\zeta = \frac{e^{m_h z}}{m_h}, \quad (\text{C.3})$$

giving

$$z = \ln(m_h \zeta) / m_h. \quad (\text{C.4})$$

We used Equation (C.3) with $z = z_{\min}$ and $z = z_{\max}$ to give ζ_{\min} and ζ_{\max} , respectively. Then we stepped from ζ_{\min} to ζ_{\max} in equal steps of $\delta\zeta = 1/k_h$, calculating the corresponding values of z using equation (C.4).

Appendix D: Mean reciprocal of the gain

In Goris et al.’s (2014) neuronal spiking model, the gain signal varies according to a gamma distribution:

$$p(g) = \frac{g^{u-1} \exp(-g/v)}{\Gamma(u)v^u}, \quad (\text{D.1})$$

where the shape parameter, u , is given by $u = 1/\sigma_G^2$, and the scale parameter, v , is given by $v = \sigma_G^2$. This gives a distribution with a mean of 1 and a variance of σ_G^2 .

The mean reciprocal of the gain is given by

$$\text{mean}[1/G] = \int_0^{\infty} \frac{p(g)}{g} dg \quad (\text{D.2})$$

$$= \frac{1}{\Gamma(u)v^u} \int_0^{\infty} g^{u-2} \exp(-g/v) dg \quad (\text{D.3})$$

$$= \frac{1}{\Gamma(u)v^u} \times \frac{\Gamma(u-1)}{(1/v)^{u-1}} \quad (\text{D.4})$$

$$= \frac{\Gamma(u-1)}{\Gamma(u)} \times \frac{(v)^{u-1}}{v^u} \quad (\text{D.5})$$

$$= \frac{1}{u-1} \times \frac{1}{v} \quad (\text{D.6})$$

Letting $u = 1/\sigma_G^2$ and $v = \sigma_G^2$, we obtain

$$\text{mean}[1/G] = \frac{1}{1 - \sigma_G^2}. \quad \square \quad (\text{D.7})$$

Appendix E: Relating decoding precision to 2AFC psychophysical performance

In this appendix, we show how to convert the decoding precision into a measure of psychophysical performance. Our focus is on 2AFC tasks, one of the most prevalent psychophysical procedures. On each trial of a 2AFC discrimination task, observers are presented with two stimuli (say, Stimulus 1 and Stimulus 2), with stimulus values x_1 and x_2 . The stimulus with the higher x value is called the target, and the task is to say which stimulus is the target.

Let us suppose the observer performs this task by estimating the value of each stimulus, and choosing the stimulus with the highest estimated value. We can formally describe this process as follows. Let \hat{x}_1 and \hat{x}_2 be the estimated x values of Stimuli 1 and 2, respectively. Define a decision variable, $d = \hat{x}_1 - \hat{x}_2$, and choose Stimulus 1 if $d > 0$, choose Stimulus 2 if $d < 0$, and guess if $d = 0$.

Let us define Δx to be the magnitude of the true difference between the stimulus values:

$$\Delta x = |x_1 - x_2|. \quad (\text{E.1})$$

If the distributions of estimated stimulus values are Gaussian and the estimates are unbiased (which, to a good approximation, are both the case for the model parameterizations that we consider in this paper), then the decision variable, d , will have a Gaussian distribution with mean $x_1 - x_2$. As illustrated in Figure E.1, if the standard deviation of the decision variable is σ_d , then the probability of a correct response, as a function of Δx , is given by

$$P(\text{correct}) = \Phi(\Delta x / \sigma_d), \quad (\text{E.2})$$

where Φ is the integral of a Gaussian with unit area and variance, and zero mean. Thus, the discrimination threshold, Δx_θ (i.e. the stimulus difference corresponding to the threshold proportion correct, P_θ), is given by

$$\Delta x_\theta = \sigma_d \Phi^{-1}(P_\theta). \quad (\text{E.3})$$

The value of P_θ that defines the threshold falls somewhere in the middle of the range 0.5 to 1; different studies use different values.

It will often be useful to take x to be the logarithm of the physical stimulus value; in this case, Δx_θ is the threshold difference of log physical values. To express the threshold in terms of physical stimulus values, ξ

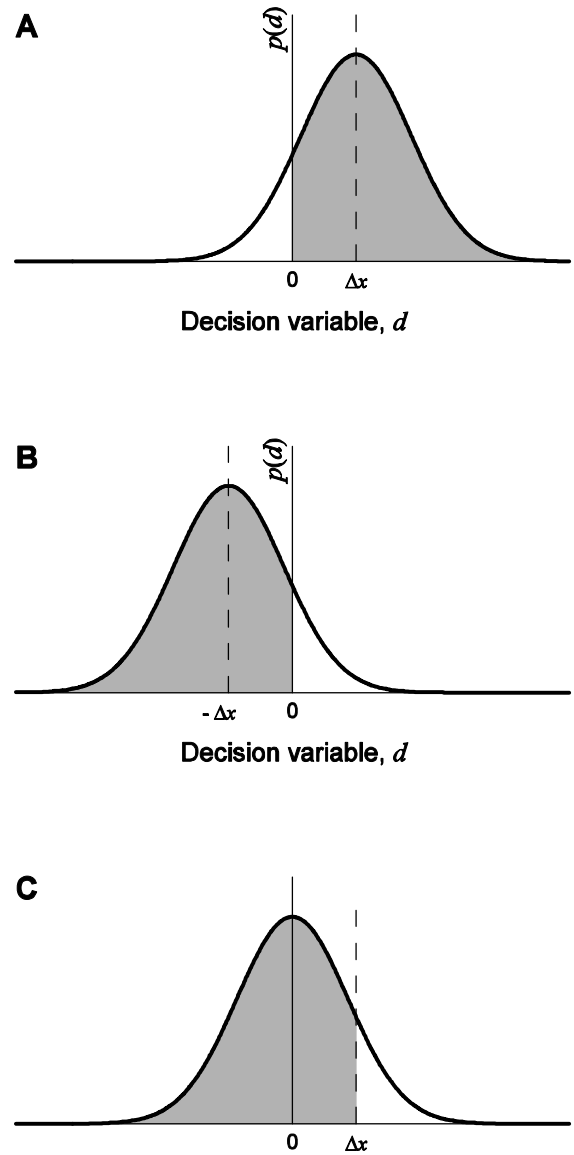


Figure E.1. The probability of a correct response. (A) The case of $x_1 > x_2$. The curve plots the probability density function of the decision variable, d , which is assumed to be a Gaussian with standard deviation, σ_d . In this case, the mean of d , i.e. $x_1 - x_2$, is given by Δx . The observer responds correctly (i.e. chooses Stimulus 1) when $d > 0$. The probability of a correct response is given by the area of the shaded portion. (B) The case of $x_2 > x_1$. In this case, the mean of d , i.e. $x_1 - x_2$, is given by $-\Delta x$. The observer responds correctly (i.e. chooses Stimulus 2) when $d < 0$. The probability of a correct response is again given by the area of the shaded portion. The areas of the shaded portions in (A) and (B) are both the same as that in (C), which shows the integral of a Gaussian with zero mean and standard deviation σ_d , between limits of $-\infty$ and Δx , i.e. $\Phi(\Delta x / \sigma_d)$.

(where $x = \log_b \xi$, giving $\xi = b^x$), we need to convert Δx_θ to a difference of physical values at threshold, which we call $\Delta \xi_\theta$. If the lower stimulus value in physical units is ξ_p (the subscript, p , standing for “pedestal”, the term often used to refer to the lower of two contrasts in a contrast discrimination experiment), then the higher stimulus value at threshold is $\xi_p + \Delta \xi_\theta$. Δx_θ is then given by

$$\Delta x_\theta = \log_b(\xi_p + \Delta \xi_\theta) - \log_b(\xi_p). \quad (\text{E.4})$$

Rearranging Equation (E.4) to make $\Delta \xi_\theta$ the subject, we get

$$\Delta \xi_\theta = (b^{\Delta x_\theta} - 1)\xi_p \quad (\text{E.5})$$

$$= (b^{\sigma_d \Phi^{-1}(P_\theta)} - 1)\xi_p, \quad (\text{E.6})$$

and so the Weber fraction, $W = \Delta \xi_\theta / \xi_p$, is given by

$$W = b^{\Delta x_\theta} - 1 = b^{\sigma_d \Phi^{-1}(P_\theta)} - 1. \quad (\text{E.7})$$

In summary, Equations (E.2) and (E.3) give, respectively, the 2AFC proportion correct and the discrimination threshold, Δx_θ , as functions of σ_d , where both Δx_θ and σ_d are measured in the same units as the stimulus value, x . For some feature dimensions, such as orientation, we would take x to be the physical stimulus value. For other feature dimensions, such as contrast or spatial frequency, it will turn out to be more appropriate to take x to be the \log_b of the physical stimulus value. In the latter case, the Weber fraction for physical stimulus units is given by Equation (E.7), which again is a function of σ_d , where σ_d is measured in the same units as x , the log of the stimulus units.

If we estimate σ_d using the Fisher information, then we obtain expressions that link all these psychophysical performance measures (proportion correct, discrimination threshold, Weber fraction) to the properties of the neurons.

Let $\sigma_{\hat{x}_1}$ and $\sigma_{\hat{x}_2}$ be the standard deviations of the estimates, \hat{x}_1 and \hat{x}_2 . The decision variable is the difference of these two estimates, so, if the two stimulus estimates are statistically independent, the variance of the decision variable is the sum of the variances of the two estimates. This gives

$$\sigma_d = \sqrt{\sigma_{\hat{x}_1}^2 + \sigma_{\hat{x}_2}^2}. \quad (\text{E.8})$$

Let us define $\tau(x)$ to be the precision with which the observer can decode a stimulus with value x . Then, by definition, $\tau(x_1) = 1/\sigma_{\hat{x}_1}^2$ and $\tau(x_2) = 1/\sigma_{\hat{x}_2}^2$. Using these expressions to substitute for $\sigma_{\hat{x}_1}^2$ and $\sigma_{\hat{x}_2}^2$ in Equation (E.8), we obtain

$$\sigma_d = \sqrt{1/\tau(x_1) + 1/\tau(x_2)}. \quad (\text{E.9})$$

Using Equations (E.1) and (E.9) to substitute for Δx and σ_d in Equation (E.2), we get

$$P(\text{correct}) = \Phi \left(\frac{|x_1 - x_2|}{\sqrt{1/\tau(x_1) + 1/\tau(x_2)}} \right). \quad (\text{E.10})$$

We can use a similar approach to find a function that gives Δx_θ for a given pedestal, x_p , i.e. the lower of x_1 and x_2 . To do this, we cannot use Equation (E.9), because that requires us to know both stimulus values at threshold, i.e. we have to know the threshold already. To get around this problem, we can assume that the two stimuli are close enough at threshold for the precision to be about the same in both cases, so

$$1/\tau(x_1) \approx 1/\tau(x_2) \approx 1/\tau(x_p). \quad (\text{E.11})$$

Using Relation (E.11) to substitute for $1/\tau(x_1)$ and $1/\tau(x_2)$ in Relation (E.9), we get

$$\sigma_d \approx \sqrt{2/\tau(x_p)}. \quad (\text{E.12})$$

We can then use Relation (E.12) to substitute for σ_d in Equations (E.3) and (E.7) to give the threshold performance in terms of the precision for decoding the pedestal:

$$\Delta x_\theta \approx \sqrt{2/\tau(x_p)} \Phi^{-1}(P_\theta), \quad (\text{E.13})$$

$$W \approx b^{\sqrt{2/\tau(x_p)} \Phi^{-1}(P_\theta)} - 1. \quad (\text{E.14})$$

These can be inverted to give

$$\tau(x_p) \approx 2 \left(\frac{\Phi^{-1}(P_\theta)}{\Delta x_\theta} \right)^2 \quad (\text{E.15})$$

and

$$\tau(x_p) \approx 2 \left(\frac{\Phi^{-1}(P_\theta)}{\log_b(W+1)} \right)^2. \quad (\text{E.16})$$

If the threshold performance level is defined as $P_\theta = 0.760$, then we have

$$\Phi^{-1}(P_\theta) \approx 1/\sqrt{2} \quad \text{when } P_\theta = 0.760. \quad (\text{E.17})$$

Using Relation (E.17) to substitute for $\Phi^{-1}(P_\theta)$ in Relations (E.13) to (E.16), we obtain some particularly simple expressions relating psychophysical performance to precision when $P_\theta = 0.760$:

$$\Delta x_\theta \approx 1/\sqrt{\tau(x_p)}, \quad (\text{E.18})$$

$$W \approx b^{1/\sqrt{\tau(x_p)}} - 1, \quad (\text{E.19})$$

$$\tau(x_p) \approx 1/(\Delta x_\theta)^2, \quad (\text{E.20})$$

$$\tau(x_p) \approx 1/[\log_b(W+1)]^2. \quad (\text{E.21})$$

In summary, Relation (E.10) describes the psychometric function for 2AFC discrimination in terms of the stimulus difference, $|x_1 - x_2|$, and the precision with which each stimulus, x , can be decoded. Relations (E.13) and (E.18) describe the relationship between Fisher information and the threshold stimulus difference, Δx_θ . x may be the physical stimulus value, or its logarithm. In the latter case, Relations (E.14) and (E.19) give the Weber fraction, i.e. the ratio of stimulus difference to pedestal at threshold when the stimulus values are expressed in physical units (although, in these expressions, the precision, $\tau(x_p)$, is still the reciprocal of the variance of decoded *log stimulus values*). Relations (E.15) and (E.20) give the precision required to yield a threshold of Δx_θ . Relations (E.16) and (E.21) give the precision required to yield a Weber fraction of W .

For Weber fractions of substantially less than 1, which tend to occur in real experiments, Relations (E.16) and (E.21) can be simplified because, for $W \ll 1$, $\log_b(W+1) = \ln(W+1)/\ln b \approx W/\ln b$ (the near-equality is derived from the Mercator series, $\ln(W+1) = W - W^2/2 + W^3/3 - W^4/4 \dots$), which approaches W for small W . Substituting $W/\ln b$ for $\log_b(W+1)$ in Relation (E.16), we get

$$\tau(x_p) \approx \frac{2[\Phi^{-1}(P_\theta)\ln b]^2}{W^2}. \quad (\text{E.22})$$

Thus, to reduce the Weber fraction by a factor of ϕ , we need to increase the precision by a factor of about ϕ^2 .

Appendix F: Investigating $S(q)$

In this appendix, we investigate the function $S(q)$ that appears in Equation (48) of the main paper.

First, we derive Equation (49):

$$S(q) = \frac{S(1)}{q^3}.$$

We can write $S(q)$ as

$$S(q) = \int_{-\infty}^{\infty} f(z) dz, \quad (\text{F.1})$$

where $f(z)$ is a function of z given by

$$f(z) = \frac{z^2 \exp(-2(qz)^2)}{\exp(-(qz)^2) + r_0/r_{\max}}. \quad (\text{F.2})$$

Let $y = qz$. Then

$$\frac{dy}{dz} = q. \quad (\text{F.3})$$

Therefore,

$$S(q) = \frac{1}{q} \int_{-\infty}^{\infty} f(z) \frac{dy}{dz} dz \quad (\text{F.4})$$

$$= \frac{1}{q} \int_{-\infty}^{\infty} f(y/q) dy \quad (\text{F.5})$$

$$= \frac{1}{q^3} \int_{-\infty}^{\infty} \frac{y^2 \exp(-2y^2)}{\exp(-y^2) + r_0/r_{\max}} dy \quad (\text{F.6})$$

$$= \frac{S(1)}{q^3}. \quad \square \quad (\text{F.7})$$

Next, we verify the approximation of $S(1)$ given in Relation (52):

$$S(1) \approx \frac{\sqrt{\pi} Q(r_0/r_{\max})}{2},$$

which can be rewritten as

$$\left[\sqrt{\pi} Q(r_0/r_{\max})/2 \right] / S(1) \approx 1. \quad (\text{F.8})$$

We arrived at this approximation of $S(1)$ by hypothesizing that Equation (53) would be a close approximation of the right hand side of Relation (50), and then working back to Relation (52). Figure F.1 plots the left hand side of Relation (F.8) for $0.0001 \leq r_0/r_{\max} \leq 10,000$. The values of $S(1)$ were calculated by sampling $f(z)$ in Equation (F.2) (with $q=1$) between $z=-10$ and $z=10$, at small intervals, $\delta z = 0.0001$, adding up the function values, and multiplying the sum by δz . For

$0 < r_0/r_{\max} < 0.119$, our approximation slightly overestimates $S(1)$ by a factor that never exceeds 0.7% of $S(1)$ (the peak of the function in Figure F.1 is 1.00687, which occurs at $r_0/r_{\max} = 0.0220$). For

$0.120 < r_0/r_{\max} < 10,000$, our approximation underestimates $S(1)$, but not by much: Even for r_0/r_{\max} as high as 1, the underestimation is only about 3% of $S(1)$, and the underestimation is always less than 6% of $S(1)$.

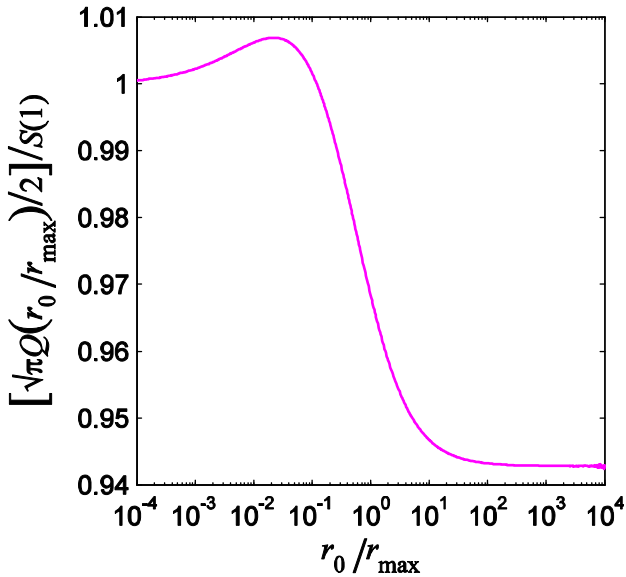


Figure F.1. Testing the accuracy of our approximation of $S(1)$ (Relation (52) of the main paper). The curve plots the approximation as a proportion of $S(1)$ for different values of r_0/r_{\max} .

Appendix G: Solving the integral in Equation (60)

We needed to find the definite integral, T , given by

$$T = \int_0^{\infty} \frac{(m_h \zeta)^{m_{r_{\max}}/m_h} b^{2q \ln(m_h \zeta)/m_h}}{(b^{q \ln(m_h \zeta)/m_h} + b^{qx})^3} d\zeta \quad (\text{G.1})$$

Mathematica was unable to solve this definite integral in the form presented in Equation (G.1). It could produce the indefinite integral, but inserting ∞ into this expression gave an indeterminate form, and we could find no way of obtaining a determinate form using either substitution or l'Hôpital's rule. However, we found that, by making substitutions, we could rewrite Equation (G.1) in a form that Mathematica was able to solve.

For convenience, we repeat Equation (C.4) (identical to Equation (57) of the main paper) here:

$$z = \ln(m_h \zeta)/m_h. \quad (\text{G.2})$$

This gives us

$$\frac{d\zeta}{dz} = m_h \zeta. \quad (\text{G.3})$$

Next, let us define

$$y = \frac{e^{mz}}{m}, \quad (\text{G.4})$$

where

$$m = m_h + m_{r_{\max}}. \quad (\text{G.5})$$

Then

$$\ln(m_h \zeta)/m_h = z = \ln(my)/m, \quad (\text{G.6})$$

and, using Equation (G.2) to substitute for z in Equation (G.4), and rearranging, we obtain

$$(m_h \zeta)^{m_{r_{\max}}/m_h} = \frac{my}{m_h \zeta}. \quad (\text{G.7})$$

Also,

$$\zeta = 0 \Rightarrow y = 0, \quad (\text{G.8})$$

$$\text{as } \zeta \rightarrow \infty, y \rightarrow \infty, \quad (\text{G.9})$$

$$\frac{d\zeta}{dy} = \frac{d\zeta}{dz} \times \frac{dz}{dy} = \frac{m_h \zeta}{my}. \quad (\text{G.10})$$

Using Equations (G.6) and (G.7) to substitute for $\ln(m_h \zeta)/m_h$ and $(m_h \zeta)^{m_{r_{\max}}/m_h}$ in Equation (G.1), we obtain

$$T = \int_0^{\infty} \frac{b^{2q \ln(my)/m}}{\left(b^{q \ln(my)/m} + b^{qx}\right)^3} \times \frac{my}{m_h \zeta} d\zeta \quad (\text{G.11})$$

Using (G.8), (G.9) and (G.10) to change the integration variable in Equation (G.11) from ζ to y , we obtain

$$T = \int_0^{\infty} \frac{b^{2q \ln(my)/m}}{\left(b^{q \ln(my)/m} + b^{qx}\right)^3} dy \quad (\text{G.12})$$

If we define

$$\eta = q \ln(b)/m, \quad (\text{G.13})$$

then Equation (G.12) simplifies to

$$T = \int_0^{\infty} \frac{(my)^{2\eta}}{\left((my)^\eta + e^{\eta mx}\right)^3} dy. \quad (\text{G.14})$$

This integral converges (i.e. has a finite solution) for $\eta > 1$. Mathematica was able to solve it, and gave the answer as

$$T = \frac{e^{-\eta mx} (1 + \eta) (e^{-\eta mx} m^\eta)^{-1/\eta} \pi \csc(\pi/\eta)}{2\eta^3}, \quad (\text{G.15})$$

which can be rewritten as

$$T = \frac{\pi m (m + \gamma) e^{(m-\gamma)x}}{2\gamma^3 \sin(\pi m/\gamma)} \quad (\text{G.16})$$

where

$$\gamma = q \ln b. \quad (\text{G.17})$$

The restriction $\eta > 1$ corresponds to $\gamma > m$.

In conclusion, then,

$$\int_0^{\infty} \frac{(m_h \zeta)^{m_{r_{\max}}/m_h} b^{2q \ln(m_h \zeta)/m_h}}{\left(b^{q \ln(m_h \zeta)/m_h} + b^{qx}\right)^3} d\zeta = \frac{\pi m (m + \gamma) e^{(m-\gamma)x}}{2\gamma^3 \sin(\pi m/\gamma)}, \quad (\text{G.18})$$

where

$$m = m_h + m_{r_{\max}} \quad (\text{G.19})$$

and

$$\gamma = q \ln b, \quad (\text{G.20})$$

provided that

$$\gamma > m. \quad (\text{G.21})$$

Appendix H: Simulation Methods

Sampling the stimulus axis

In each simulation of contrast discrimination, the stimulus value, x , fell between -2 and 0 ; for simulations of spatial frequency discrimination, x fell between 0.4 and 1.2 . Between the lowest and highest stimulus level, the stimulus axis was sampled in steps of $\sqrt{\tau_{\max}}/5$, where τ_{\max} is the maximum predicted precision across the range of stimulus levels.

Generating spike counts

For each stimulus value, x , we simulated the presentation of 10,000 stimuli. On each simulated stimulus presentation, we sampled a pseudorandom gain value, g , from a Gamma distribution with mean 1 and standard deviation σ_G . For each neuron, j , on that stimulus presentation, we sampled a pseudorandom spike count from an independent Poisson distribution with mean $g \times r_j(x)$, where $r_j(x)$ is the output of neuron j 's tuning function, given by Equation (4) or (6) of the main paper, as appropriate.

Decoding the spike counts

The spike counts were decoded as described in the main paper.

Evaluating model performance

The model's precision for each stimulus level, x , was defined as the reciprocal of the variance of estimated stimulus values across the 10,000 presentations of that stimulus level.

We used the estimated stimulus values to simulate 2AFC discrimination experiments. For each value of x ,

we took each numbered stimulus estimate (1 to 10,000) and compared it with the same numbered stimulus estimate for all higher stimulus values. Each comparison represented a trial in a 2AFC discrimination experiment, in which the lower-valued stimulus was the pedestal, and the higher-valued stimulus was the target. The response was taken to be correct if the stimulus with higher x had a higher *estimated* x , and incorrect if the stimulus with higher x had a lower estimated x . For each pair of stimuli, we then found the number of correct responses, and divided by the total number of 2AFC trials for that pair (always 10,000). This gave us a psychometric function of stimulus difference against probability correct for the model. Assuming that the stimulus level, x , was the log of the physical stimulus values, we expressed each stimulus difference in physical stimulus units, and fitted a Weibull psychometric function (May & Solomon, 2013) to the model's percent-correct data. We used a Weibull function with two parameters: the "shape" parameter, β , and the "threshold" parameter, α , which gives the stimulus difference corresponding to a proportion correct of $1 - 0.5/e \approx 0.816\dots$. For the simulations of Meese et al.'s (2006) experiment, the discrimination threshold is simply given by α , since that is how Meese et al. defined it. For the other simulations, the discrimination threshold was defined as the stimulus difference (in physical units) that corresponded to a proportion correct of 0.75 on the fitted Weibull function.

Appendix I: Fisher information of the gamma-Poisson mixture distribution

The Fisher information, J , for decoding stimulus x is given by the average negative 2nd derivative of the log-likelihood function:

$$J = \left\langle -\frac{d^2 \ln P(\mathbf{N} = \mathbf{n} | X = x)}{dx^2} \right\rangle, \quad (\text{I.1})$$

where $\langle y \rangle$ is the trial-averaged value of y . If there are just two neurons, i and j , then, from Equation (22) of the main paper, we have

$$\ln P(\mathbf{N} = \mathbf{n} | X = x) = A + B - C \\ + \text{terms independent of } x, \quad (\text{I.2})$$

where

$$A = n_i \ln(r_i), \quad (\text{I.3})$$

$$B = n_j \ln(r_j), \quad (\text{I.4})$$

$$C = (n_i + n_j + 1/\sigma_G^2) \ln(r_i + r_j + 1/\sigma_G^2). \quad (\text{I.5})$$

Note, we have used r_i and r_j in place of $r_i(x)$ and $r_j(x)$ to reduce notational clutter. Then,

$$\frac{d^2 \ln P(\mathbf{N} = \mathbf{n} | X = x)}{dx^2} = \frac{d^2 A}{dx^2} + \frac{d^2 B}{dx^2} - \frac{d^2 C}{dx^2}, \quad (\text{I.6})$$

where

$$\frac{d^2 A}{dx^2} = \frac{n_i [r_i r_i'' - (r_i')^2]}{r_i^2}, \quad (\text{I.7})$$

$$\frac{d^2 B}{dx^2} = \frac{n_j [r_j r_j'' - (r_j')^2]}{r_j^2}, \quad (\text{I.8})$$

$$\frac{d^2 C}{dx^2} = \frac{(r_i + r_j + 1/\sigma_G^2) (r_i'' + r_j'') - (r_i' + r_j')^2}{(r_i + r_j + 1/\sigma_G^2)^2} \\ \times (n_i + n_j + 1/\sigma_G^2). \quad (\text{I.9})$$

To find the mean value of the negative 2nd derivative of the log-likelihood function, we make multiple usage of the following easily proved theorem:

$$\langle ay + b \rangle = a \langle y \rangle + b, \quad (\text{I.10})$$

Because of Equation (I.10), the mean values of the expressions in Equations (I.7) to (I.9) can be found simply by replacing n_i and n_j with their mean values, which are r_i and r_j , respectively. This gives us

$$\left\langle \frac{d^2 A}{dx^2} \right\rangle = r_i'' - \frac{(r_i')^2}{r_i}, \quad (\text{I.11})$$

$$\left\langle \frac{d^2 B}{dx^2} \right\rangle = r_j'' - \frac{(r_j')^2}{r_j}, \quad (\text{I.12})$$

$$\left\langle \frac{d^2 C}{dx^2} \right\rangle = r_i'' + r_j'' - \frac{(r_i' + r_j')^2}{r_i + r_j + 1/\sigma_G^2}. \quad (\text{I.13})$$

Thus,

$$J = -\left\langle \frac{d^2 A}{dx^2} \right\rangle - \left\langle \frac{d^2 B}{dx^2} \right\rangle + \left\langle \frac{d^2 C}{dx^2} \right\rangle \quad (\text{I.14})$$

$$= \frac{(r'_i)^2}{r_i} + \frac{(r'_j)^2}{r_j} - \frac{(r'_i + r'_j)^2}{r_i + r_j + 1/\sigma_G^2}. \quad \square \quad (\text{I.15})$$

It is important to note that, for this 2-neuron system, the Fisher information can greatly overestimate the decoding precision. Suppose that $r'_i = -r'_j$ at x (i.e. the tuning curve slopes are equal in magnitude and opposite in sign). Then,

$$J = \frac{(r'_i)^2}{r_i} + \frac{(r'_j)^2}{r_j}. \quad (\text{I.16})$$

Equation (I.16) gives the Cramér-Rao bound on the decoding precision when the gain is unknown. This is higher than the decoding precision that we derived from the Cramér-Rao bound when the gain is known, which would be the value in Equation (I.16) multiplied by $1 - \sigma_G^2$ (see Relation (31) of the main paper). This would seem to imply that decoding precision gets better if we ignore our knowledge of the gain level! The resolution of this apparent paradox is that the Cramér-Rao bound is not a good estimate of the decoding precision when the number of neurons is small: It is an upper bound, but is not always achievable, even as the spike rate approaches infinity. As a check, we set up a Monte Carlo simulation of a 2-neuron system like the one in this appendix for Gaussian-tuned neurons such that $r'_i = -r'_j$ at x . We carried out maximum-likelihood decoding both with and without knowledge of the gain, and found that the decoding precision was consistently slightly better when the gain was known.