Neural Computation, 23(6):1536-67 (2011).

Multiplicative Gain Modulation Arises through Unsupervised Learning in a Predictive Coding Model of Cortical Function

Kris De Meyer
Michael W. Spratling
Division of Engineering, King's College London, WC2R 2LS, United Kingdom

Keywords: Gain Modulation, Population Coding, Predictive Coding, Neural Network, Basis Functions

Abstract

The combination of two or more population-coded signals in a neural model of predictive coding can give rise to multiplicative gain modulation in the response properties of individual neurons. Synaptic weights generating these multiplicative response properties can be learned using an unsupervised, Hebbian, learning rule. The behaviour of the model is compared to empirical data on gaze-dependent gain modulation of cortical cells, and found to be in good agreement with a range of physiological observations. Furthermore, it is demonstrated that the model can learn to represent a set of basis functions. The current paper thus connects an often-observed neurophysiological phenomenon and important neurocomputational principle (gain modulation) with an influential theory of brain operation (predictive coding).

1 Introduction

Predictive Coding (PC) provides an elegant neural theory of how perceptual information can be combined with prior experience in order to compute the most likely interpretation of sensory data. PC is based on the principle of minimising the residual error between bottom-up, stimulus-driven activity and top-down predictions generated from an internal representation of the world. Several past proposals for how PC could be implemented in neural circuitry have all suggested that cortical feedback connections carry the predictions, acting on regions at preceding stages of a hierarchical information processing pathway in order to calculate the residual error which is then propagated via cortical feedforward connections (Barlow, 1994; Friston, 2005; Jehee, Rothkopf, Beck, & Ballard, 2006; Kilner, Friston, & Frith, 2007; Mumford, 1992; Murray, Schrater, & Kersten, 2004; Rao & Ballard, 1999). An alternative implementation of PC, the PC/BC model (Spratling, 2008a,b) proposes that residual error calculation is performed by connections intrinsic to each cortical region, rather than via feedforward and feedback connections between cortical regions. When viewed in this way PC can be interpreted as a mechanism of competition between different representations of the sensory world.

In parallel, *Gain Modulation* (GM) has been proposed as an important neurocomputational principle allowing neurons to combine information from multiple sources (Salinas & Sejnowski, 2001). GM arises when one input signal (a modulatory one) affects the sensitivity of the neuron to another signal, without modifying its selectivity (ibid.). The sources of interacting signals can

be sensory, postural, motor-related or cognitive. Here we focus on the gaze-dependent modulation of visual responses first observed by Andersen & Mountcastle (1983) in area 7a in parietal cortex, and later confirmed in many other cortical areas (e.g., Andersen, Bracewell, Barash, Gnadt, & Fogassi, 1990; Bremmer, Ilg, Thiele, Distler, & Hoffmann, 1997; Bremmer, Distler, & Hoffmann, 1997; Cassanello & Ferrera, 2007; Galletti & Battaglini, 1989; Galletti, Battaglini, & Fattori, 1995). Andersen & Mountcastle (1983) noted that visual responses of many parietal cortical cells were systematically affected by direction of gaze or eye position in a way that suggested a *multiplicative* interaction between visual and eye position signals. They coined the term *Gain Field* (GF) to describe the sensitivity of the visual response to eye position, in parallel to the concept of *Receptive Field* (RF) which describes sensory selectivity. On the theoretical side, it has been proposed that gain-modulated neurons form the basis for a generic class of computations (most notably, coordinate transformations between different frames of reference) through their capacity to form *basis function sets* (Pouget & Sejnowski, 1997).

This article aims to make a connection between PC and GM by demonstrating that multiplicative GM arises naturally when two population-coded input signals combine in the PC/BC model – as a consequence of competitive interactions between PC/BC neurons. Furthermore, synaptic weights generating multiplicative response properties are easily learned using an unsupervised, activity-dependent learning rule. Previous modeling studies have shown that multiplicative interactions can arise through supervised learning (Zipser & Andersen, 1988), or that multiplicative interactions can emerge through population effects in a network with fixed, recurrent connections (Salinas & Abbott, 1996). The current results add to this by demonstrating that multiplicative interactions can also emerge through unsupervised learning, and by tentatively linking GM to the theoretical framework of PC. The behaviour of the model is in good agreement with a range of physiological observations regarding gaze-dependent GM. Moreover, we demonstrate that the network is capable of generating a basis function set, and can thus form part of larger networks that perform more complex computations.

Note that previous simulations using the PC/BC model (Spratling, 2008a) have explored how attentional signals carried by feedback connections to a cortical area modulate the response of neurons to sensory-driven signals carried by the feedforward connections to that area. Here, we are exploring a distinct form of gain modulation, one in which different feedforward inputs to the same cortical area are combined such that one modulates the response to the other.

2 Methods

2.1 Model

Spratling (2008a) introduced the ***nonlinear PC/BC*** model, a reformulation of predictive coding consistent with the biased competition theory of attention. PC/BC is a hierarchical model with multiple cortical areas connected through feedforward and/or feedback links. In this article we report, similar to Spratling (2010), experiments for a single cortical area in isolation. We therefore present a simplified, single-area model, and refer to (Spratling, 2008a) for a description of the full hierarchical model.

The single-area PC/BC model is illustrated in Fig. 1 and implemented by the following two equations:

$$\mathbf{e} = \mathbf{x} \oslash \left(\epsilon_2 + \hat{\mathbf{W}}^T \mathbf{y}\right) \tag{1}$$

$$\mathbf{y} \leftarrow (\epsilon_1 + \mathbf{y}) \otimes \mathbf{W}\mathbf{e} \tag{2}$$

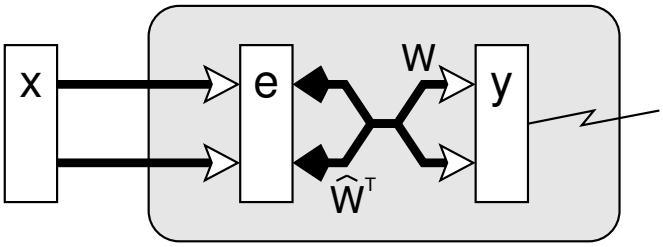

Figure 1: The single-area PC/BC model. Rectangles represent populations of nodes, with $\mathbf{y}$ labeling a population of prediction nodes, $\mathbf{e}$ labeling the error nodes, and $\mathbf{x}$ labeling the input units. Open arrows signify excitatory connections, filled arrows indicate inhibitory connections, crossed connections signify a many-to-many connectivity pattern between the neurons in two populations, parallel connections indicate a one-to-one mapping between the neurons in two populations. The large shaded box with rounded corners indicates the bounds of the single PC/BC area.

$\mathbf{x}$ is a $(m \times 1)$ vector containing the input to the PC/BC network. $\mathbf{e}$ is a $(m \times 1)$ vector of *error node* activations. $\mathbf{y}$ is a $(n \times 1)$ vector of *prediction node* activations. $\mathbf{W}$ is a $(n \times m)$ matrix of synaptic weight values. $\hat{\mathbf{W}}$ is a matrix representing the same synaptic weight values as $\mathbf{W}$ but with each row normalised to have a maximum value of 1. ϵ_1 and ϵ_2 are parameters. $\oslash$ and $\otimes$ indicate element-wise division and multiplication respectively. After suitably initialising $\mathbf{x}$ and $\mathbf{y}$ (see below) Eqs. 1 and 2 are evaluated iteratively for a number of time-steps. Values of $\mathbf{y}$ calculated at time t are substituted back into the equations to obtain the node activations at time $t+1$. After a number of iterations $\mathbf{e}$ and $\mathbf{y}$ approach steady-state values. We always evaluated the equations for 60 time-steps, a value amply sufficient to reach steady-state.[1] For each input stimulus, $\mathbf{x}$ was initialised to the values generated by the input units (as explained in Sect. 2.2) and $\mathbf{y}$ was initialised to contain all zeros.[2] ϵ_2 is a small constant introduced to prevent division-by-zero errors. The ratio $\frac{\epsilon_1}{\epsilon_2}$ determines the input/output gain of the network when $\mathbf{y} \approx 0$, i.e., at the first iteration. We used $\epsilon_1 = 0.001$ and $\epsilon_2 = 0.05$, the same values as used previously in (De Meyer & Spratling, 2009).

Equation 1 describes the calculation of the neural activity of the error-detecting nodes. These values are a function of the input to the PC/BC network divisively modulated by a weighted sum of the output of the prediction nodes. Equation 2 describes the updating of the prediction node activations. The response of each prediction node is a function of its activation at the previous iteration and a weighted sum of afferent inputs from the error nodes. The activation of the error nodes can be interpreted in two ways. Firstly, $\mathbf{e}$ can be considered to represent the residual error between the input $\mathbf{x}$ and the reconstruction of the input $\left(\hat{\mathbf{W}}^T\mathbf{y}\right)$ generated by the prediction nodes. The values of $\mathbf{e}$ indicate the degree of mismatch between the top-down reconstruction of the input and the actual input (assuming ϵ_2 is sufficiently small). When a value within $\mathbf{e}$ is greater than 1 it indicates that a particular element of the input is under-represented in the reconstruction, a value of less than 1 indicates that a particular element of the input is

[1]The temporal dynamics of $\mathbf{y}$ values can be seen in the 4 'Response' graphs of Figs. 3(b) to (e), and one such graph in Fig. 4(a). These responses all reach their steady-state swiftly (within less than 20 time-steps). Early in the training procedure, when weights are less differentiated, it may take longer for the response to reach steady-state. The choice of 60 time-steps thus errs on the side of caution.

[2]Initialising $\mathbf{y}$ to non-zero, randomised values has no effect on the steady-state values reached. The only exception to this rule occurs when the input is "ambiguous" (Spratling & Johnson, 2001), but this condition has no consequences for the results presented in this article.

over-represented, and a value of 1 indicates that the top-down reconstruction perfectly predicts the bottom-up stimulation. A second interpretation is that $\mathbf{e}$ represents the inhibited inputs to a population of competing prediction nodes. Each prediction node modulates its own inputs, which helps stabilise the response, since a strongly (or weakly) active prediction node will suppress (magnify) its inputs, and hence, reduce (enhance) its own response. Prediction nodes that share inputs (i.e., that have overlapping RFs) will also modulate each other's inputs. This generates a form of competition between the prediction nodes, such that each node effectively tries to block other prediction nodes from responding to the inputs which it represents. According to this interpretation, therefore, *prediction nodes compete to represent input*.

During the network training phase, synaptic weight values were adjusted using the following learning rule (Spratling, De Meyer, & Kompass, 2009):

$$\mathbf{W} \leftarrow \mathbf{W} \otimes \{1 + \beta \mathbf{y} \left(\mathbf{e}^T - 1\right)\} \tag{3}$$

$\mathbf{e}$ and $\mathbf{y}$ are the steady-state activations of error nodes and prediction nodes respectively (obtained after 60 time-steps as explained above). β is a positive constant determining the learning rate. A value of $\beta = 0.01$ was used for all experiments described in this article. Following learning, weights were clipped at zero to ensure that they were non-negative. Weights were initialised to random values chosen from a Gaussian distribution with mean 0.5 and standard deviation 0.125.

The learning rule operates by minimising the error between the actual input stimulus $\mathbf{x}$ and the input reconstructed from the prediction node responses $\left(\hat{\mathbf{W}}^T\mathbf{y}\right)$. It *increases* the weights between under-represented error nodes (i.e., for $\mathbf{e} > 1$) and active prediction nodes, while it *decreases* the weights between over-represented error nodes (i.e., for $\mathbf{e} < 1$) and active prediction nodes. A weight stops changing value when the top-down reconstruction is perfect (i.e., when $\hat{\mathbf{W}}^T\mathbf{y} = \mathbf{x}$) or when the weight is zero. An additional advantage of the learning rule is that it is *self-normalising*, i.e., over the course of the training it drives the sum of the synaptic weights received by each prediction node towards a value of one. Following from Eq. 2, strong weights in the network generate high $\mathbf{y}$ values. This, in turn, means strong divisive feedback which keeps $\mathbf{e}$ values low (see Eq. 1). For low $\mathbf{e}$ values, the learning rule (Eq. 3) causes synaptic weights to decrease. Conversely, weak weights produce weak divisive feedback and large values of $\mathbf{e}$, causing weights to increase. Self-normalisation is attractive from the point of view of biological plausibility, as synaptic weights cannot increase without bound.

2.2 Input

Similarly to many related models (e.g., Zipser & Andersen, 1988; Salinas & Abbott, 1996; Pouget & Sejnowski, 1997; Cassanello & Ferrera, 2007), input to the PC/BC network is generated by populations of topographically-organised input units with Gaussian or sigmoid response profiles. These responses encode the input variables: in particular, the retinal locations of visual stimuli and eye position or direction of gaze.

Given a 1D Gaussian response profile, the response h_i of unit i is generated by:

$$h_i(x) = h_{max} \exp\left(-\frac{(x - a_i)^2}{2\sigma^2}\right) \tag{4}$$

With a_i the centre of the Gaussian response profile of input unit i, x the value for which the response is to be determined, σ the standard deviation and h_{max} the amplitude (maximum) of the Gaussian curve. In all experiments in this article, h_{max} was set to a value of 1. A population signal is constructed from input units $i = 1, 2, 3, ...$ with different a_i values spread

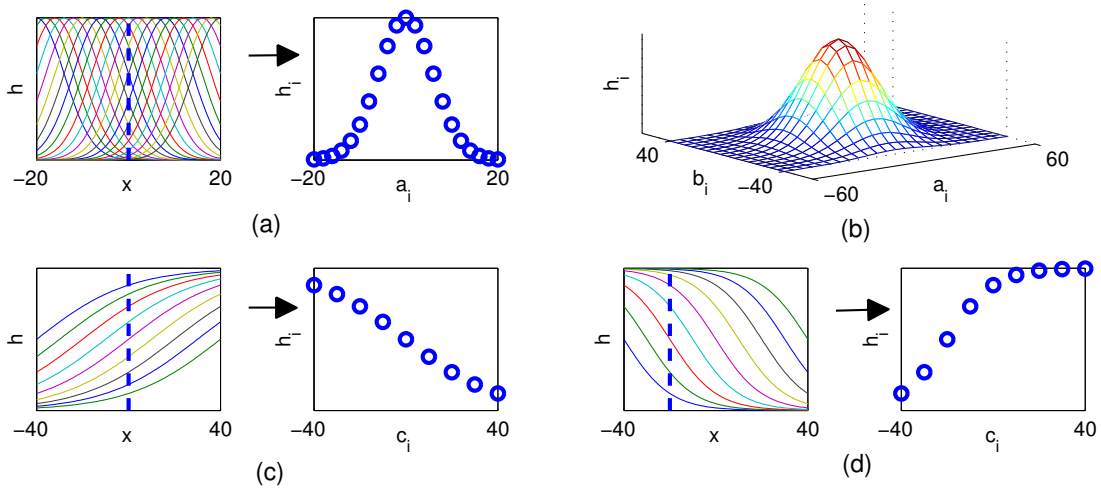

Figure 2: Examples of input used in the experiments in Sect. 3. (a) Input units with 1D Gaussian response profiles (left) generate the population signal (right) for $x = 0°$ (indicated by the vertical dashed line in the left-hand graph). The peaks of the Gaussian profiles a_i are spaced $2°$ apart, with standard deviation $\sigma = 6°$ (see Eq. 4). (b) Population signal generated by units with 2D Gaussian response profiles for $(x, y) = (0°, 0°)$. Gaussian peaks (a_i, b_i) are spaced $5°$ apart in both dimensions, with $\sigma = 16°$ (see Eq. 5). (c) Input units with sigmoid response profiles (left) and the corresponding population signal (right) for $x = 0°$. Inflection points c_i are spaced $10°$ apart, slope factor $T = +20°$ (see Eq. 6). (d) Input units with response profiles similar to (c), but with slope factor $T = -10°$. The corresponding population signal is shown for $x = -20°$. The strength of each input (and hence the scale of the vertical axes) depends on the value of h_{max} in Eqs. 4, 5 and 6; in all experiments reported in this paper $h_{max} = 1$.

evenly across a given range. Figure 2(a) depicts the response profiles of such a population and the corresponding population signal for $x = 0°$.

Given a 2D Gaussian response profile, the response h_i of unit i is calculated as:

$$h_i(x, y) = h_{max} \exp\left(-\frac{(x - a_i)^2 + (y - b_i)^2}{2\sigma^2}\right) \tag{5}$$

A population signal generated by input units with evenly-spaced Gaussian centres (a_i, b_i) and for input $(x, y) = (0°, 0°)$ is shown in Fig. 2(b).

One-dimensional sigmoid responses are generated by:

$$h_i(x) = \frac{h_{max}}{1 + \exp\left(-\frac{x - c_i}{T}\right)} \tag{6}$$

With c_i the inflection point (midpoint) and T the slope factor, a parameter determining the steepness of the sigmoid curve. For positive values of T the curve increases for increasing x, whereas for negative values it decreases for increasing x. Higher absolute values of T correspond to more shallow sigmoids. Figure 2(c) and (d) show examples of sigmoid response profiles for two different slope factors, and the corresponding population signal for different values of x.

In some experiments, we investigate the influence of noise on the training and response properties of the PC/BC network by applying noise to the input:

$$h_i = h_i[1 + \rho]_+ \tag{7}$$

Each input signal is independently modified by random variable ρ sampled from a normal distribution with zero mean and given standard deviation. $[\,]_+$ means positive rectification to prevent the occurrence of negative input signals. This multiplicative noise model is equivalent to additive noise models with variable variance as used by e.g., Salinas & Abbott (1996).

2.3 Training and Testing Procedures

In Sects. 3.2 and 3.3 each network undergoes a training phase consisting of 30000 training epochs. In each training epoch a random combination of visual and eye position signals was presented to the network, the steady-state values of $\mathbf{y}$ and $\mathbf{e}$ calculated using Eqs. 1 and 2, and the synaptic weights adjusted using Eq. 3. The form of the input stimuli was specific to different experiments and will be discussed as part of each experimental setup. In general, the functional form of the response properties we report in the results section tended to appear after 5000-20000 training epochs, dependent on the type of experiment. We trained significantly longer, to 30000 training epochs, to allow response properties to settle fully. The response properties remained stable for many 10000s training epochs (beyond 120000 epochs in the case of 1D signals).[3] A measure we used to check that response properties had indeed fully formed after 30000 training epochs in all experimental conditions was the *average reconstruction error*. As we noted in Sect. 2.1, the activation of the error nodes ($\mathbf{e}$) can be regarded as the residual error between the input ($\mathbf{x}$) and the reconstruction of the input $\left(\hat{\mathbf{W}}^T\mathbf{y}\right)$. Individual values of $\mathbf{e}$ tend to 1 for each entry of $\mathbf{x}$ that is perfectly reconstructed. The average of $\mathbf{e}$ over all nodes and over a number of training epochs (500 was used here) thus provides a measure of how well, on average, the network reconstructs the input. We observed in experiments that this value increased rapidly at the start of the training phase[4] and then approached a steady-state as response properties started to settle. In all training experiments reported here the average reconstruction error had reached its steady state value well before reaching 30000 epochs.

We conducted experiments where the visual stimuli were 1D Gaussian population signals (Eq. 4), and experiments where the visual stimuli were 2D Gaussian signals (Eq. 5). For computational reasons we wanted to use similar numbers of prediction nodes and training epochs in both types of experiments. One important prerequisite to achieve this is to keep the ratio of the area covered by an individual Gaussian input unit to the full area of the visual field approximately constant. This ratio determines how often, on average, a particular input unit is strongly activated by a randomly selected visual input r_x or (r_x, r_y) pair during the training phase. With $fw_{\frac{1}{2}}$, the *Full Width at Half Maximum* of the Gaussian (see below), the ratio for the 1D case is given by:

$$\frac{fw_{\frac{1}{2}}}{(r_x^{max} - r_x^{min})} \tag{8}$$

For the 2D case, it is the area of a circle with diameter $fw_{\frac{1}{2}}$ divided by the size of the rectangular

[3] Training for a very long time led to further change in the response properties. This change is caused by the fundamental difference in stimulation provided by spatially-localised Gaussian signals and sigmoid signals which are not localised spatially (i.e., the latter inputs are 'on' more frequently than the former). We do not further address this technical issue but note that the problem can easily be resolved by techniques that are used in many neural network algorithms to avoid overtraining: e.g., by reducing the learning rate β as training progresses, or by terminating the training using a formal stopping criterion based on a measure of training progress.

[4] At the start of training all weights are relatively large, meaning that each input is over-represented and many values of $\mathbf{e} << 1$. As weight values decrease and start to represent the input better, values of $\mathbf{e}$ increase towards 1, and the mean of $\mathbf{e}$ increases as a consequence.

visual field:

$$\frac{\pi}{4}\frac{(fw_{\frac{1}{2}})^2}{(r_x^{max}-r_x^{min})(r_y^{max}-r_y^{min})} \tag{9}$$

For a 1D Gaussian, $fw_{\frac{1}{2}}$ is the distance between the two points at half the peak value. For a 2D Gaussian, it is the diameter of the circular contour at half the peak value. Its value can be obtained by solving Eq. 4 or Eq. 5 for $h_i = \frac{1}{2}h_{max}$ (Weisstein, 1999), which results in:

$$fw_{\frac{1}{2}} = 2\sqrt{2\ln 2}\sigma \approx 2.36 \times \sigma \tag{10}$$

When *testing* the networks the response properties of the prediction nodes were determined by systematically varying the retinal location of the visual input and the eye position input over their respective domains. For each combination of visual and eye position input, the temporal response of each prediction node was recorded and averaged, over 60 time-steps,[5] to obtain the response values reported below. The specific details for individual experiments are explained with each experimental setup.

2.4 Analysis

We applied methods of quantitative analysis to the response properties of the prediction nodes in order to address 3 questions: firstly, to what degree can response properties be separated into visual and eye position components that interact multiplicatively? Secondly, what is the functional form of these response components, and how do they relate to the functional form of the input signals? Thirdly, how do response properties of model neurons compare to neurophysiological data recorded from real neurons? To address the last question, we chose curve fitting and regression analysis techniques that have previously been used in the analysis of neurophysiological data – meaning that a direct comparison of statistical properties of the results is possible. The first two questions are addressed by the nature of the methods we applied. For instance, Cassanello & Ferrera (2007) fitted a nonlinear function, consisting of the product of a Gaussian RF and a piece-wise linear GF, to the response data of individual cells in cortical area FEF (Frontal Eye Fields). Equation 11 describes the same function, except for the omission of a constant parameter accounting for the background activity of cortical cells (model prediction nodes do not display background activity in the absence of any input).

$$R(r_x, e_x) = \alpha_1 \exp\left(-\frac{(r_x-\alpha_2)^2}{2\alpha_3^2}\right)[1+\alpha_4 e_x]_+ \tag{11}$$

For each cortical cell, Cassanello & Ferrera (2007) extracted a set of parameter values α_i using MATLAB fitting routine `nlinfit` – with as independent variables the retinal location of visual stimulation r_x and the eye position e_x. The dependent variable was the firing rate averaged across all trials recorded for the same stimulus conditions. To assess the quality-of-fit, they plotted neural response versus predicted response for each combination of visual stimulus and eye position and calculated the coefficient of correlation (r_{nl}^2) between these two data sets. r_{nl}^2 is a measure for the amount of variance explained by the nonlinear model (Eq. 11), relative to the total variance in the response data. A value of 1 signifies that the model explains all the variance in the data; a value of 0 signifies that it does not explain any variance at all.

We applied the same analysis to the simulated response data of the prediction nodes to obtain r_{nl}^2 values. The higher the value of $r_{nl}^2$the more reliable the response data can be described as the product of two independent components: a Gaussian one and a linear-rectified one. A good

[5] See Sect. 2.1 for the justification of this value.

fit of Eq. 11 thus answers the first two questions raised at the start of this section. Based on visual inspection of response data and corresponding r_{nl}^2 values we developed the following classification: prediction nodes with values of $r_{nl}^2 > 0.95$ are said to be well-fitted by Eq. 11, whereas prediction nodes with $r_{nl}^2 < 0.95$ are said to be not well-fitted. Although the actual boundary is fuzzy rather than exact, in general this criterion served as a good indicator: response data with $r_{nl}^2 < 0.95$ had either non-Gaussian RFs, strongly nonlinear GFs, or a mixture of both.

We also generalised the analysis of Eq. 11 to experiments where both the retinal location of the visual stimulus and the eye position signal are variable in two dimensions (denoted by (r_x, r_y) and (e_x, e_y) respectively):

$$R(r_x, r_y, e_x, e_y) = \zeta_1 \exp\left(-\frac{(r_x - \zeta_2)^2 + (r_y - \zeta_3)^2}{2\zeta_4^2}\right)[1 + \zeta_5 e_x + \zeta_6 e_y]_+ \tag{12}$$

Several past physiological papers have investigated the linearity of GFs by applying linear regression analysis (e.g., Zipser & Andersen, 1988; Andersen et al., 1990; Galletti et al., 1995; Bremmer, Ilg, et al., 1997). This entails fitting either Eq. 13 to GF data where eye position is variable in one dimension (e.g., horizontal only) or Eq. 14 to data where eye position is variable in both horizontal and vertical directions.

$$R(x) = \gamma_1 + \gamma_2 x \tag{13}$$

$$R(x, y) = \delta_1 + \delta_2 x + \delta_3 y \tag{14}$$

In similar vein to r_{nl}^2 a coefficient of correlation r_l^2 between real and predicted GF data can be used as a measure of quality-of-fit for the linear models. For experiments where both horizontal and vertical eye position are variable, it is also possible to obtain *partial* correlation coefficients by regressing against one eye direction only (making partial models equivalent to Eq. 13). This allows to assess the degree to which horizontal or vertical eye position *alone* accounts for the variance in the GF data. We denote these partial coefficients with $r_{l,x}^2$ and $r_{l,y}^2$ for horizontal and vertical eye position respectively, and use $r_{l,xy}^2$ for the correlation coefficient obtained from the full model (Eq. 14).

We applied linear regression analysis to the GFs of prediction nodes to estimate to what degree the functional form of the GFs can be described as linear. Based on visual inspection of GF plots, graphical analysis of residuals and the corresponding r_l^2 values, we determined the following criteria for quality-of-fit: values of $r_l^2 > 0.95$ indicate a good fit, values between 0.8 and 0.95 are described as moderate, and values of $r_l^2 < 0.8$ indicate a poor fit. Good fits correspond to GFs that are largely linear and show significant modulation. Moderate fits correspond to GFs that display some linearity but have a non-linear component (e.g., saturation at the edges). Poor fits are indicative of strong nonlinearity or very weak modulation. The true boundaries are again fuzzy rather than exact, yet the criteria are indicative of the prototypical behaviour just described.

Slope parameters for the fitted line (Eq. 13) or plane (Eq. 14) can be obtained from the γ_i or δ_i parameters as follows:

$$sl_x = \frac{\gamma_2}{\gamma_1} \;\; or \;\; sl_x = \frac{\delta_2}{\delta_1}, sl_y = \frac{\delta_3}{\delta_1} \tag{15}$$

The direction of the gradient of the fitted plane is calculated as:

$$\theta_g = \arctan \frac{\delta_3}{\delta_2} \tag{16}$$

2.5 Code

Software, written in MATLAB, which implements the experiments described in this paper is available at `http://www.corinet.org/mike/code.html`.

3 Results

In this section we proceed as follows: we first present simulation results for prediction nodes with predetermined, fixed weights to explain how competition between nodes can give rise to multiplicative interactions between population-coded input signals. We then demonstrate that these synaptic weight distributions can indeed be learned using an unsupervised learning rule, and we statistically analyse various aspects of the resulting response properties of the prediction nodes. Finally, we look at how, at the network level, the response of the prediction nodes *tiles* the space defined by the input variables.

3.1 Competition Leads to Multiplicative Gain Modulation

In a first experiment we obtained response properties from prediction nodes in 4 networks with a predetermined, fixed weight matrix $\mathbf{W}$. The networks all had the same input and simulation parameters, but differed in the number of prediction nodes used. The number of error nodes equals by definition the number of input units (see Eq. 1) and hence was the same for all 4 networks. The visual and eye position input to the networks were both one-dimensional, a simplification that allows the full response properties of nodes to be easily visualised (Salinas & Abbott, 1996). The visual stimuli were generated by 61 input units with Gaussian response profiles (see Eq. 4). The Gaussian RFs had a standard deviation $\sigma = 6°$ and their peaks were spaced at $2°$ intervals from $-60°$ to $60°$. A subset of these response profiles and the corresponding population input signal generated for visual input $r_x = 0°$ can be seen in Fig. 2(a). Eye position was encoded by 9 sigmoid units with positive slope factor $T = +20°$ and 9 with negative slope factor $T = -20°$ (see Eq. 6). Inflection points of the sigmoids were spaced at $10°$ intervals between $-40°$ and $40°$. Figure 2(c) displays the response profiles with positive slope factor and the corresponding population input signal generated for eye position input $e_x = 0°$. Profiles with negative slope factor are mirror images of the ones shown.

For each prediction node we set weight values to be scaled versions of the population signal for specific values of the input variables r_x and e_x. Figure 3(a) shows a prediction node whose retinal weights are a scaled version of the visual input for $r_x = 0°$. Its eye position weights are a scaled version of the population signal generated by sigmoid units with positive and negative slope factor T for $e_x = -40°$. The weight values were normalised such that the sum of weights for each prediction node equals 1. In subsequent sections we will demonstrate that the learning rule generates weights that are similar in distribution and magnitude. The 4 networks differed in the number and weights of the prediction nodes. The first network (***N1***) contained only a single prediction node, i.e., one with weights as shown in Fig. 3(a). In the second network (***N2***), a second node was added with the same retinal weights but with eye position weights for the *opposite* eye position ($e_x = +40°$). The third network (***N3***) consisted of 13 prediction nodes with eye position weights for $e_x = -40°$ and with Gaussian retinal weights whose peaks were evenly spaced from $-60°$ to $60°$ in steps of $10°$. The last network (***N4***) contained 26 prediction nodes: 13 nodes as in ***N3***, and 13 nodes with the same distribution of retinal weights but with opposing eye position weights.

The response properties of all 4 networks were determined as explained in Sect. 2.3: the retinal location of the visual input r_x was varied from $-60°$ to $60°$ in steps of $1°$, and eye po-

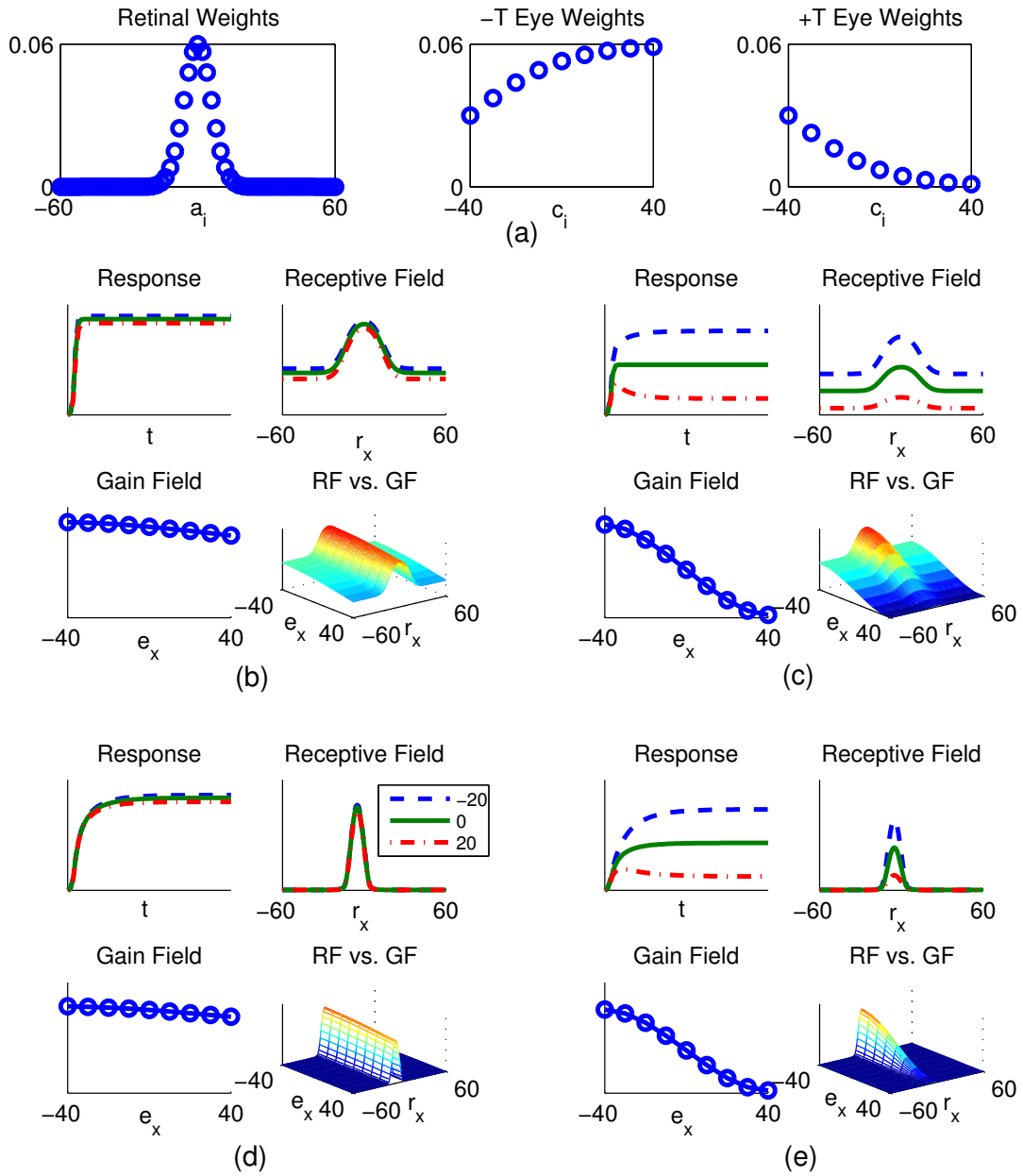

Figure 3: Response properties of a prediction node with predetermined weights, recorded in 4 different networks. (a) The weight values of the node are scaled versions of the population input for $r_x = 0°$ and $e_x = -40°$. (b) – (e) Temporal response, RF and GF response properties. The meaning of the different response graphs is described in the main text. (b) Response properties in a network consisting of that node only (***N1***). (c) Response properties in a network consisting of 2 nodes with the same retinal weights but opposing eye position weights ($e_x = -40°$ and $e_x = 40°$, ***N2***). (d) Response properties in a network consisting of 13 nodes with the same eye position weights but with the peaks of the retinal weights spread evenly between $-60°$ and $60°$ (***N3***). (e) Response properties in a network consisting of 26 nodes, 13 nodes as in ***N3***, and 13 nodes with the same retinal weights but opposing eye position weights (***N4***).

sition input e_x was varied from $-40°$ to $40°$ in steps of $10°$. For each combination of r_x and e_x the temporal response of the prediction nodes was recorded and averaged over 60 time-steps. Figures 3(b)–(e) display for the 4 different networks the response properties of the prediction node, with the weights shown in Fig. 3(a). For each case the graphs show the temporal response to the node's preferred visual stimulus for 3 different eye positions; the visual RF (i.e., the change in averaged response to varying visual input) for the same 3 eye positions; the GF (i.e., how the averaged response to the preferred visual stimulus is affected by eye position); and a 3D graph displaying the response measured for all combinations of visual and eye position stimuli. These figures demonstrate the crucial role of competition between the prediction nodes in shaping the response properties. For the single node in ***N1***, the response to the visual and eye position input combined additively. The response measured as a function of r_x (for constant e_x, the 'RF' curves) was not limited to part of the visual domain. For those visual stimuli that fell entirely outside the span of the retinal weights the node still recorded a response to the eye position signal. The effect of the eye position signal varied little with e_x, as is evident from the nearly-flat 'GF' curve and the planar component of the full response graph. In ***N2***, the two nodes had the same visual preference and received the same visual stimulus. There was hence no competition in the r_x domain, and the *shape* (but not the vertical offset) of the visual 'RF' curves remained similar to the ones seen in ***N1***. The opposing eye position preference of the two nodes led to competition in the e_x domain. Each node tried to prevent the other from representing the input, succeeding better in this regard when the eye position signal was closer to its own preferred eye position. The competition gave rise to a significant slope in the 'GF' curve, a sign of GM. However, the full response profile shows that the interaction between the visual and eye position sensitivities was neither simply additive nor purely multiplicative, but a mixture of both. The node in ***N3*** has a localised, bell-shaped visual RF. The competition between the 13 prediction nodes – equally distributed over the r_x domain – ensures that the "constant" response to the eye position input, observable over the entire visual domain in ***N1***, is suppressed for all but the node's visual RF proper. As in ***N1***, however, the absence of competition in the e_x domain gave rise to a nearly-flat 'GF' curve. In ***N4***, where the nodes competed in both the r_x and e_x domains, the visual RF was well-defined and bell-shaped, and the 'GF' curve was a monotonic function of e_x. The interaction between the two input variables was multiplicative: the shape of the overall response profile can be generated by multiplying together two independent response components: a Gaussian visual component and a monotonically-decreasing eye position component. In conclusion, the response profile of a prediction node does not only depend on the values of its weights, but is crucially dependent on the competition with neighbouring nodes. Moreover, a requirement for the emergence of multiplicative GM is that competition occurs over all of the input domains.

3.2 Unsupervised Learning of Multiplicative Gain Fields

3.2.1 One-Dimensional Gain Fields

We now turn to network training experiments to demonstrate that the weights postulated in Sect. 3.1 are easily learned, and to show that multiplicative GM is also robustly observed in these trained networks. The first experiment in this section used exactly the same input settings as the networks in Sect. 3.1: visual input was encoded by 61 Gaussian input units with standard deviation $\sigma = 6°$ and peaks distributed at $2°$ intervals from $-60°$ to $60°$. Eye position was encoded by 9 sigmoid units with positive slope factor $T = +20°$ and 9 with negative slope factor $T = -20°$. Inflection points of the sigmoids were spaced at $10°$ intervals between $-40°$ and $40°$.

The network contained 79 error nodes (by definition the same number as the number of input units – see Eq. 1) and 25 prediction nodes (a parameter that can be freely chosen – later experiments will clarify its effect). Training proceeded as explained in Sect. 2.3: the network was subjected to 30000 training epochs, each epoch consisting of a random combination of a visual stimulus and an eye position signal. For each combination, the retinal location of visual input r_x was selected with uniform probability from the interval $[-60°, 60°]$, and eye position e_x was chosen with uniform probability from $[-40°, 40°]$. After training, the response properties of the prediction nodes were determined, as in Sect. 3.1, by systematically varying the network's visual input r_x from $-60°$ to $60°$ in steps of $1°$ and eye position input e_x from $-40°$ to $40°$ in steps of $10°$.

A typical example of neural response properties after training is shown in Fig. 4(a), and the corresponding weights of this prediction node are shown in Fig. 4(b). The weights after training are indeed scaled versions of population input signals, and similar to the ones shown in Fig. 3(a). This pattern was confirmed for the other prediction nodes in the network. The response properties have virtually the same shape as shown in Fig. 3(e): the visual RF is a bell-shaped curve modulated by eye position. Its peak value decreases monotonically and almost linearly for eye position changing from left to right, but its shape and location are largely independent of eye position. To quantify this result we fitted Eq. 11 to the full (r_x, e_x) response data. The correlation coefficient r^2_{nl} between measured and fitted response is 0.994, indicating a very high quality-of-fit. The response properties of this node are thus well described as the product of an independent Gaussian RF and a linear-rectified eye-position GF. The width of the Gaussian RF is determined by the distance between the peaks of the retinal weights of neighbouring prediction nodes rather than by the width of the input signal. The estimated width of the Gaussian, $fw_{\frac{1}{2}} \approx 2.36 \times \alpha_3 = 10.71°$, is therefore slightly narrower than the width of the Gaussian input signals used to train the network ($fw_{\frac{1}{2}} = 2.36 \times \sigma = 14.16°$). We also fitted linear regression model Eq. 13 to the GF curve separately and found a correlation coefficient $r^2_l = 0.983$, a strong indication that the GF can indeed be described as linear.

To establish how common this phenomenon was we repeated the training procedure for 10 networks with identical parameters,[6] and fitted Eq. 11 to the response data of the resulting 250 prediction nodes. Values for r^2_{nl} ranged from 0.846 to 0.998, with a mean value of 0.986. Overall, 241/250 nodes displayed a good quality-of-fit ($r^2_{nl} > 0.95$). All of these 241 nodes developed Gaussian RFs, with the mean of the estimated Gaussian width $\overline{fw_{\frac{1}{2}}} = 11.05°$ and sample standard deviation $s = 1.37°$. Of the 9 remaining nodes, 3 had become *eye-position-only* nodes. They had very weak retinal weights and had lost visual responsiveness, resulting in an overall response profile that depends on eye position e_x only. Three nodes had *irregular* RFs: these nodes had relatively weak retinal weights but larger than for the eye-position only nodes; responsiveness was limited to some range of r_x values but the shape of the response was not Gaussian. The final 3 nodes were characterised by weak or virtually absent responses; their weights were small and undifferentiated. To investigate the linearity of the GFs, we fitted the linear regression model (Eq. 13) to the GFs of all prediction nodes with Gaussian RFs. Of 241 nodes, 172 were well-fitted by the linear model ($r^2_l > 0.95$); their GFs are mostly linear. Thirty five nodes had a moderate quality-of-fit ($0.80 < r^2_l < 0.95$). The GFs of these nodes have substantial linear components, but display some nonlinear saturation at one or both ends of the eye position range. Finally, 34 nodes were poorly fitted by the linear model ($r^2_l < 0.80$). These nodes are characterised by nonlinear GFs that peak at or near the centre of the eye-position

[6] Here and in subsequent experiments, 'identical parameters' means that all network setup and training parameters are the same, but that randomised quantities such as initial weights and the sequence of training stimuli are constructed anew for each repetition.

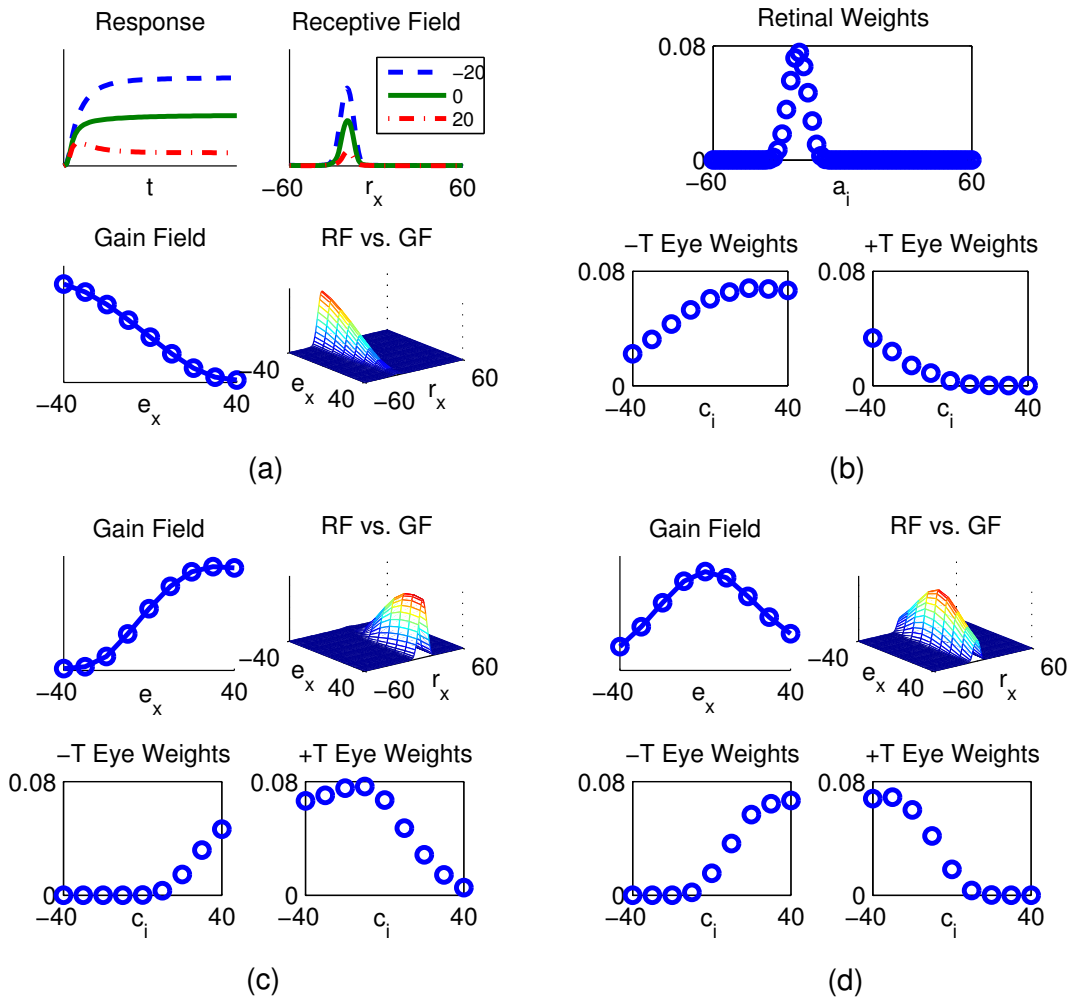

Figure 4: Response properties and weights after training with 1D Gaussian visual and 1D sigmoid eye position stimuli. (a) Temporal response, RF and GF for a single prediction node. The format of this figure equals the format used in Fig. 3. (b) Weights after training for the same prediction node. (c)–(d) GF properties and eye position weights of two additional nodes trained with steeper sigmoid eye position signals than in (a) (slope factors $T = \pm 10°$ as opposed to $T = \pm 20°$). All other simulation parameters were identical. 'Response' graphs, 'RF' graphs and retinal weights for these nodes are omitted as they are similar to the ones shown in (a). The vertical scale of all these and subsequent response graphs are the same as for Fig. 2.

range. However, the peaks are not very pronounced, and overall these nodes show relatively weak modulation in comparison with linear-GF nodes in the same experiment. ***Exp1*** of Table 1 summarises these results.

Table 1: Summary of quantitative analysis of experiments described in Sect. 3.2.1. ***Gaussian***: number of nodes with well-defined Gaussian RFs (out of a total of 250). ***Good***: GFs with good fit of the linear regression model (Eq. 13) ($r_l^2 > 0.95$). ***Moderate***: GFs with moderate fit ($0.8 < r_l^2 < 0.95$). ***Poor***: GFs with poor fit ($r_l^2 < 0.8$). ***Exp1***: primary experiment of Sect. 3.2.1. ***Exp2***: slope factor for sigmoids $T = \pm 10°$. ***Exp3***: noisy input. The number between brackets is the percentage of nodes in each category relative to the number of nodes in the first column.

	Gaussian	***Good***	***Moderate***	***Poor***
Exp1	241	172(71%)	35(15%)	34(14%)
Exp2	244	67(27%)	106(44%)	71(29%)
Exp3	239	163(68%)	36(15%)	40(17%)

These data show that training the PC/BC model with population input signals encoding two quantities (in this case retinal location r_x and eye position e_x) results in nodes that are responsive to both quantities in ways that suggest a multiplicative interaction of independent sensitivities. To further investigate the influence of the nature of the eye position signal on the response properties, we trained 10 more networks with identical parameters as in previous experiment, but with slope factors $T = \pm 10°$ (see Fig. 2(d)). Two representative GFs and their corresponding eye position weights from this experiment are shown in Figs. 4(c) and (d). In both cases the visual RF was Gaussian but the GFs display nonlinearities stronger than could be observed in the first experiment. The GF in Fig. 4(c) has a linear component over the central e_x range, but displays nonlinear saturation at either end. Given that the eye position input in this experiment is steeper (and so are the resulting eye position weights of this node), a steeper GF is perhaps not a surprise. However, the relationship between the nature of the eye position signal and the resulting GF is not always one-to-one: the GF of Fig. 4(d) peaks at the centre of the eye position range, even though the functional form of the eye position signal was sigmoid. The explanation for this behaviour can be found in the eye position weights: they are roughly the same shape as the population signal for $e_x \approx 0°$, instead of one of the extreme eye positions as in previous examples. The nature of the eye position signal therefore does not only control the steepness of the learned GFs, but it also affects the combinations of retinal and eye position weights that can be learned. We return to this issue in Sect. 3.3.

Analysis of the response properties for the node of Fig. 4(c) gave $r_{nl}^2 = 0.971$ and $r_l^2 = 0.946$, reflecting the saturation at the edges in an otherwise linear GF. For the node in Fig. 4(d) the values were $r_{nl}^2 = 0.825$ and $r_l^2 = 0.027$, reflecting the strong nonlinearity of the GF. Analysis of all 250 prediction nodes across the 10 networks reveals that the visual RF properties of the nodes are very similar to the ones in the previous experiment: 244/250 nodes developed well-defined Gaussian receptive fields. The quality-of-fit of the nonlinear model (Eq. 11) is somewhat lower than before: r_{nl}^2 values range from 0.653 to 0.995 with a mean of 0.965. This difference is largely due to more pronounced nonlinearities in many of the GFs, a trend confirmed by the analysis of the linear regression model (Eq. 13) – summarised in ***Exp2*** of Table 1. The 71 GFs with a poor fit of the linear model tend to peak near or at the centre of the eye position range, but the peaks are more pronounced than in the previous experiment (as shown

in Fig. 4(d)). There are also many more nodes with moderate fit of the linear model, displaying significant saturation at either end of the eye position range (as in Fig. 4(c)). This latter group is most likely the direct consequence of the steeper eye position signal, as discussed above.

Together, the results of the two preceding experiments display trends that have also been observed in physiological recordings. Various studies have shown a separability of visual and eye position sensitivities together with a mixture of linear and nonlinear GFs. For instance, Cassanello & Ferrera (2007) fitted the nonlinear model (Eq. 11) to the response of neurons in frontal cortical area FEF and found many nodes with a high quality-of-fit. r_{nl}^2 values for individual cells ranged from 0.40 to 0.98. For the entire population of cells analysed the model explained 94.76% of the total variance in the response data. In parietal cortical areas LIP and 7a, Andersen et al. (1990) estimated roughly one third of gaze-modulated cells to have linear GFs, one third of cells to have linear GFs with a nonlinear component, and one third of cells to have nonlinear GFs. In another study in posterior parietal cortex, Andersen & Mountcastle (1983) reported four types of GFs: two linear, monotonic types (with either positive or negative slope); one peaking at the centre of the eye position range; and one the inverse of a central peak (i.e., a central depression). Our experiments show that by training the PC/BC model with positive and negative sloping sigmoid eye position signals the first three of these GF types can be obtained. The fourth type - the central depression - was rarely observed in our experiments: eye position weights for the central-depression node would need to encode both extremes of the eye position range, a rather unlikely event given the training procedure. It is much more likely that a centrally-peaking node is flanked by two separate nodes in the e_x dimension – one with left and one with right sloping GF. The proportion of GFs of each type depends on the nature of the eye position signal (as shown by the differences in results across these two experiments), but also on the range of values for which the eye position effect is tested: many GFs with saturation near $e_x = -40°$ or $e_x = 40°$ would be classified as linear if tested over a smaller eye position range, e.g., $[-20°, 20°]$.

We conducted a third experiment to investigate how the training procedure is affected by noise, in recognition of the fact that neural population signals are often very noisy. We applied the noise model of Eq. 7 to each input value h_i during training. ρ is a normally distributed random variable with zero mean and standard deviation $\sigma = \frac{1}{3}$. This means that the strength of each input can be modulated up or down by as much as 100% of its original value ($3\sigma = \pm 1$). Despite the strong noise in the input, the response properties for 250 prediction nodes, summarised in ***Exp3*** of Table 1, show no significant differences with the noise-free case. We also applied the same noise model *after* training to determine how noise propagates through the network. We found that noise measured in the output was one to two orders of magnitude smaller than noise in the input. These experiments thus demonstrate that similar results can be obtained under noisy conditions. This *noise suppression* characteristic has also been observed in related neural models (Salinas & Abbott, 1996; Pouget, Deneve, & Duhamel, 2002).

3.2.2 Two-Dimensional Gain Fields

In this section we demonstrate that the emergence of multiplicative visual/eye position interactions in the PC/BC model generalises to the more natural configuration of 2D visual and compound eye position signals. Visual input in this section is generated by input units with 2D Gaussian response profiles - as defined by Eq. 5. The centres of the Gaussian input units are spaced $5°$ apart in a field of vision ranging from $-60°$ to $60°$ in horizontal direction and from $-40°$ to $40°$ in vertical direction (resulting in a total of 425 visual input units). We increased σ from $6°$ to $16°$ to keep the ratio of the area covered by an individual Gaussian input unit to the

full area of the visual field approximately constant (see Sect. 2.3).[7] Experiments not reported here indicate that neither the exact value of this ratio nor the value of σ itself are critical for the reliable emergence of multiplicative gain modulation. A typical example of the population signal encoding visual input can be seen in Fig. 2(b). Eye position is encoded by two populations of sigmoidal input units independently encoding horizontal and vertical eye position. Each of the two populations contains the same number and type of input units as in previous section (with slope factors $T = \pm 10$), resulting in a total of 36 eye position units.

Training and determination of the response properties followed the same procedure as outlined before. A network of 461 error nodes (one for each input) and 40 prediction nodes (a freely chosen parameter) was trained for 30000 epochs. Training stimuli were constructed by uniformly sampling (r_x, r_y) values from the intervals $([-60°, 60°], [-40°, 40°])$ and (e_x, e_y) from $([-40°, 40°], [-40°, 40°])$. After training, response properties were determined by systematically varying visual input in steps of $5°$ and eye position input in steps of $20°$ over the same ranges used for training. A typical example of neural response properties after training can be seen in Fig. 5(a). The graph on the left shows the peak-shaped RF for the node's preferred eye position. The graph on the right shows the GF for the preferred visual stimulus of the same node. The GF is almost planar, with its preferred eye position down and to the right. Figure 5(b) shows 2 additional GFs from the same experiment: the left GF has a planar shape but a different orientation than the one in Fig. 5(a); the right GF has a non-planar shape.

The results obtained from fitting the nonlinear (Eq. 12) and linear regression models (Eq. 14) to the response properties of the 3 nodes from Fig. 5 are summarised in Table 2. These values quantify the observations made above: ***GF1*** and ***GF2*** have high r^2_{nl} values, indicating that these response properties can be regarded as the product of a Gaussian and a planar component. Both nodes are also well-fitted by the linear model. The difference between ***GF1*** and ***GF2*** is apparent from the *partial* correlation coefficients: for ***GF1*** regression on individual predictors e_x and e_y each explains about half the variance in the GF data, but for ***GF2*** all variance is explained by vertical eye position e_y only. The slopes and gradient angles obtained from the estimated coefficients of the regression model (Eqs. 15 and 16) confirm the orientation differences of ***GF1*** and ***GF2***. ***GF3*** only just fails the mark as being well-fitted by Eq. 12 as its r^2_{nl} value is just below the cutoff value of 0.95. This is due to its GF not being planar, as confirmed by its low $r^2_{l,xy}$ value. In all three cases the visual RF was a well-defined 2D Gaussian curve, with estimated $fw_{\frac{1}{2}}$ values well below the width of the Gaussian input signals ($37.8°$ – the reasons for this difference are explained in Sect. 3.1).

We repeated the training procedure for 10 networks with identical parameters, and quantitatively analysed the resulting pool of prediction nodes by fitting Eq. 12 and Eq. 14 to the response properties. Values for r^2_{nl} ranged from 0.934 to 0.996, with a mean value of 0.981. 397/400 nodes displayed a good quality-of-fit ($r^2_{nl} > 0.95$). The three nodes that failed to make the mark did so because of the strongly nonlinear shape of their GFs. All nodes developed Gaussian RFs, with the mean of the estimated Gaussian width $\overline{fw_{\frac{1}{2}}} = 25.11°$ and sample standard deviation $s = 3.55°$. For a total of 400 nodes, 202(51%) GFs were well-fitted by the planar model, 152(38%) displayed a moderate fit, and 46(11%) were poorly fitted. Detailed analysis of the poorly-fitting nodes showed that some had non-planar GFs such as the one shown in Fig. 5(b). However, many nodes in this group were only weakly modulated by eye position. Given the small total variance in the GF data for such nodes, small deviations from a perfectly planar shape strongly reduce the coefficient of correlation. Overall, GFs did not display the amount and variety of nonlinearities observed in the 1D experiment with the same sigmoid

[7] For these values of σ the ratio for the 1D case can be calculated from Eq. 8 to be 0.1180, for the 2D case it is calculated from Eq. 9 as 0.1166.

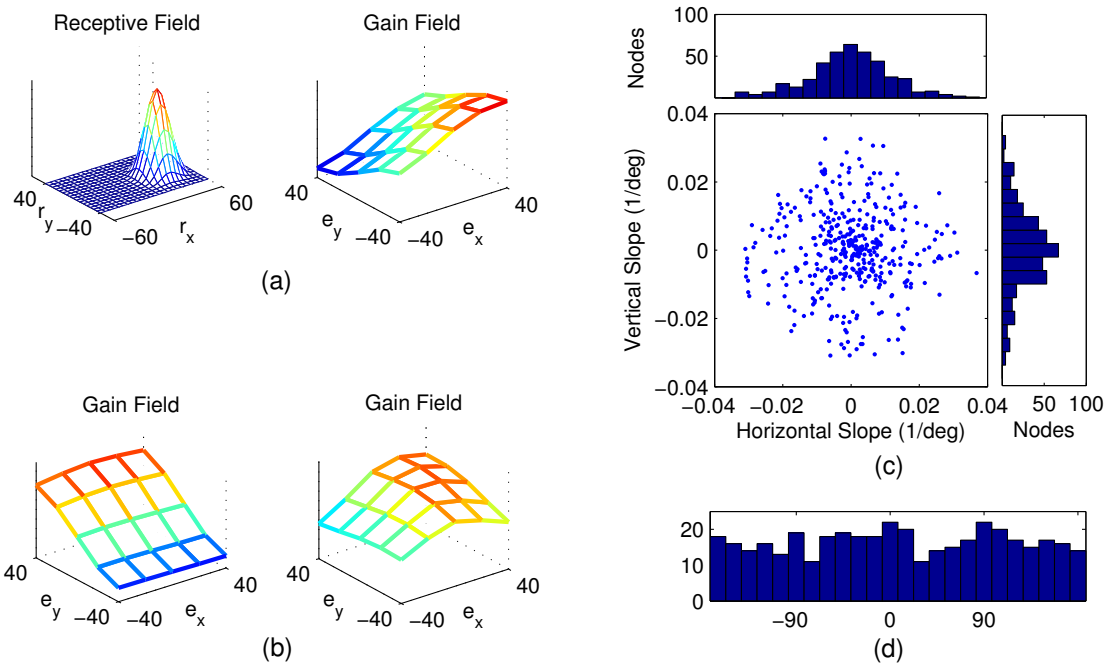

Figure 5: RF and GF properties after training with 2D Gaussian visual and two 1D sigmoid eye position signals. (a) Response properties of a single prediction node in the network. Left: visual RF measured for the node's preferred eye position. Right: GF measured for the node's preferred visual stimulus. (b) Two additional GFs measured from nodes in the same experiment, with different shapes. (c) Distribution of horizontal and vertical slopes of the fitted regression planes for 400 prediction nodes. Central: scatter plot of horizontal versus vertical slope values. Each node is represented by a single point in the graph. Nodes in the central area of this plot are only weakly modulated by eye position. Top: histogram of horizontal slope values. Right: histogram of vertical slope values. (d) Distribution of the gradient angles (θ_g) of the regression planes. A value of 0° corresponds to a rightwards-up, horizontal direction.

Table 2: Analysis of the response properties of the nodes depicted in Fig. 5. r^2_{nl}: correlation coefficient for the nonlinear regression model (Eq. 12). $fw_{\frac{1}{2}} \approx 2.36 * \zeta_4$, ζ_4 is estimated using Eq. 12. $r^2_{l,xy}$: correlation coefficient obtained for the linear regression model (Eq. 14) with predictors (e_x, e_y). $r^2_{l,x}$,$r^2_{l,y}$: correlation coefficients for single predictor e_x or e_y. sl_x, sl_y: horizontal and vertical slopes obtained from fitted planes (Eq. 15). θ_g: angle of gradient of fitted planes (Eq. 16). ***GF1***: node of Fig. 5(a). ***GF2***: left node of Fig. 5(b). ***GF3***: right node of Fig. 5(b).

	r^2_{nl}	$fw_{\frac{1}{2}}$	$r^2_{l,xy}$	$r^2_{l,x}$	$r^2_{l,y}$	sl_x	sl_y	θ_g
GF1	0.988	22.7°	0.976	0.488	0.488	0.0107	−0.0107	−45°
GF2	0.980	25.0°	0.984	~ 0	0.984	~ 0	0.0190	90°
GF3	0.946	24.2°	0.431	0.397	0.034	0.0046	0.0014	16°

slope factors $T = \pm 10$ (see Fig. 4(b) and ***Exp2*** in Table 1). One influential factor underlying the distinctive results across these two experiments is the strong difference in the balance of the total activation coming from the two different input sources – i.e., the ratio of the summed strength of the population signals generated by visual and eye position input units. For the 1D experiments the ratio is 0.83 for $r_x = 0°$, reducing to half this value at either end of the r_x training interval. The ratio is independent of e_x as the sum of the total activation of positive and negative-sloping sigmoid signals is the same for all e_x values. For the 2D experiment the ratio is 3.54 for $(r_x, r_y) = (0°, 0°)$, reducing to a quarter of that value for extreme (r_x, r_y) values. Yet, multiplicative modulatory interactions reliably emerged even for these strongly different signal strength balances.

Horizontal and vertical eye position in this experiment are encoded independently by separate populations of sigmoidal input units. However, as is apparent from the example shown in Fig. 5(a), the resulting GFs do not necessarily have only vertical or horizontal gradient orientations. We analysed the orientation distribution of the GFs by analysing the orientation of the fitted planes for all 400 prediction nodes. Fig. 5(c) shows the distribution of horizontal and vertical slopes obtained from the parameters estimated through regression analysis (see Eq. 15). Both horizontal and vertical slope distributions are normal in shape. We also calculated the distribution of the direction of the GF gradients (see Eq. 16). Fig. 5(d) shows that all orientations are quite evenly represented. Excluding the 46 poorly-fitted nodes from this analysis did not fundamentally change the results. Its largest effect was to reduce the cluster of nodes with slope values near the origin of Fig. 5(c), confirming our earlier observation that many of the poorly-fitted nodes are not strongly modulated by eye position.

Similar physiological observations regarding GF slopes and gradient orientations have been made in several cortical areas: Andersen et al. (1990) found a fairly even distribution of GF gradient orientations in parietal cortical areas LIP and 7a. Bremmer, Ilg, et al. (1997) obtained similar results in extrastriate areas MT and MST: they reported normal distributions for horizontal and vertical GF slope values, and a uniform distribution for GF gradient orientations (the format of Fig. 5(c) is based on, and the results are similar to, Fig. 9 in (Bremmer, Ilg, et al., 1997)). In conclusion, the results described in this section demonstrate that the emergence of multiplicative GFs in the PC/BC model extends to configurations with 2D visual and compound eye position signals. Many GFs are planar, and they can have any orientation – even if eye position input only contains signals encoding the cardinal directions.

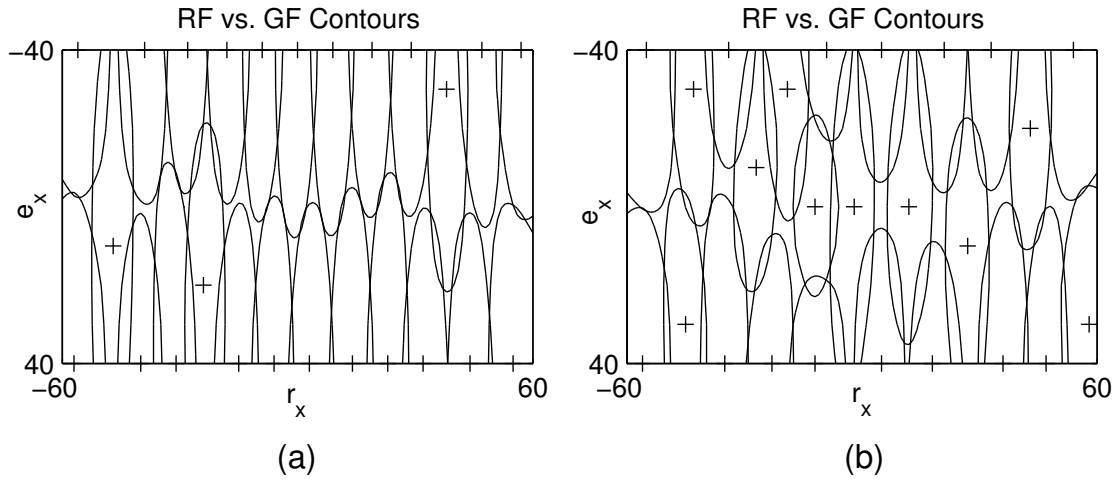

Figure 6: Prediction node responses tile the input space. (a) Contours at half the maximum response strength for the 25 prediction nodes of one network from the primary experiment of Sect. 3.2.1 (***Exp1*** of Table 1 – sigmoid slope factors $T = \pm 20°$). Cross-hairs indicate the location of the maximum response for each node; when the maximum coincides with e_x values at the edges, the cross-hairs may look like tick marks of the axes - but no actual tick marks were plotted. (b) Same graph as (a), but for a network from the second experiment in Sect. 3.2.1 (***Exp2*** of Table 1 – slope factors $T = \pm 10°$).

3.3 Prediction Node Responses Tile the Input Space

In Sect. 3.2 we focused on the response properties of individual prediction nodes. We demonstrated that – under a wide range of conditions – multiplicative GFs with linear as well as nonlinear profiles arise through unsupervised learning in the PC/BC model. In this section we investigate how the PC/BC network as a whole represents input. In all of our experiments populations of input units encode variables that are defined in Cartesian coordinates. We define the *input space* as the multi-dimensional space spanning the ranges of those different variables. In Sect. 3.2.1 a 1D visual and 1D eye position range were combined to define a 2D input space. The experiment in Sect. 3.2.2 defined a 4D input space (2D visual and two 1D eye position signals). Because 2D input spaces are easier to visualise, we return in this section to examples using 1D visual and eye position signals.

We explained in Sect. 2.1 that prediction nodes compete to represent input, and demonstrated in Sect. 3.1 that the competition gives rise to multiplicative GM. Two additional aspects of the training procedure have important implications: firstly, the *self-normalising* effect of the learning rule (see Sect. 2.1) has as consequence that individual nodes represent only *parts* of the input space; secondly, the topographical organisation of input units that encode Cartesian variables ensures that the section of the input space represented by each node is most often *contiguous*. The latter is caused by the fact that the weights, through training, become scaled versions of the population input signals for particular values of the input variables. When the response profiles of all prediction nodes in the network are plotted together, the result is typically a rudimentary *tiling* of the input space. Figure 6 shows the tiling for two networks of Sect. 3.2.1: one selected from the networks in ***Exp1*** (with sigmoid slope factors $T = \pm 20°$) and one from ***Exp2*** (with $T = \pm 10°$). The graphs show contour plots at half of the maximum response for each prediction node; the cross-hairs indicate the location of each maximum.

In general, the tiling of the input space depends on its size, the nature of the input signals and the number of prediction nodes in the network. The qualitative differences between the two graphs of Fig. 6 provide an illustrative example: the network in Fig. 6(a) was trained with shallow sigmoid eye position signals. It contains an almost regular tiling that divides the plane roughly in two for eye position. The network shown in Fig. 6(b), trained with steeper sigmoid input, contains several nodes with peak-shaped GFs (as the GF shown in Fig. 4(d)). The steeper sigmoid eye position signal allowed some nodes to be "squeezed" in between nodes with left and right sloping GFs. This divides the plane in three in the e_x dimension for at least some r_x values. In the r_x dimension for the two graphs, node responses cover a width somewhat smaller than the width of the Gaussian input. However, responses of some left and right sloping nodes are slightly wider in Fig. 6(b) as several nodes in that network have been squeezed into the middle of the input space, thereby reducing the competition at the edges of the e_x range. In networks with far fewer prediction nodes, responses would be "stretched" to cover the r_x dimension – given that self-normalisation drives the sum-of-weights for each prediction node towards the same, bounded value (i.e., a value of one). There would not be enough nodes to fill the input space with opposing left/right configurations in the e_x dimension. Indeed, networks with 10 prediction nodes but otherwise identical parameters as in the primary experiment of Sect. 3.2.1 consistently generate 8 nodes with wide, purely visual bell-shaped RFs, and 2 eye-position-only nodes – one with negative slope and one with positive slope. The two input signals thus remain separate in such a network, and are simply transmitted from input to output. As the number of prediction nodes increases beyond the 25 used in these examples the competition to dominate parts of the input space becomes fiercer. This results in some nodes being "squeezed out" of the representation altogether: they do not significantly respond to any combination of r_x and e_x. Their weights are small and quite evenly distributed over all inputs.

One way of interpreting the tiling of the input space is as a rudimentary set of *basis functions*. Basis functions were proposed as a generic mathematical framework to explain how gain-modulated signals in cortex can be used to compute coordinate transformations from one frame of reference to another (Pouget & Sejnowski, 1997; Pouget & Snyder, 2000; Pouget et al., 2002; Salinas & Sejnowski, 2001). For instance, in the basis function framework the combination of an eye-centred visual signal with an eye-position signal can be used – under a wide range of conditions – to calculate an eye-independent, head-centred visual signal. Pouget & Sejnowski (1997) showed that a set of functions consisting of the product of two basis functions (e.g., a Gaussian and a sigmoid, or two Gaussian functions) itself forms a basis function set for the combined variable space but only if all (or a sufficient number of) product combinations are present. The tilings of Fig. 6 form incomplete basis function sets, and would allow to perform only very rough approximations to coordinate transformations. However, different eye position signals can generate more fine-grained tilings that constitute a more complete basis function set. As an example we trained a network with an input configuration similar to the one used in (Pouget & Snyder, 2000) and (Pouget et al., 2002): both the visual and eye position signals were generated by identical populations of input units with Gaussian response profiles. We used 31 visual inputs with $\sigma = 6°$ spaced $2°$ apart in the interval $[-30°, 30°]$, and an identical population of eye position units. A network with 40 prediction nodes and 62 error nodes was trained for 30000 epochs. After training the response properties were determined by systematically varying r_x and e_x from $-30°$ to $30°$ in steps of $2°$. A representative example of the response properties of a single prediction node can be seen in Fig. 7(a). Both the resulting RF and GF are Gaussian. The "RF vs. GF" plot shows that the full response profile of the node is a 2D Gaussian curve, i.e., the product of two 1D Gaussian curves. This result was repeated throughout the network. The corresponding tiling of the input space is shown in Fig. 7(b). The result is an almost uniformly distributed grid of nodes that represent specific combinations of visual and eye position

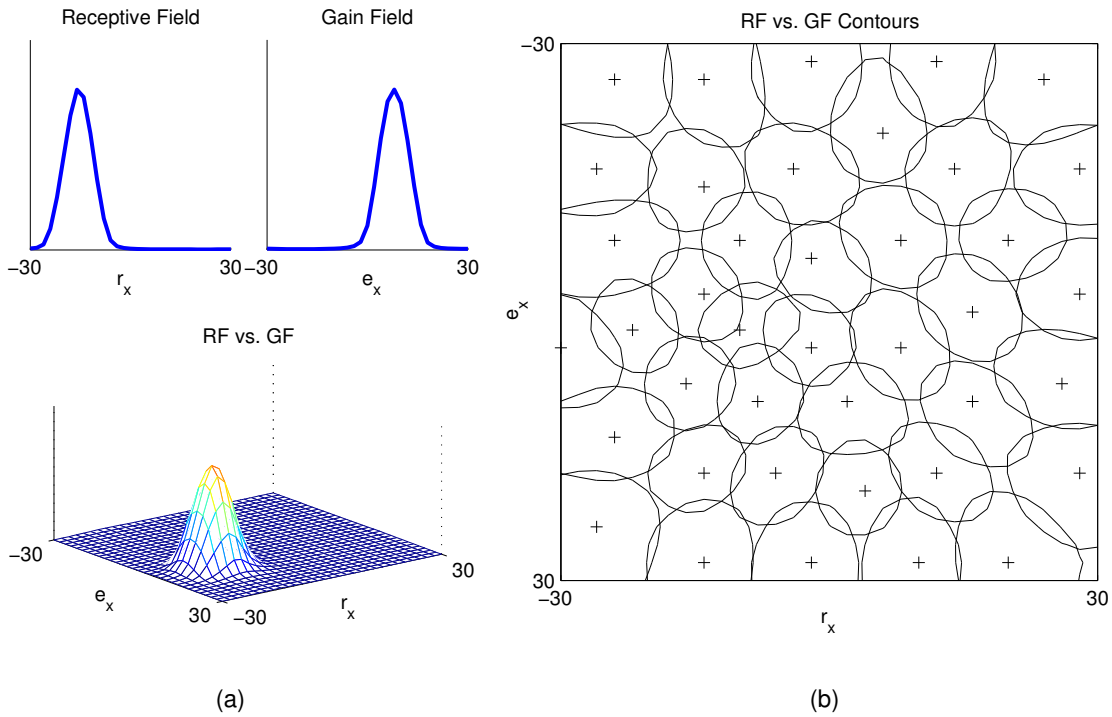

Figure 7: RF and GF properties after training with identical 1D Gaussian visual and eye position signals. (a) Response properties for a single prediction node. (b) Contours at half the maximum response strength for the 40 prediction nodes in the network. The cross-hairs indicate the location of the maximum response for each node.

input. As such, the layer of prediction nodes in the PC/BC model has response properties that are equivalent to the intermediate basis function layer in (Pouget & Snyder, 2000; Pouget et al., 2002).

4 Discussion

In Sect. 3.2 we demonstrated that GM emerges robustly and under a wide range of conditions in the PC/BC network. We showed examples of how the weights of individual prediction nodes, through training, take on the shape of population input signals for particular values of the input variables. In Sect. 3.1 we demonstrated that the response of a node not only depends on the values of its weights, but that it is significantly shaped by the competition with other nodes. In particular, multiplicative GM is only observed in the response profile of individual cells if competition in the network acts over all input domains. Section 3.3 demonstrated that this is exactly what the training procedure achieves: the entire input space is represented and tiled by the response of the prediction nodes.

4.1 Interpretation of the Results

The statistical analysis of Sect. 3.2 indicated that the response of the prediction nodes can often be decomposed into visual and eye position components which interact multiplicatively. All inputs to the PC/BC network, whether they arise from visual or eye position units, are treated

equally in Eq. 1. This makes the multiplicative interaction between the input variables, measured in the response of the prediction nodes, an emergent property of the network. Note that, although Eqs. 1 and 2 contain division and multiplication operations, these do not constitute a direct multiplication of the input variables. They are more akin to the type of operations found in the divisive normalisation model of Heeger (1992). The shapes of the GFs are the result of competition between prediction nodes. Indirectly, they depend on the conditions generated by the training stimuli: the nature of the eye position signal (sigmoid or Gaussian); in case of a sigmoid eye position signal, its steepness; the number of prediction nodes available to fill the input space, etc. These differences in input conditions give rise to differences in the ratios of GF types (linear versus nonlinear; monotonic versus peaked) reminiscent of discrepancies in the physiological results found in different cortical areas. For instance, whereas GFs appear to be mostly linear in cortical areas V3 (Galletti & Battaglini, 1989) and FEF (Cassanello & Ferrera, 2007), nonlinear GFs have been reported for areas 7a and LIP (Andersen & Mountcastle, 1983; Andersen et al., 1990) and V6A (Galletti et al., 1995). The model thus predicts how different local factors – e.g., origin and number of input connections arriving in a patch of cortex, the signal they carry, etc. – may contribute to shaping the response properties of cortical cells across different cortical areas. Our results also demonstrate that GFs can adopt shapes that are not present in the input signal: e.g., peak-shaped GFs in networks trained with sigmoid eye position (Sect. 3.2.1); planar GFs with all orientations in networks trained with horizontal and vertical eye position signals only (Sect. 3.2.2).

One interpretation of the operation of the PC/BC model is that the training procedure extracts correlations between the input signals from the training set. In other words, prediction nodes learn to associate inputs that frequently co-occur. We showed examples of how, during training, weights take on the shape of the population input signals for particular values of the input variables (see Fig. 4). This results in nodes that are maximally sensitive to conjunctions of those input values.

A second interpretation of PC/BC network operation is that prediction nodes compete to represent input. They do so by attempting to suppress *inputs* to neighbouring nodes via divisive feedback to the error nodes. It is this input suppression that is the cause of the nonlinear, multiplicative interactions between two or more population signals. An individual prediction node is well-tuned to a certain visual stimulus and eye position. If the same visual stimulus is presented as input together with a strongly different eye position signal then other nodes with the correct combination of sensitivities will suppress the input to the first node, thereby reducing its response to that part of the input to which it is well-tuned. On a network level, the competition between prediction nodes results in a tiling of the input space in which individual prediction nodes occupy roughly equal volumes. The population as a whole tries to fully tile the input space with response activity. With suitable input signals this process can give rise to a network representation that forms a basis function set (Sect. 3.3). It could therefore form part of a larger network performing complex, nonlinear computations on its input.

4.2 Comparison with Related Models

A number of neural network models have previously been proposed to study various aspects of GM. They can roughly be divided into three groups:

1. Models that have multiplicative interactions between different input signals built into the response equations of single nodes (Cassanello & Ferrera, 2007; Droulez & Berthoz, 1991; Pouget & Sejnowski, 1997; Quaia, Optican, & Goldberg, 1998; Salinas & Abbott, 1995; Siegel, 1998). These models generally serve to demonstrate how gain-modulated

responses can be used to perform a specific computation, or simulate a particular physiological phenomenon.

2. Models in which multiplicative interactions between different input signals are an emergent property of the network, but where the network structure (i.e., synaptic weights) are predetermined (Salinas & Abbott, 1996; Deneve, Latham, & Pouget, 2001).

3. Feedforward neural networks trained with a supervised learning algorithm (e.g., the back-propagation algorithm), where the teacher signal forces the network to learn to perform a certain nonlinear computation (Keith & Crawford, 2008; White & Snyder, 2004; Xing & Andersen, 2000a,b; Zipser & Andersen, 1988). The hidden layer neurons of these networks have frequently been shown to display nonlinear interactions between visual and eye position input.

The PC/BC model differs from all three categories as multiplicative GM is an emergent property of the network (unlike 1.), but it is obtained through learning (unlike 1. and 2.) using an *unsupervised* training procedure (unlike 3.).

Of all the above, the model of Salinas & Abbott (1996) is conceptually closest to the PC/BC model proposed here. In their network, the population as a whole performs a multiplication on inputs that are simply added at the single-neuron level. The multiplication is an emergent property of the network, achieved through recurrent connections that are partly excitatory and partly inhibitory. Interestingly, suppression in this network occurs between differently tuned neurons. Contrarily, in the PC/BC network suppression occurs between prediction nodes with similarly tuned responses. The synaptic weights in the Salinas & Abbott (1996) network are pre-specified, and there is no mention of how they could be obtained through unsupervised learning.

The PC/BC network is capable of generating a set of basis functions, making the work by Pouget and co-workers on networks with basis function units also of relevance. There are, however, some conceptual differences. In (Pouget & Sejnowski, 1997) a multiplicative interaction between input signals was programmed into the equation governing the basis function units. The resulting model was used to demonstrate the type of computations that can be performed using basis functions. Deneve et al. (2001) subsequently proposed a network where the neural activities were determined by a set of coupled non-linear equations that did not contain a direct multiplication of input signals. Deneve et al. (2001) did not focus on GM in particular, and present only two examples of GFs. In the absence of recurrent connections in the network, the reported GM appears to be additive rather than multiplicative. With recurrent connections, the interaction between the different input signals is nonlinear, but it is unclear whether it is truly multiplicative. Further work (e.g., Pouget et al., 2002; Avillac, Deneve, Olivier, Pouget, & Duhamel, 2005) also focussed on the type of computations and coordinate transformations that can be achieved using basis function networks. We see these models as complementary to the work presented here, in which we focussed on the emergence of GFs rather than on what can be computed with them.

4.3 Further Work

A natural extension of the work presented here would be to investigate what happens when eye position signals are repeatedly applied to visual stimuli in successive stages of a hierarchical PC/BC network. Indeed, Galletti et al. (1995) suggested that nonlinear GFs in parietal cortex could be the result of a hierarchical network in which eye position signals are re-applied to neural responses that display already linear GFs. Although we have demonstrated that, under

some conditions, nonlinear GFs can also appear in a single-area PC/BC network, the resulting peak-shaped GFs do not show strong localisation. A multi-area PC/BC network may be able to generate more strongly localised GFs in later stages of the hierarchy. In similar vein, the fine-grained tiling of Fig. 7 may arise as the final stage of a hierarchical network that starts with the more rudimentary tilings of Fig. 6. Another interesting research avenue would be to investigate what happens when postural signals from multiple sources (e.g., head, body) are combined with multiple sensory (e.g., visual, auditory) signals in a multi-stage PC/BC model. We expect that coordinate transformations and partially-shifting RFs (Pouget et al., 2002) will arise in such networks.

The tiling of the input space in Fig. 7 fulfills the necessary requirements of a basis function set: a product set of Gaussian curves tiling the entire input space (Pouget & Snyder, 2000). The next step would be to investigate if the tiling is fine-grained enough to perform actual computations such as coordinate transformations. This can be done by placing this single-area PC/BC network in a larger network and see how it compares to statistically optimal estimation methods, an analysis that was carried out for the recurrent basis function network in (Deneve et al., 2001).

Conclusion

We presented simulation results to demonstrate that gain modulation can reliably appear in a predictive-coding model of cortical function using an unsupervised training procedure. The behaviour of the model is in good agreement with various physiological results. Moreover, under the right input conditions the network learns to represent a basis function set, making this single-area model a candidate for use in multi-stage networks performing coordinate transformations and other nonlinear computations.

Acknowledgements

This work was funded by EPSRC Research Grant EP/D062225/1.

References

Andersen, R. A., Bracewell, R. M., Barash, S., Gnadt, J. W., & Fogassi, L. (1990). Eye position effects on visual, memory, and saccade-related activity in areas LIP and 7a of macaque. *Journal of Neuroscience*, *10*(4), 1176-96.

Andersen, R. A., & Mountcastle, V. B. (1983). The influence of the angle of gaze upon the excitability of the light-sensitive neurons of the posterior parietal cortex. *Journal of Neuroscience*, *3*(3), 532-48.

Avillac, M., Deneve, S., Olivier, E., Pouget, A., & Duhamel, J. R. (2005). Reference frames for representing visual and tactile locations in parietal cortex. *Nature Neuroscience*, *8*(7), 941-9.

Barlow, H. (1994). What is the computational goal of the neocortex? In C. Koch & J. L. Davis (Eds.), *Computational neuroscience series; large-scale neuronal theories of the brain* (p. 1-22). Cambridge, MA: MIT Press.

Bremmer, F., Distler, C., & Hoffmann, K. P. (1997). Eye position effects in monkey cortex. II. pursuit- and fixation-related activity in posterior parietal areas LIP and 7a. *Journal of Neurophysiology*, *77*(2), 962-77.

Bremmer, F., Ilg, U. J., Thiele, A., Distler, C., & Hoffmann, K. P. (1997). Eye position effects in monkey cortex. I. visual and pursuit-related activity in extrastriate areas MT and MST. *Journal of Neurophysiology*, *77*(2), 944-61.

Cassanello, C. R., & Ferrera, V. P. (2007). Computing vector differences using a gain field-like mechanism in monkey frontal eye field. *Journal of Physiology*, *582*(Pt 2), 647-64.

De Meyer, K., & Spratling, M. W. (2009). A model of non-linear interactions between cortical top-down and horizontal connections explains the attentional gating of collinear facilitation. *Vision Research*, *49*(5), 553-68.

Deneve, S., Latham, P. E., & Pouget, A. (2001). Efficient computation and cue integration with noisy population codes. *Nature Neuroscience*, *4*(8), 826-31.

Droulez, J., & Berthoz, A. (1991). A neural network model of sensoritopic maps with predictive short-term memory properties. *Proceedings of the National Academy of Sciences of the USA*, *88*(21), 9653-7.

Friston, K. (2005). A theory of cortical responses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *360*(1456), 815-836.

Galletti, C., & Battaglini, P. P. (1989). Gaze-dependent visual neurons in area V3A of monkey prestriate cortex. *Journal of Neuroscience*, *9*(4), 1112-25.

Galletti, C., Battaglini, P. P., & Fattori, P. (1995). Eye position influence on the parieto-occipital area PO (V6) of the macaque monkey. *European Journal of Neuroscience*, *7*(12), 2486-501.

Heeger, D. J. (1992). Normalization of cell responses in cat striate cortex. *Visual Neuroscience*, *9*(2), 181-97.

Jehee, J. F. M., Rothkopf, C., Beck, J. M., & Ballard, D. H. (2006). Learning receptive fields using predictive feedback. *Journal of Physiology (Paris)*, *100*(1-3), 125-132.

Keith, G. P., & Crawford, J. D. (2008). Saccade-related remapping of target representations between topographic maps: a neural network study. *Journal of Computational Neuroscience*, *24*(2), 157-78.

Kilner, J., Friston, K., & Frith, C. (2007). Predictive coding: an account of the mirror neuron system. *Cognitive Processing*, *8*(3), 159-66.

Mumford, D. (1992). On the computational architecture of the neocortex 2. the role of cortico-cortical loops. *Biological Cybernetics*, *66*(3), 241-251.

Murray, S. O., Schrater, P., & Kersten, D. (2004). Perceptual grouping and the interactions between visual cortical areas. *Neural Networks*, *17*(5-6), 695-705.

Pouget, A., Deneve, S., & Duhamel, J. R. (2002). A computational perspective on the neural basis of multisensory spatial representations. *Nature Reviews Neuroscience*, *3*(9), 741-7.

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

Pouget, A., & Snyder, L. H. (2000). Computational approaches to sensorimotor transformations. *Nature Neuroscience*, *3 Suppl*, 1192-8.

Quaia, C., Optican, L. M., & Goldberg, M. E. (1998). The maintenance of spatial accuracy by the perisaccadic remapping of visual receptive fields. *Neural Networks*, *11*(7-8), 1229-1240.

Rao, R. P. N., & Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, *2*(1), 79-87.

Salinas, E., & Abbott, L. F. (1995). Transfer of coded information from sensory to motor networks. *Journal of Neuroscience*, *15*(10), 6461-74.

Salinas, E., & Abbott, L. F. (1996). A model of multiplicative neural responses in parietal cortex. *Proceedings of the National Academy of Sciences of the USA*, *93*(21), 11956-61.

Salinas, E., & Sejnowski, T. J. (2001). Gain modulation in the central nervous system: where behavior, neurophysiology, and computation meet. *Neuroscientist*, *7*(5), 430-440.

Siegel, R. M. (1998). Representation of visual space in area 7a neurons using the center of mass equation. *Journal of Computational Neuroscience*, *5*(4), 365-81.

Spratling, M. W. (2008a). Predictive coding as a model of biased competition in visual attention. *Vision Research*, *48*(12), 1391-408.

Spratling, M. W. (2008b). Reconciling predictive coding and biased competition models of cortical function. *Frontiers in Computational Neuroscience*, *2*, 4.

Spratling, M. W. (2010). Predictive coding as a model of response properties in cortical area V1. *Journal of Neuroscience*, *30*(9), 3531-3543.

Spratling, M. W., De Meyer, K., & Kompass, R. (2009). Unsupervised learning of overlapping image components using divisive input modulation. *Computational Intelligence and Neuroscience*, 381457.

Spratling, M. W., & Johnson, M. H. (2001). Dendritic inhibition enhances neural coding properties. *Cerebral Cortex*, *11*(12), 1144-9.

Weisstein, E. (1999). Gaussian function. In *Mathworld – A Wolfram Web Resource.* Available from http://mathworld.wolfram.com/GaussianFunction.html

White, R., R. L., & Snyder, L. H. (2004). A neural network model of flexible spatial updating. *Journal of Neurophysiology*, *91*(4), 1608-19.

Xing, J., & Andersen, R. A. (2000a). Memory activity of LIP neurons for sequential eye movements simulated with neural networks. *Journal of Neurophysiology*, *84*(2), 651-65.

Xing, J., & Andersen, R. A. (2000b). Models of the posterior parietal cortex which perform multimodal integration and represent space in several coordinate frames. *Journal of Cognitive Neuroscience*, *12*(4), 601-14.

Zipser, D., & Andersen, R. A. (1988). A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, *331*(6158), 679-84.