Linear Discriminant Analysis and Noise Correlations in Neuronal Activity

Matias Calderini

Thesis submitted to the University of Ottawa
in partial Fulfillment of the requirements for the
Master of Arts in Psychology

School of Psychology
Faculty of Social Sciences
University of Ottawa

Abstract

The effects of noise correlations on neuronal stimulus discrimination have been the subject of sustained debate. Both experimental and computational work suggest beneficial and detrimental contributions of noise correlations. The aim of this study is to develop an analytically tractable model of stimulus discrimination that reveals the conditions leading to improved or impaired performance from model parameters and levels of noise correlation. We begin with a mean firing rate integrator model as an approximation of underlying spiking activity in neuronal circuits. We consider two independent units receiving constant input and time fluctuating noise whose correlation across units can be tuned independently of firing rate. We implement a perceptron-like readout with Fisher Linear Discriminant Analysis (LDA). We exploit its closed form solution to find explicit expressions for discrimination error as a function of network parameters (leak, shared inputs, and noise gain) as well as the strength of noise correlation. First, we derive equations for discrimination error as a function of noise correlation. We find that four qualitatively different sets of results exist, based on the ratios of the difference of means and variance of the distributions of neural activity. From network parameters, we find the conditions for which an increase in noise correlation can lead to monotonic decrease or monotonic increase of error, as well as conditions for which error evolves non-monotonically as a function of correlations. These results provide a potential explanation for previously reported contradictory effects of noise correlation. Second, we expand on the dependency of the quantitative behaviour of the error curve on the tuning of specific subsets of network parameters. Particularly, when the noise gain of a pair of units is increased, the error rate as a function of noise correlation increases multiplicatively. However, when the noise gain of a single unit is increased, under certain conditions, the effect of noise can be beneficial to stimulus discrimination. In sum, we present a framework of analysis that explains a series of non-trivial properties of neuronal discrimination via a simple linear classifier. We show explicitly how different configurations of parameters can lead to drastically different conclusions on the impact of noise correlations. These effects shed light on abundant experimental and computational results reporting conflicting effects of noise correlations. The derived analyses rely on few assumptions and may therefore be applicable to a broad class of neural models whose activity can be approximated by a multivariate distribution.

Acknowledgements

I would like to thank the Fonds de Recherche Nature et Technologies (FRQNT) for providing funding for my research, thus allowing me to focus on what is important instead of what is urgent. I would like to thank my supervisor Jean-Philippe Thivierge for all of his guidance and help, particularly at the moments where the important and the urgent were not mutually exclusive. A special thank to Eric, who opened the gate of this fascinating universe to a young and naive undergraduate me. Nareg and Philippe, who helped me learn how to speak and walk in a world of programming and numerical methods. Finally a special thanks to my family and friends who supported me during the stresses and the excitement of the last two years of publishing and conferences, particularly Claire who was always near with a smile and a tasty meal when the connectivity matrices and the balanced networks would start fighting back.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Neuronal noise

A particularly important hurdle in both experimental and computational work is the fact that neural communication, as with every biological process, is intrinsically noisy. The population activity of a given network rarely visits the same patterns of activity, even under the same stimulus.

The variability in the behaviour of individual cells arises most fundamentally at the biophysical level from thermal or Johnson-Nyquist noise. This source of noise relates to the voltage fluctuations across a resistance which are intrinsic to every system operating at a temperature above absolute zero [1, 2, 3]. Still at the molecular level, ionic conductance and ionic channel noise play a role in neural variability by modulating ion migration in and out of membrane channels [4, 5, 6, 7]. At the synaptic level, noise can arise due to neurotransmitter release mechanisms. Release behaviour in chemical synapses is a probabilistic phenomenon that can both happen even in the absence of incoming spikes or fail to happen in the presence of them. Further, neurotransmitter release probability is highly modulated by the spiking history of pre- and post-synaptic cells which is further influenced by time-dependent plasticity processes such as spike-time dependent plasticity (STDP).

All sources of noise considered, the most important source of variability is believed to arise from the spatial and temporal integration of afferent synapses. Even at low levels of synaptic noise, the compounding of thousands of single-cell contributions in time and space, is shown to give rise to a rich dynamic range of network activity. This can range from asynchronous, non-oscillatory patterns closely captured as Poisson processes, to oscillatory, synchronous activity on a time-scale that can largely exceed that of the synaptic transmission itself [8].

Given its intrinsically noisy nature, population activity is better described

as a probabilistic system of myriad random variables. To mathematically model and describe such an intricately coupled system the dependence or more properly speaking, the correlation in noise across units is a key feature to consider.

1.2 Correlations in neuronal noise

While some of the aforementioned noise is inherent to individual units, much of it arises from neuronal communication. If noise is shared across cells in direct or indirect connections, it is possible then that correlations in the fluctuations of activity arise. These correlations are best categorized into "signal" and "noise" correlations. Signal correlation between two units refers to the correlation in the average response of each unit as stimulus is changed, *e.g.* the correlation in preferred orientation of gratings in a visual cue task by neurons in primary visual cortex [9]. On the other hand, noise correlations refer to how similar the fluctuations around the mean response are across neurons. This is the type of correlation studied in this work, which, for brevity might be simply called correlation.

The simplest network model of a noisy neural system would assume independence between units. Historically, this has not only been a practical mathematical abstraction, but due to technological limitations, correlations were experimentally unexplored. Indeed, until recent years, the extent of our understanding of neuronal systems was limited to the electrophysiological and neurochemical study of individual or pairs of neurons. Many great insights have been derived in such manners, most notably in neuronal learning processes such as spike-timing dependent plasticity (STDP). However, the limited view of the neuronal network that such methodology provides, unavoidably misses larger scale computational properties. Considering that pairwise noise correlations can be quite weak [10], it is only when a sufficiently large sample of neurons

are being observed simultaneously that they can be reliably measured. Even though empirically measured pairwise correlations can be low, they can lead to significant differences in terms of information encoding [11] when observing dynamics at the level of the network and even in psychophysical tasks [12].

Under a paradigm where noise is assumed to be independent across units, its effect on neural encoding has been thoroughly described in previous works. This ranges from the factors controlling information transmission by population codes [13, 14, 15, 16], as well as the influence of processes such as learning and memory [17, 18, 19], to optimal computations in units acting as readouts from population codes [20].

However, the assumption of uncorrelated neural noise cannot be taken lightly. From a simple signal-processing perspective, ignoring the presence of correlation in noise can lead to biased estimates of the measured signal when attempting to "average-out" the noise. Indeed, if noise is sufficiently correlated across trials, averaging the noise would increase it proportionally to the signal. Second, as it will be shown in this work, the presence of correlations, or more precisely, the dependence of a given group of units gives rise to qualitatively different emergent behaviour in terms of information decoding. Therefore, the conclusions on the effect of neural noise have to be revisited under the assumption of correlation across units.

1.3 Noise correlations and decoding

One first approach to study the effects of noise correlations is to analyse information encoding. This is done by quantifying the amount of information encoded within a network with correlated noise, in comparison to a network with uncorrelated noise. This is normally done by computing the information of correlated responses and comparing it to the estimate given by shuffling the

responses. Theoretical and empirical results on this approach have shown that encoded information can increase or decrease depending on the relationship between noise and signal correlations [21], network size, [11, 22], the relationship between correlations and tuning curves through Fisher Information [23], etc. The clearest conclusions one can make on the subject are that encoding information cannot be derived exclusively from the individual responses of each unit and that information and correlation are not linearly, nor monotonically dependent.

While much insight can be gained from an approach based on information encoding, ultimately, what most matters is how the brain performs computations on the transmitted information (correlated or not). The second approach to the study of noise correlations, and the one taken in this work is therefore to look at information decoding. The goal in this case is to determine the amount of information that is gained, if any, when applying a decoding algorithm that considers the correlations in noise, against one that doesn't. This approach has the advantage of remaining (mostly) computation and implementation independent and it helps setting upper bounds on computations. In this case, instead of comparing shuffled against un-shuffled data, one is interested in the information loss when decoding correlated data with a decoder trained on shuffled data.

In this work, instead of taking an explicitly information-theoretic approach based on the absence or presence of noise correlations, we analyse the accuracy rate of a read-out unit tasked with classifying different stimuli. The activity to be decoded is derived from a rate-model allowing to provide a full picture not only of the effect of neural correlations on decoding accuracy, but also its interplay with relevant network parameters such as network noise gain and input magnitude.

1.4 Linear decoder

As our decoding mechanism, we choose a binary linear classifier, which is defined as a function that takes an input $x \in \mathbb{R}$ and attributes it to one of two classes y from the rule:

$$y = \begin{cases} 1 & \mathbf{w} \cdot \mathbf{x} > c \\ 0 & else \end{cases} \tag{1}$$

where $w \in \mathbb{R}^n$ is the set of weights that best classifies the data and c a threshold value. In the case of neuronal responses, the input can be the read-out input vectors expressed as instantaneous frequencies and the classes y represent different stimuli that generate the response y.

A linear classifier is chosen for multiple reasons. First, because it is a biologically plausible readout mechanism. Indeed linear classification can be performed via a perceptron rule, *i.e.* as a weighted sum of the pre-synaptic inputs passing through a threshold function. Further, while linear separation might be inadequate in low-dimensionality spaces, where non-linearities are likely to arise, it can be shown that as dimensionality increases it becomes more likely that the distributions become linearly separable in state-space. Since read-out units in cortex tend to have thousands of synaptic inputs, cortical state distributions have a high likelihood of being separable. This is also true of trajectories of cortical activity [24]. Also, while a perceptron learning rule can arguably be unrealistic in a biological context (i.e. as a supervised, delta learning rule), previous studies have shown that it can result from Hebbian learning through synaptic spike-time-dependent plasticity (STDP) and spike-frequency adaptation (SFA) rules in realistic spiking networks [25]. While here a rate-model was specifically used because one can abstract from spiking and integration mechanisms to streamline the theoretical studies, one could expand the model to a spiking network without loss of generalizability. It would suffice to establish

a biologically relevant and plausible measure of spike integration such as an instantaneous spiking frequency in sufficiently short sliding time-bins.

1.5 Linear Discriminant Analysis

From the possible plethora of linear models, we choose particularly to use Fisher Linear Discriminant Analysis (LDA). The most important reason for LDA to be chosen is that its weights can be calculated from a closed form solution. This implies two advantages. First, for low dimensionality problems, such as the one studied here, it is not necessary to algorithmically train the classification model, which can be, among some disadvantages, a lengthy process. Most importantly, it allows for mathematical tractability. As opposed to other, generally better performing categorization algorithms such as Support Vector Machines, or even simpler ones such as Logistic Regression, LDA weights can be calculated directly from the statistical parameters (*i.e.* means and covariances) of the distributions. If one succeeds in relating these parameters to the network's parameters (*i.e.* noise gains, time constants), then one can re-frame the classification problem explicitly from relevant biological quantities. In this sense, LDA allows to set an upper bound on classification performance, while allowing profound mathematical insight on interactions within parameter-space.

LDA can be approached in two ways. First, we can frame it under Bayesian theory. In that case, we are interested in determining the probability that a given point X belongs to class $y = k$, where the priors are estimated from data of known class:

$$P(y = k|X) = \frac{P(X|y = k)\,P(y = k)}{P(X)} \tag{2}$$

$$= \frac{P(X|y = k)\,P(y = k)}{\sum_l P(X|y = l)\,P(y = l)} \tag{3}$$

In the case of LDA and quadratic discriminant analysis (QDA), the data is assumed to be sampled from a multivariate Gaussian distribution with class covariance matrix Σ_k and class mean vector μ_k, giving:

$$P\left(X|y=k\right)=\frac{1}{\left(2\pi\right)^{N/2}\left|\Sigma_k\right|}exp\left(\frac{-1}{2}\left(X-\mu_k\right)^T\Sigma_k^{-1}\left(X-\mu_k\right)\right) \tag{4}$$

With N being the dimensionality of the feature space. For LDA, we assume the class covariance matrix to be equal across classes, *i.e.* the variance in input is independent of its mean, therefore $\Sigma_1=\Sigma_2=\Sigma$

However, here we take an approach closer to that of the perceptron rule described above and as stated by Fisher in his original article [26]. Fisher LDA attempts to find a projection line w (perpendicular to the discriminant hyperplane or decision boundary) onto which the input space is projected. The optimal projection line is that which maximizes the Fisher criterion $J\left(w\right)$ defined as the ratio of the projected between to within class variances:

$$J\left(w\right) = \frac{\left(\tilde{\mu_2}-\tilde{\mu_1}\right)^2}{\tilde{\sigma_1}^2+\tilde{\sigma_2}^2} \tag{5}$$

$$= \frac{w\cdot\left(\mu_2-\mu_1\right)^2}{w^T\cdot\Sigma_W\cdot w} \tag{6}$$

With $\Sigma_W=2\Sigma$ since LDA assumes equal class covariance across classes. By taking the derivative of $J\left(w\right)$ w.r.t w and setting it to 0, one can find the closed-form solution for the optimal projection line (weights) to be:

$$W=\left(2\Sigma\right)^{-1}\left(\mu_2-\mu_1\right) \tag{7}$$

In sum, LDA considers two multivariate Gaussian distributions with shared covariance matrix. It finds the optimal linear projection, *i.e.* the line of projection that allows for the greatest distance between distributions. It uses this line

to reduce the dimensionality of the multivariate distributions and classify them given a chosen threshold. The limitations and flexibility on the assumptions are discussed in later chapters.

2 From model to multivariate Gaussian

2.1 Basic definitions

Our network consists of a pair of disconnected sub-networks receiving a common input ν_i and feeding forward to a single read-out unit tasked with determining the identity i of the input. For this work, the set of possible stimuli is limited to two distinct stimuli, ν_0 and ν_1. The extension of the model to multi-class categorization is discussed further in section 6.3.2 and should not differ significantly from theory derived by the binary classification model. To limit the number of arbitrarily chosen assumptions about topology, cell dynamics and other physiological considerations, we were interested in a model-agnostic measure of network activity that would result in the feature space of the classifier. In other words, as the property to be read-out by the classifier, we chose a measure that could be naturally generalized to different types of network models, be it fully biophysical or closer to the mathematical abstraction. Specifically, the activity of each sub-network was condensed to its mean firing-rate by fully describing them with a linear neural integrator model as shown in equation 8. Such models have been shown to represent an approximation of underlying spiking activity in neuronal circuits and capture the quasi-linear responses of firing rates found across many neuronal types [27, 21].

The resulting network schematic can be seen in figure 1. A constant or slow-varying, common input ν_i originating from upstream networks is fed into the two sub-networks, which are disconnected from each other to fully isolate the effect of noise correlations and signal correlations. The input is processed by each sub-network, resulting in a mean firing rate which is dependent exclusively on its own network's parameters and subject to a time fluctuating noise-source independent of input or network dynamics. The mean firing rates are then being read at time intervals on a time scale proportional to that of the integrator

model by the downstream read-out unit which attempts to linearly separate the resulting two-dimensional firing rate distributions by the input that generated them.

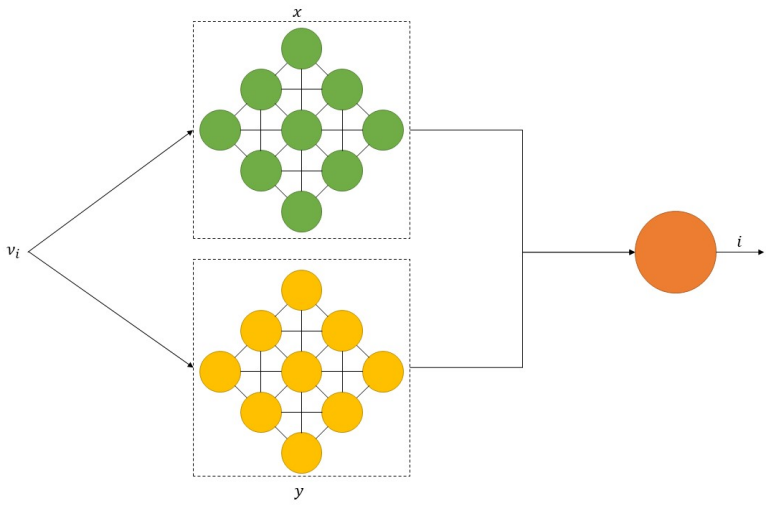

Figure 1: Schematic of network organization. Green and yellow circles represent single neurons within a sub-network or unit, represented as dashed black boxes. The orange circle represents the readout unit, which must categorize the input category i from the network activity of units x and y under stimulus ν_i

The dynamics of the units' firing rates follow the following ODE system:

$$\begin{aligned} \tau_x \frac{dx}{dt} &= -\alpha_x x + \nu_i + \beta_x \xi_x(t) \\ \tau_y \frac{dy}{dt} &= -\alpha_y y + \nu_i + \beta_y \xi_y(t) \end{aligned} \tag{8}$$

where x and y are respectively the firing rates of the first and second unit, τ is a rate time constant, α a leak term, $\xi(t)$ a Gaussian white noise ($\mathcal{N}(0,1)$) term and β the gain of the noise. The network parameters τ, α and β are bound to the positive real line $\mathbb{R}_{>0}$. While the inputs ν_i can take any real value ($\nu_i \in \mathbb{R}$), we are interested in classifying across distinct categories i, which, for simplicity, are notated throughout this text as non-negative natural numbers ($i \in \mathbb{N}_0$).

An important property of the system is that the equations for the firing rate of x and y are actually uncoupled, thus solvable separately. This holds true, even when considering the processes $\xi(t)$ to be correlated, since they are independent of either unit's rates. For ease of manipulation, the correlation in the noise terms $\xi(t)$ is later introduced in the rate distributions' covariance matrix through an easily tunable parameter ρ.

A sample of activity from an individual unit is shown in figure 2. The top left panel shows the long-term behaviour of the units' rate. Irrespective of initial value, the units' rates will tend towards a time-independent mean. The mean of this stationary regime depends on each unit's dynamic parameters. The bottom left panel gives a closer look at the stationary regime, and the right panel shows its distribution of rates. The stationary regime has a unimodal distribution, seemingly symmetric resembling a Gaussian distribution. This is explored shortly.

2.2 Statistical properties

As previously discussed, the read-out unit of the joint rate distributions is modelled as a perceptron whose weights are derived from a Linear Discriminant Analysis framework. This means that through training, the read-out unit can build a representation of the class distributions from their first two statistical moments, *i.e.* their multivariate means $\boldsymbol{\mu_i}$ and class covariance matrices $\boldsymbol{\Sigma}_i$. In order to develop an analytical estimate of the read-out's behaviour we find these first, by solving 8. Since the rate equations for x and y are uncoupled, the ODEs can be solved as two separate stochastic linear differential equations. The full derivation of such approach is found in section A.1. Simpler however is to first rearrange equation 8 (x is arbitrarily used as an example) as:

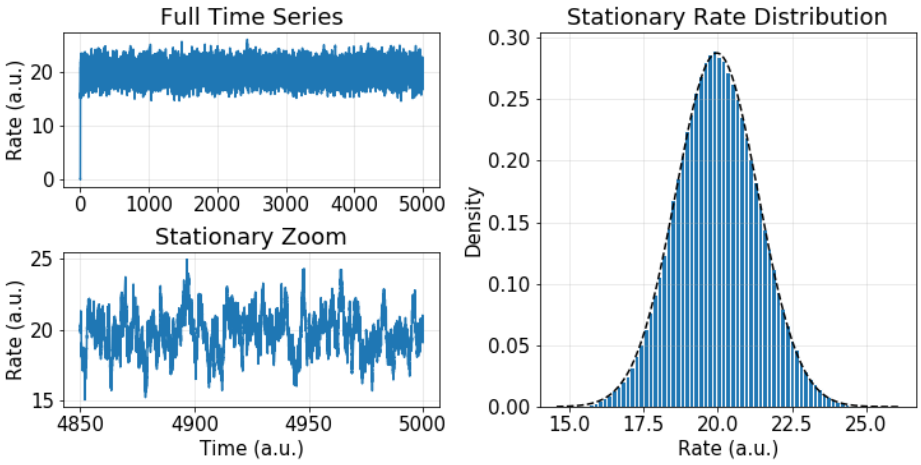

Figure 2: Temporal dynamics of the integrator model. Top-left panel shows the long-term behaviour of the integrator model with constant input. Bottom-left panel zooms on the last instantiations of the stationary regime. Right panel shows the overall distribution of instantaneous firing rates. The blue histogram represents firing rates from numerical simulations. Black dotted line shows the probability density function for a normal distribution with the same mean and variance as the stationary regime of the numerical simulations.

$$dx = \frac{\alpha_x}{\tau_x}\left(\frac{\nu_x}{\alpha_x} - x\right)dt + \frac{\beta_x}{\tau_x}\xi_x(t)dt \tag{9}$$

Additionally, a white noise process is by definition the time derivative of a Weiner process: $\xi(t) = \frac{dW_t}{dt}$. Thence, we can use this identity to rewrite the last expression as:

$$dx = \theta_x(\mu_{ix} - x)dt + \lambda_x dW_{x,t} \tag{10}$$

with $\theta_x = \frac{\alpha_x}{\tau_x}$, $\mu_{ix} = \frac{\nu_x}{\alpha_x}$ and $\lambda_x = \frac{\beta_x}{\tau_x}$. This rearrangement is useful in that equation 10 can now be more easily recognized as an Orstein-Uhlenbeck (O-U) process. This is advantageous in that historically, O-U processes have been exhaustively studied with many applications identified in pure mathematics, finance and most importantly for this work, neuroscience. They can be considered as the continuous time analogue of AR(1) processes. O-U processes have

a known solution of the form:

$$x(t) = \mu_{ix} + (x_0 - \mu_{ix})e^{-\theta_x t} + \lambda \int_0^t e^{-\theta_x (t-s)} dB(s) \tag{11}$$

which corresponds to the solution in A.1 once the change of variables is applied. Among the dynamic properties of O-U processes, two are of particular value for the underlying rate model application and the read-out mechanisms chosen here.

First, O-U processes are mean reverting stochastic processes. This means that under a constant stimulus, their mean activity will tend towards an equilibrium, long-term mean. This corresponds exactly to our previous observation about the integrator's model behaviour in figure 2. The mean reverting property is crucial for a decoding model. Indeed, the read-out unit requires a sufficiently large sample of activity to accurately estimate its mean and variance around the mean. If these parameters were to change rapidly (relatively to the time-scale of significant acquisition) over time, the read-out's estimation would lag behind the input's true distribution and risk misclassifying inputs. Moreover, the mean reverting property makes a classifying network such as the one proposed more resilient to noise. This concept is exemplified in figure 3. The red line in the top red panel represent the input into an O-U unit. If sudden changes in its magnitude occur, the unit's activity, (represented in blue on the bottom panel) will react accordingly and deviate from its long-term mean proportionately to the magnitude of the input perturbation. As soon as the perturbation disappears and the input goes back to baseline, the O-U unit will decay back to its stable activity (dashed line on bottom panel). If instead, the change in input was low, the O-U unit adapts to its instantaneous equilibrium activity. This is particularly useful in our model, since each unit represents the global average of a sub-network. This means that to be a fair representation of underlying activity,

the unit must be able to accommodate for noise, whether it comes from outside or from within the sub-netwrok, without compromising too much accuracy in its representation of the true unit's distribution.

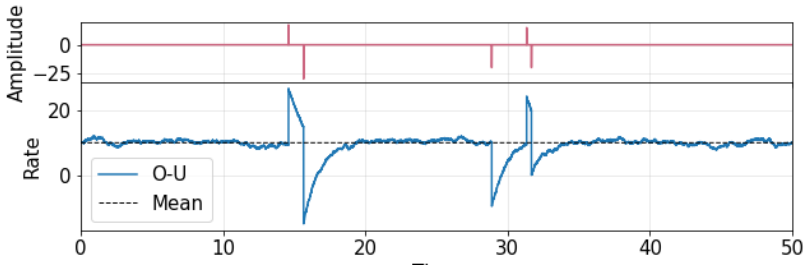

Figure 3: Mean-reverting property of integrator model. Top panel, impulse input into integrator model. Bottom panel, response in the firing rate of integrator model to impulse inputs and decay to baseline firing rate (black dashed line)

Second, O-U processes can be shown to be Gaussian distributed at their stable regime. Unlike Weiner processes, which can lead to non-finite variance as a function of time, O-U processes have bound mean and variance. This corresponds to the second observation of figure 2. Most importantly, this mean and variance can be calculated directly from model parameters. The full derivation can be found in section A.2, here, we present the main results:

$$E[x(t)] = \mu_{ix} + (x_0 - \mu_{ix})e^{-\theta_x t} \tag{12}$$

$$Var(x(t)) = \frac{\lambda_x^2}{2\theta_x}\left(1 - e^{-2\theta_x t}\right) \tag{13}$$

As discussed before, we are interested in the stationary behaviour of the O-U process, since the time-scale of convergence to the stationary mean and variance could lead to an inaccurate representation of the true rate space created by the

different units. The relevant statistical measures therefore become:

$$\lim_{t\to\infty} E[x] = \mu_{ix} = \frac{\nu_i}{\alpha_x} \tag{14}$$

$$\lim_{t\to\infty} Var(x) = \frac{\lambda_x^2}{2\theta_x} = \frac{\beta_x^2}{2\tau_x\alpha_x} = \sigma^2 \tag{15}$$

We previously defined the domain of α and τ to be $\mathbb{R}_{>0}$. In terms of the variance (eq. 15), first, this ensures that the parameters are mathematically valid, in that $\sigma^2 > 0 \; \forall \; \alpha, \; \tau$. While strictly this would be true as long as $sign(\alpha) = sign(\tau)$, this more stringent condition also insures that the biological meaning of the parameters (*e.g.* the time constant being a non-negative measure of time) is conserved. This is less straightforward for the inputs ν_i. Indeed, since $\alpha > 0$, from 14, we see that a negative value of ν_i would result in a negative long-term mean of the unit's rate. Even if ν_i were positive, if it was close enough to 0 relative to its variance, then the stochastic nature of the unit's rate might result in negative rate samples to be generated. Biologically, a negative rate should not be accepted, and the model should be modified to correct for such behaviour, either by rescaling, by modifying the probability distribution of rate's to be Gaussian only for positive numbers and 0 otherwise or by defining it as a half-normal distribution. Here, in this aspect we will however take a more generalized approach and allow negative values for the inputs ν_i. We will later see that for classification, the actual measure of importance is not the inputs ν_i themselves, but the difference of input across classes, which even if ν_i was also restricted to $\mathbb{R}_{>0}$ could easily be negative. Biologically, this is justifiable in that the value of x could actually reside in a linearly shifted space of rates, where the real, biological rate x' is given by $x' = x - min(x)$. In that sense, even if negative values of x are allowed by a negative (or small) ν_i, all derived results remain valid and generalizable.

Figure 4 compares the numerical estimates of mean and variance of the integrator model in equation 8 to their analytical estimates for a Ornstein-Uhlenbeck process as in 10 and 11 for all four model parameters (τ, α, ν, β). For all measures, the analytical expressions in 14 and 15 provide a good estimate of the numerical mean and variance. Aside from the qualitatively different contributions of each variable to the staitstical measures, it is worth noticing that the variance in firing rate is independent on input magnitude. This is seen in the absence of ν_i in equation 15 as well as the constant behaviour of the variance as a function of ν_i in figure 4, second row, third column. Irrespectively, the variance in activity of each unit may still be unique, since all other relevant parameters are allowed to differ across units.

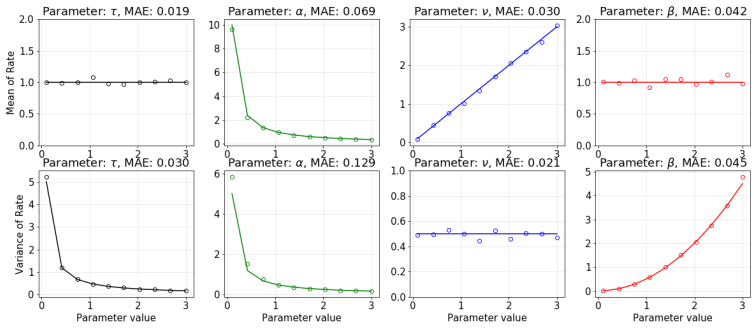

Figure 4: Comparison of numerical and theoretical values of the integrators model's mean and variance as a function of network parameters. Top row contains the estimates of mean firing rate of integrator model. Bottom row contains the estimates of the variance in firing rate of the integrator model. Solid lines are expected theoretical values, circles represent numerical estimates. Modified parameter and resulting mean-absolute-error (MAE) are written above each plot

2.3 Multivariate definitions

From these latest definitions, it is possible to better define the linear readout model operationally. For a given, shared input v_i, one can approximate the

activity of each sub-network by their respective mean firing rates x and y which have been shown to behave as two independent random variables that converge to a stable, Gaussian distributed equilibrium rate. From the perspective of the read-out unit, each stimulus response consists of a joint, bivariate normal distribution Z_i centred around a stimulus-dependent mean position $\boldsymbol{\mu_i}$ and stimulus-independent covariance matrix Σ.

Up to this point, the correlation between the random processes $\xi(t)$ (or dW_t) has not be addressed. One could add explicit dependence between the noisy terms ξ by making them linear combinations of shared common noise sources. The advantages and disadvantages of such an approach are discussed in more depth in section 6.3.1. Here, instead, we take a different approach. Irrespective of how the correlation of the random processes arises, its effect on the multivariate distributions examined would be the same: the distributions would stretch along a diagonal axis, whose slope is proportional to the correlation between its horizontal and vertical components (equal if the variables are standardized). Therefore, without loss of generalizability, one could simplify the inclusion of correlations in noise terms by setting the non-diagonal elements of Σ to $\rho\sigma_x\sigma_y$ where ρ is the correlation between the noise process of unit x and unit y and the σ's refer to the units' rate variance as defined in 15. This also allows to more simply increase correlation without increasing the noise gain as opposed to the proposed approach using linear combinations of common noise sources.

Together, all the previous definitions result in the following multivariate rate

distributions, or feature-space of the linear read-out unit:

$$Z_i = \sim \mathcal{N}(\boldsymbol{\mu_i}, \boldsymbol{\Sigma}) \tag{16}$$

$$\boldsymbol{\mu_i} = [\mu_{xi}, \mu_{yi}]^T \tag{17}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix} \tag{18}$$

As mentioned previously and seen in equation 18, the class covariance matrices are independent of input ν_i. The property of homoscedasticity (or equal covariance across classes) in the multivariate distributions is crucial to the model. Equality across class covariance is a basic assumption of LDA. If Σ were to be dependent on the magnitude of the stimuli, this would create angles between the correlated class distributions. In that case, a linear classifier would be inadequate to discriminate between classes. This situation is further discussed in section 6.3.3.

A sample response distribution is shown in figure 5. The top and right-most panels show the individual response distribution from unit x and y respectively to two different inputs (each input represented by a different color). These can be seen as individual dimensions of the feature space from which the read-out unit has to perform its classification. In this space, the central panel of the figure, each input corresponds to a unique multivariate Gaussian distribution. Based on the statistics of these multivariate distributions, the read-out unit defines an optimal linear boundary between the classes, shown as the dashed black line. The analytical estimate of this optimal discrimination is studied in the next section.

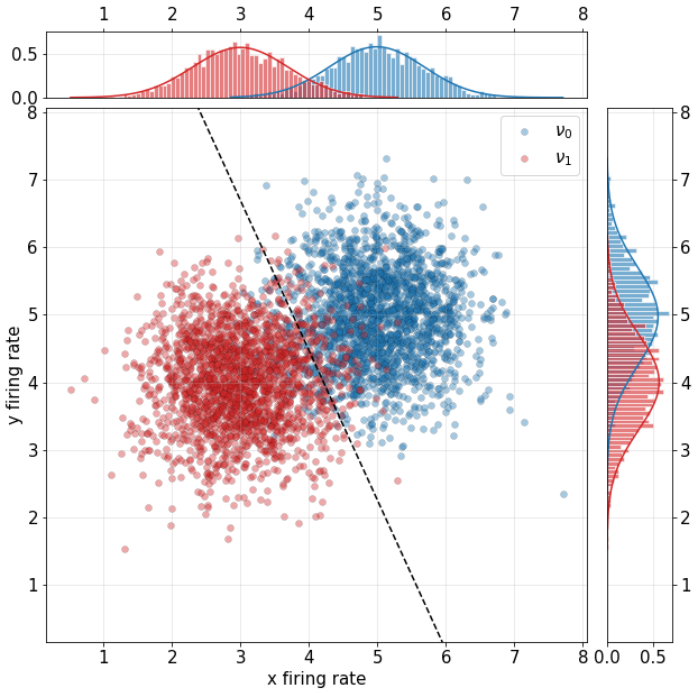

Figure 5: Top panel, firing rate distributions of unit x under two different inputs (ν_1 in red and ν_2 in blue). Right panel, firing rate distributions of unit y under the two different inputs. Central panel, joint multivariate distribution of firing rates for each input.

3 From multivariate Gaussian to error rate

3.1 Optimal linear projections

In the previous section we explored the behaviour of the units' rates. This was necessary since the rates x and y are the meaningful inputs from the read-out unit's perspective. We have shown that these are both normally distributed and have stable and finite mean and variance. Together, for each input, one can create a two-dimensional feature space as the joint distributions Z_i with known

mean vector $\boldsymbol{\mu_i}$ and stimulus-independent covariance matrix $\boldsymbol{\Sigma}$. In that sense, all basic assumptions are met and the stage is set to apply LDA as the readout mechanism.

As a perceptron, the readout unit is tuned so that the rule defined in 1 optimally discriminates samples from each distribution. In this case, optimal tuning is achieved by finding the weights W resulting in the best linear boundary between classes. For most classifiers, the tuning or training of the model can only be done iteratively until it converges to a solution resulting in a satisfactory accuracy. Within the LDA framework however, the weights have a closed form solution given by equation 7. This means that from our previous definitions of the rate model and the joint distributions, we can rewrite W explicitly from statistical and network parameters. First, we consider the total within class scatter S_w. As previously discussed, from 18 we see that class covariance matrix is independent of input, therefore:

$$\begin{aligned} S_w &= 2\Sigma \\ &= 2\begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix} \\ &= 2\begin{bmatrix} \frac{\beta_x^2}{2\tau_x\alpha_x} & \rho\sqrt{\frac{\beta_x^2}{2\tau_x\alpha_x}\frac{\beta_y^2}{2\tau_y\alpha_y}} \\ \rho\sqrt{\frac{\beta_x^2}{2\tau_x\alpha_x}\frac{\beta_y^2}{2\tau_y\alpha_y}} & \frac{\beta_y^2}{2\tau_y\alpha_y} \end{bmatrix} \end{aligned} \tag{19}$$

From 7 it is straightforward to calculate the weights from statistical and network parameters. To alleviate the notation, we first define $\Delta\boldsymbol{\mu}^T = [\Delta\mu_x, \Delta\mu_y]^T = \boldsymbol{\mu_0} - \boldsymbol{\mu_1}$. With this notation:

$$
\begin{aligned}
W &= (2\Sigma)^{-1}\Delta\boldsymbol{\mu} \\
&= \frac{1}{2(1-\rho^2)} \begin{bmatrix} \frac{1}{\sigma_x^2}\Delta\mu_x - \rho\frac{1}{\sqrt{\sigma_x^2\sigma_y^2}}\Delta\mu_y \\ \frac{1}{\sigma_y^2}\Delta\mu_y - \rho\frac{1}{\sqrt{\sigma_x^2\sigma_y^2}}\Delta\mu_x \end{bmatrix} \\
&= \frac{1}{2(1-\rho^2)} \begin{bmatrix} \frac{\tau_x}{\beta_x^2}\Delta\nu - \rho\sqrt{\frac{\tau_x\alpha_x}{\beta_x^2}\frac{\tau_y}{\beta_y^2\alpha_y}}\Delta\nu \\ \frac{\tau_y}{\beta_y^2}\Delta\nu - \rho\sqrt{\frac{\tau_y\alpha_y}{\beta_y^2}\frac{\tau_x}{\beta_x^2\alpha_x}}\Delta\nu \end{bmatrix}
\end{aligned}
\tag{20}
$$

While technically the bias term b from equation 1 should be calculated as well, we see shortly that it is unnecessary given that a practical shift of means will cancel the effect of the bias. An example of a linear boundary derived from 20 is shown in figure 5 as a black dashed line separating the distributions by stimulus category.

Previously, we treated W as weights for a perceptron's weighted sum and as resulting in a linear boundary. Strictly speaking though, W is a projection line perpendicular to the classification boundary. In that sense, the dot product $W \cdot z$ (where z is an arbitrary point in the two-dimensional feature-space) results in the scalar projection of point z onto W. In the general case of LDA applied to N-dimensional spaces, the projection is done onto an $N-1$ dimensional hyperplane. This property, as well as a comparison with alternative dimensionality reduction methods, is further discussed in section 6.3.4. In the two-dimensional feature space, the projection to W results on all data points from each distribution (all instances of instantaneous rates for both stimuli) to be distributed onto a line following a Gaussian density function (as seen on figure 6).

3.2 Error rate

From the projected distributions, classification is straightforward. Mathematically, from the law of total probability, the probability of misclassification, or

error rate ε is given by:

$$\varepsilon = P[y = 0|k = 1]P[k = 1] + P[y = 1|k = 0]P[k = 0] \tag{21}$$

where $P[k = j]$ is the probability of observing class j, *i.e.* the probability that a randomly sampled point from any distribution belongs to class j and $P[y = i|k = j]$ the probability that a point is classified as coming from class i when it actually belongs to class j. For the purposes of a general solution, we assumed so far that communication between both units and the readout unit was equally likely. That is, from our definitions, there is no reason to believe that information transmission into the readout unit is more favourable for the rates of one unit with respect to the other. In this case, $P[k = 0] = P[k = 1] = 0.5$. To model unequal probabilities of successful transmission it would be necessary to chose the probabilities $P[k = j]$ such that $P[k = 0] \neq P[k = 1]$. To calculate the conditional probabilities $P[y = i|k = j]$, first, one must define a threshold position c on the projection line that serves as a boundary between the two distributions. Without any assumptions on the distributions, the value c is chosen to be the mid-point between the two projected distributions' means. In a different context, this value could be scaled or shifted depending for example on the relative importance of Type-I and Type-II errors in classification. Since there is no *a priori* reason to assume any significant difference between projected distributions, in this work, the mid-point definition of c suffices. In figure 6, the threshold c can be seen as a black vertical dashed line. With a threshold value c, one can then calculate the misclassification probability $P[y = i|k = j]$ as the probability of a point from one class to be found on the "wrong" side of the threshold, i.e. the side opposite to its class' mean. Intuitively, in figure 6, this is equivalent to finding the probability that a blue point is found on the right or "red" side of the vertical boundary or a red point on its left or "blue" side.

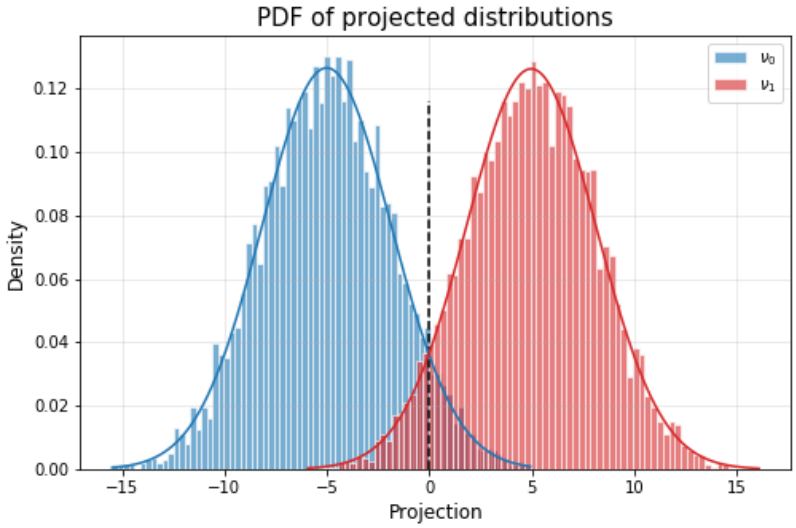

Figure 6: Optimal linear projections of each multivariate response distributions. Solid lines represent the predicted probability density functions of firing rates. Histogram shows numerical estimates of density from the projection of original multivariate distributions onto the optimal projection line given by W

If one were to reproduce that classification on every point in the original space, one could create a classification surface such as that in figure 7. Here, the scatter points represent rate pairs from the integrator model used to train the linear classifier, from which the black linear boundary is defined. The rest of the classification space is covered in a fine mesh for which each point is classified using the procedure described above. The color of the mesh points reflect the probability of being classified as the blue or red class. At a high correlation level as that presented ($\rho = 0.9$), the distinction between the classes is clear except for the region closest to the boundary, which tends to white, meaning that points in that region could be attributed to either class.

One could explicitly calculate $P[y = i|k = j]$ as the area under the curve of the density function of j in the region where i is considered the right class. However, given that the density functions of the distributions are Gaussian,

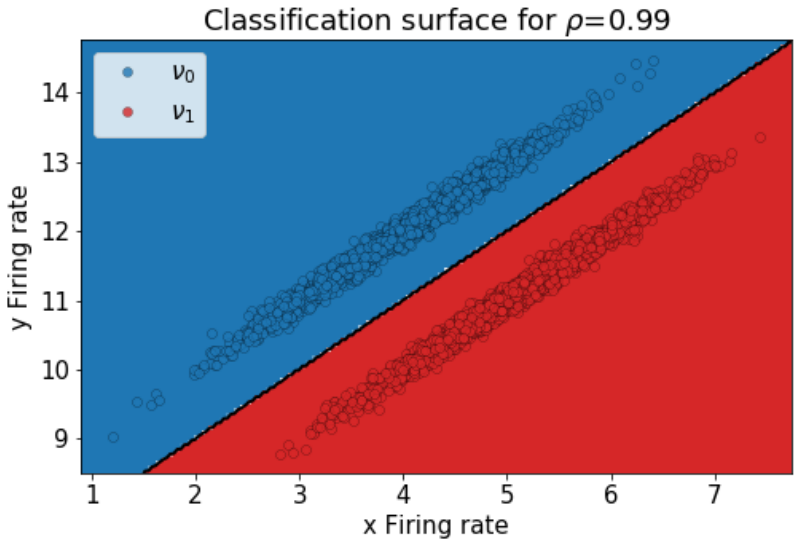

Figure 7: Optimal classification surface from multivariate response sample. Solid black line represents the optimal linear boundary between classes. Red and blue background represents the probability of a point on that point in feature space to be of the red or blue class respectively.

and Gaussian distributions are more simply integrated when they are bound between $\pm\infty$ and 0, as opposed to an arbitrary value c, we will first perform a useful shift on the projected distributions. Instead of using the raw projected distributions, we will shift them by c, so the threshold is poised at a value of 0. The un-shifted threshold c, the means of the shifted distributions η_i and their variance ς^2 can then be expressed as:

$$c = W \cdot \frac{1}{2}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0}) + b \tag{22}$$

$$\eta_i = W \cdot \boldsymbol{\mu_i} + b - c \tag{23}$$

$$\varsigma^2 = W^T \Sigma W \tag{24}$$

As mentioned earlier, it is at this point that the bias term b will cancel itself in what concerns the solution for error rate. Indeed, although b appears in 23, it

also appears in c. Consequently, when the shift occurs (the $-c$ term in eq. 23), the bias term disappears. The shifted distributions with a threshold of 0 at the midpoint between their means are represented in figure 6.

With well defined expressions for mean and variances of the Gaussian density functions, it is possible to calculate the misclassification probabilities $P[y = i|k = j]$. From the shifted notation, the error rate from equation 21 written as a function of integrals of the Gaussian density functions become:

$$\varepsilon = \frac{1}{2}\int_{-\infty}^{0} \frac{1}{\sqrt{2\varsigma^2\pi}} e^{\frac{-(w-\eta_1)^2}{2\varsigma^2}} dw + \frac{1}{2}\int_{0}^{\infty} \frac{1}{\sqrt{2\varsigma^2\pi}} e^{\frac{-(w-\eta_0)^2}{2\varsigma^2}} dw$$

The full integration of the error can be found in section A.3, the final expression is:

$$\varepsilon = \frac{1}{2} erfc\left(\frac{\eta_1}{\sqrt{2\varsigma^2}}\right)$$

As concise as this last expression is, closer inspection of η_i and ς^2 from 23 and 24 reveal that this can be further simplified. For clarity, we first introduce the squared Mahalanobis distance d^2:

$$d^2 = \Delta\boldsymbol{\mu}^{\boldsymbol{T}}\Sigma^{-1}\Delta\boldsymbol{\mu} \tag{25}$$

The Mahalanobis, as opposed to the more widely used Euclidian distance is used to measure the distance between a point and a distribution instead of the distance between two points. It is a standardized distance in that its reference point is the mean of the distribution and every dimension of the space on which it is used is scaled by its variance. For multivariate Gaussian distributions, such as the ones studied here, the Mahalanobis distance can be seen as the higher-dimensional extension of the z-score. Generally, contrary to Euclidian distance, the Mahalanobis distance between one distribution and its counterpart's mean is not assured to be equal to the distance between the second distribution and

the first distribution's mean *i.e.* $d(\boldsymbol{\mu_0}, \boldsymbol{\mu_1}) \neq d(\boldsymbol{\mu_1}, \boldsymbol{\mu_0})$. This is true if the covariance matrices of the distributions were not equal, because the resulting rescaling of space for each distribution would be different. Here, however, the covariance of both distributions is equal so we can call the expression in 25 the squared Mahalanobis distance between means without loss of generality ($\Sigma_0 = \Sigma_1 = \Sigma \Rightarrow d(\boldsymbol{\mu_0}, \boldsymbol{\mu_1}) = d(\boldsymbol{\mu_1}, \boldsymbol{\mu_0}) = d$).

As will be discussed further in section 6.2, in related works this measure is used by itself as a metric of dissimilarity between distributions as opposed to looking into categorization error rate. One can show that this coefficient of discriminability can be related to the Fisher Information of a population code [28]. Here, this variable arises naturally from the analytical derivation of error rate.

From the definition in 25, we can rewrite d^2 from statistical and network parameters:

$$\begin{aligned} d^2 &= \Delta\boldsymbol{\mu}^{\boldsymbol{T}}\Sigma^{-1}\Delta\boldsymbol{\mu} \\ &= \frac{1}{1-\rho^2}\left[\frac{1}{\sigma_x^2}\Delta\mu_x^2 + \frac{1}{\sigma_y^2}\Delta\mu_y^2 - 2\rho\frac{1}{\sqrt{\sigma_x^2\sigma_y^2}}\Delta\mu_x\Delta\mu_y\right] \\ &= \frac{2\Delta\nu^2}{1-\rho^2}\left[\frac{\tau_x}{\beta_x^2\alpha_x} + \frac{\tau_y}{\beta_y^2\alpha_y} - 2\rho\sqrt{\frac{\tau_x}{\beta_x^2\alpha_x}\frac{\tau_y}{\beta_y^2\alpha_y}}\right] \end{aligned}$$

In these expressions, the ratio $\frac{\Delta\mu_u}{\sqrt{\sigma_u^2}}$ naturally appears in every term. This should be unsurprising, since it is the fundamental expression to be optimized with the goal of finding the optimal weights vector W from the original definition of LDA as presented in equation 6 at the beginning of this text. This ratio will become particularly important throughout the rest of the analysis so we define it clearly here as:

$$r_u = \frac{\Delta\mu_u}{\sqrt{\sigma_u^2}} = \Delta\nu\sqrt{\frac{\tau_u}{\beta_u^2\alpha_u}} \tag{26}$$

which makes the squared Mahalanobis distance be more concisely written as:

$$d^2 = \frac{1}{1-\rho^2}\left[r_x^2 + r_y^2 - 2\rho r_x r_y\right] \tag{27}$$

This theoretical aside was useful in that closer inspection of the expressions for the means and variances of the shifted distributions as well as the expression for c and W (equations 23, 24, 22 and 20 respectively) shows that the error is closely related to the Mahalanobis distance between means. The full derivations can be found in sections A.3 and A.4, where the final and most concise expression for error is:

$$\varepsilon = \frac{1}{2} erfc\left(\frac{1}{2\sqrt{2}}\sqrt{d^2}\right) \tag{28}$$

where d^2 contains all of the statistical or network parameters, including the correlation ρ. Here, we chose to write $\sqrt{d^2}$ over $|d| = d$ to hint at the potential interpretation of the term as the magnitude of the Mahalanobis distance if this model was to be extended to higher-dimensional spaces. This is further discussed in section 6.3.2. Since the complementary error function $erfc(\cdot)$ is monotonically decreasing for its whole domain, an expression of error as a function exclusively of distance also allows an intuitive read on the behaviour of the curve. Indeed, as d is increased, the argument of the error function increases, thus the error decreases. This would be an expected behaviour of the error since geometrically, it is easier to draw a classification line between far off distributions than between distributions that are very close to each other.

At this stage, our previous definitions of parameter domains (sec. 2.1) become particularly important. The stimulus ν_i was allowed to be negative against potential biological considerations among other reasons because the value of ν_i itself was mentioned to be of little interests in comparison to the difference in stimuli $\Delta\nu$. From the expressions in 26 and 27, it becomes apparent that this is

indeed the case. In the final expression for error rate (eq. 28), at no point are the ν_i implicated in the error rate, if it is not through the $\Delta\nu$ terms.

Figure 8 compares the expression in 28 to numerical estimates of the error rate as a function of noise correlation values. The latter were obtained by iterating through correlation values and generating samples of 3000 firing rate pair instances for two categories of stimuli, randomly selecting a subset (80%) of points from each distribution to train a LDA classifier and measuring the proportion of misclassified rate pairs on the remaining (20%) of the samples. The analytical (black line) and numerical (red line) estimates were in good agreement.

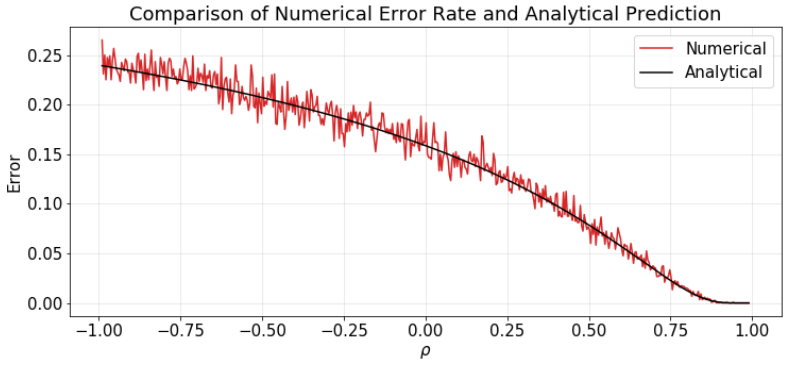

Figure 8: Comparison of numerical and analytical estimates of classification error rate as a function of noise correlation. Black line, analytical error estimate. Red line, numerical error estimate

We will further explore several properties of the error rate in the next section. Particularly, the contribution of each parameter, notably noise correlations, in the emergence of qualitatively varied and counter-intuitive results on the behaviour of the error curve.

4 Qualitative behaviour of error rate

In the previous section, we managed to successfully leverage the closed-form solution of LDA weights to develop a concise expression for classification error rate. In that respect, equation 28 consists of the main results of this work since it allows to relate parameters of neuronal network processing to an accurate estimate of error rate. This is useful because it allows a controlled, full exploration of parameter space and its influence on the qualitative behaviour of the error curve, which we will shortly show to be very rich. Without such an expression, this would not be possible because inherent fluctuations in numerical estimates would obscure fine differences in error estimates under controlled conditions.

We start exploiting our solution for error rate by first determining the minima and maxima of the error rate as a function of noise correlations. The full derivation can be found in section A.6. Here, we present the types of extrema found for different conditions and their respective position in the ρ axis:

Table 1: Conditions and Positions of error extrema

	Condition	Position of Extrema	Type
1.	If 2 does not apply	$\rho \to 1$	Min
2.	$r_x \to r_y$	$\rho \to 1$	Max
3.	$r_x = 0$ or $r_y = 0$	$\rho = 0$	Max
4.	$r_x \neq 0$ and $r_y \neq 0$	$\rho = \frac{min\left(r_x^2, r_y^2\right)}{r_x r_y}$	Max
5.	$r_x \to -r_y$	$\rho \to -1$	Max
6.	If 5 does not apply	$\rho \to -1$	Min

Together, from all of these conditions, four qualitatively different cases naturally arise. First, what will be referred to as the "general" case. We refer to it as general, since it is the natural behaviour of the error curve when no special or edge conditions are applied. This case refers to rows 1, 4 and 6 of table 1 and can be seen in the main plot of figure 9. On both extremes of correlation, the error curve is at a minimum or more precisely, the error will tend towards

0. Formally, $|\rho| \to 1 \ \Rightarrow \varepsilon \to 0$. In between the extremes of correlation, there is a maximum of error at the position ρ_* given by:

$$\rho_* = \frac{min\left(r_x^2, r_y^2\right)}{r_x r_y} \tag{29}$$

Irrespective of its position, as correlation is increased, error would monotonically increase from its null value at a correlation tending to -1 to $\varepsilon_* = \varepsilon(\rho_*)$ and then monotonically decrease back to 0 as correlation tends to 1. Globally, this means that in the general case, the error curve behaves non-monotonically with respect to correlation. In terms of the multivariate distributions of neuronal units' rates, the general case occurs when correlation stretches both distributions on parallel lines and their centroids do not align on either dimension as seen in the inset plot of figure 9. This shift also gives rise to the non-monotonicity of the general case. If one were to consider the multivariate distributions as the correlation level is increased, initially, when $\rho = 0$ the distributions expand equally in each direction of the 2-dimensional space. Intuitively, as correlation is increased, the distribution will stretch towards each other across parallel axis, thus increasingly overlapping each other and consequentially making discrimination across classes harder. After a maximal overlap (at ρ_*), further stretch of the distributions will begin to make them too thinly spread for them to overlap, until the extreme case of a correlation of 1, where both distributions appear as perfectly parallel lines and linear discrimination becomes perfect. More strictly speaking, the non-monotonic behaviour as well as the lack of symmetry around $\rho = 0$ can be better explained by the Mahalanobis distance between means. As discussed earlier, the rescaling of each dimension of space by its variance, makes it so that the greater the variance in a given direction, the shorter the Mahalanobis distance to a point in that direction is. In this case, the direction of greater variance corresponds to that artificially introduced by the correlation. The angle

(or slope) of this artificial stretch is directly proportional to the value of ρ and its range is limited to $[-45\,\text{deg}, 45\,\text{deg}]$ with ρ being in $[-1, 1]$. If one were to connect the two distribution means via a vector (from leftmost to rightmost distribution for simplicity), it would be at a given angle with the direction of greater variance. To minimize the Mahalanobis distance (thus maximizing error), it would then suffice that the two proposed lines be parallel. Therefore, when the centroids are not aligned in any axis, the non-zero slope of the line joining the two means, makes it that only a non-zero correlation could maximize the error. The case were the centroids are aligned is explored next.

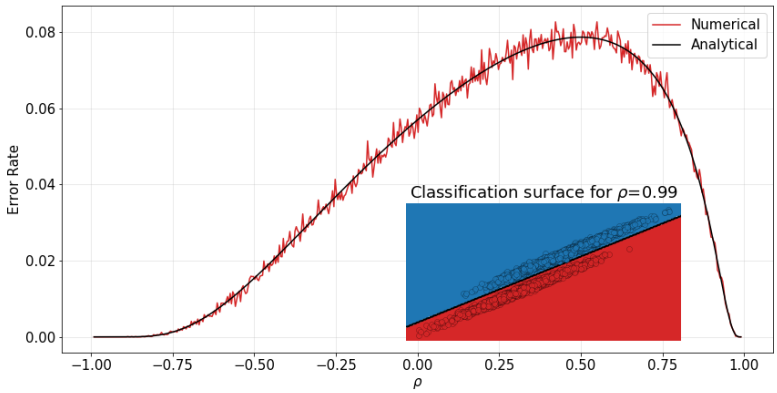

Figure 9: General behaviour of classification error curve. Error minimums found at the extreme values of correlation and a maximum peack in between them. Black line, Analytical error estimate. Red line, numerical error estimate.

The second case occurs when the maximum ε_* occurs at a value of $\rho_* = 0$. This is considered an altogether different case since it happens at the special condition where one of the ratios r_u is null. This is condition 3 in 1 and it is considered separately, because it has to be treated differently in the derivations to avoid a division by 0. Because of this condition and its qualitative shape, we refer to this case as the 0th or symmetrical case. The behaviour of the error

curve is shown in figure 10. Qualitatively, it is equivalent to the general case if the maximum error occurred at $\rho_* = 0$. The same type of non-monotonic behaviour can be observed. The condition for one of the ratios to be null can be seen in the inlet distributions plot of 10. In this case, the centroids are aligned on the horizontal axis but shifted on the vertical axis (if both ratios r_u were null, then it would not be possible to discriminate across classes, since the centroids of the distributions would overlap). As explained before, the non-monotonicity is explained by the angle between the vector connecting the means and the direction of greatest variance of the distributions: as correlation is modified, the angle between the two lines tends to zero and further tuning of the correlation after the point where the lines are parallel can only increase the angle between them. Whether the shift is at the vertical or the horizontal axis does not matter since the choice of x being the horizontal and y the vertical axis is arbitrary, thus the results are invariant to a rotation of space. Since in this case, the vector joining the means is vertical, as correlation tends to zero, the angle between the two lines of reference tends to zero as well.

The third case is an edge case, when the position $\rho_* \to -1$. This results in a monotonically decreasing error curve since the maximum is technically at the lower bound of the domain of correlations. In terms of network parameters, this happens when $r_x \to -r_y$, which corresponds to condition 5 in 1. Intuitively, the monotonic behaviour occurs because if one where to start at $\rho = 0$ and decrease its value towards $\rho \to -1$, the distributions will stretch on a common line, instead of parallel line. In that sense, the increase in the magnitude of ρ can only create more overlap between distributions. This is not true for the positive values of ρ. With respect to Mahalanobis distance, the monotonic behaviour of the error curve is due to the fact that condition 5, makes is so that the vector joining the means is at a $-45°$ angle. As previously mentioned, the

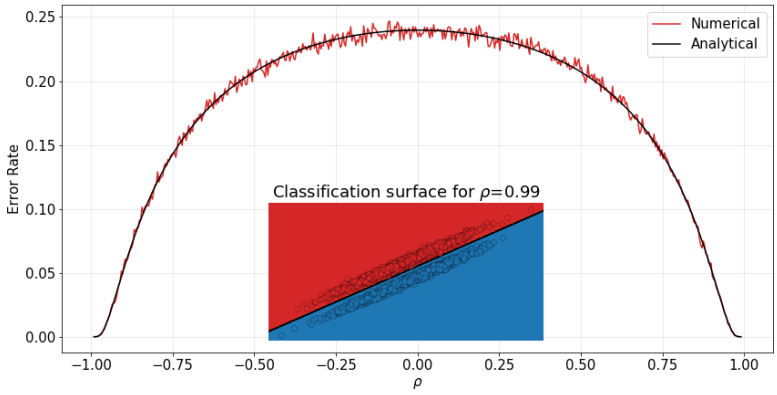

Figure 10: Error curve 0^{th} or symmetrical case. The peak error can be found at a correlation of $\rho = 0$, resulting in a perfectly symmetric error curve as a function of correlation. Black line, Analytical error estimate. Red line, numerical error estimate.

direction of largest variance is bound to the range $[-45°, 45°]$. Therefore, at its minimum value, associated with the minimum value of correlation, the vector and the direction of largest variance are parallel, thus error is maximal (d^2 is minimized). At the peak value of correlation however, the vector and direction of largest variance are perpendicular, therefore the error is at is minimum. This case and a sample of its respective distributions is shown in figure 11.

Finally, the fourth case is the case where error can only monotonically increase as a function of correlation. As before, this is equivalent to the general case when $\rho_* \to 1$ and corresponds to condition 2 in table 1. Relative to the monotonically increasing case seen before, here, the multivariate rate distributions stretch towards each other on a common line when $\rho \to -1$ from zero. The increase in overlap therefore can only increase the error rate. In terms of angles, this case is orthogonal to the previous one, in that the condition on the ratios r_u makes the vector joining the means be at $45°$, thus only an increase in correlation towards its positive bound can reduce the Mahalanobis distance be-

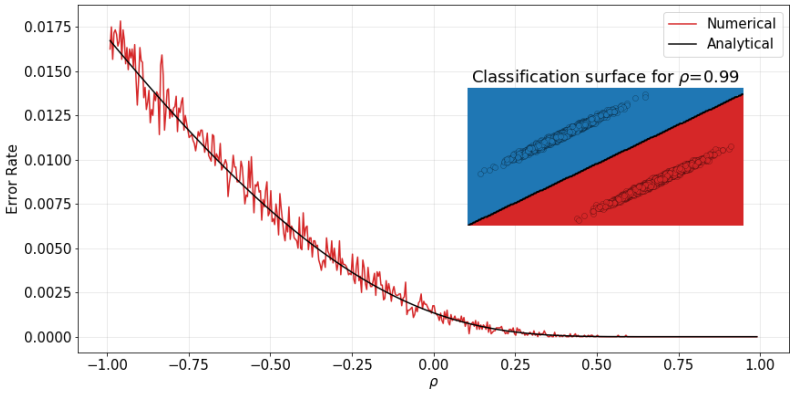

Figure 11: Error curve decreasing case. The general peak of error is found at a correlation of $\rho = -1$ resulting in a monotonically decreasing error curve. Black line, Analytical error estimate. Red line, numerical error estimate.

tween distribution means. This case and a sample of its respective distributions are shown in figure 12

All of the qualitative cases of error curve behaviour were derived from general conditions on the network parameters. On the next section, we explore more finely tuned control of the error curve from specific parameter tuning.

5 Parameter tuning and error rate

5.1 Network parameter tuning and ρ_*

The global behaviour of the error rate has been explained in four different cases and explained by implicating the Mahalanobis distance d^2. However, this approach abstracts greatly from the rich network dynamics offered by the potential tuning of network parameters (α, τ, β) and stimuli ($\Delta\nu$). In this section, we focus on how these parameters contribute to the behaviour of the error curve in the general case, specifically, at the position of maximum error

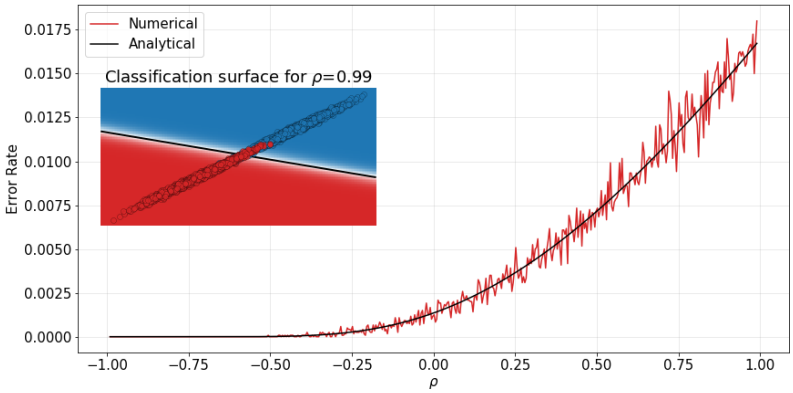

Figure 12: Error curve increasing case. The general peak of error is found at a correlation of $\rho = 1$ resulting in a monotonically increasing error curve. Black line, Analytical error estimate. Red line, numerical error estimate.

ρ_*. To begin, we first offer a different expression for ρ_* (eq. 29) that will be more intuitive to use:

$$\rho_* = \frac{min\left(r_x^2, r_y^2\right)}{r_x r_y} = \begin{cases} \frac{r_x}{r_y} & if|r_x| < |r_y| \\ \\ \frac{r_y}{r_x} & if|r_x| > |r_y| \end{cases} \tag{30}$$

Since ρ_* will be the object of this section and it depends exclusively on $\frac{r_x}{r_y}$, we present this ratio in its explicit form for both network and statistical parameters:

$$\begin{aligned} \frac{r_x}{r_y} &= \frac{\beta_y\sqrt{\tau_x\alpha_y}}{\beta_x\sqrt{\tau_y\alpha_x}} \\ &= \frac{\Delta\mu_x}{\Delta\mu_y}\sqrt{\frac{\sigma_y^2}{\sigma_x^2}} \end{aligned} \tag{31}$$

From this latest expression of ρ_*, it should be apparent than the relationship between the error curve and individual parameters does not behave as simply as its relationship with Mahalanobis distance. Without loss of generalizability,

we take as an example the case where initially $|r_x| < |r_y|$. According to 30 $\rho_* = \frac{r_x}{r_y}$. if we were to increase this ratio, for example by increasing the time constant of unit x, τ_x, ρ_* would increase until the condition $|r_x| \to |r_y|$ becomes true and $|\rho_*| \to 1$, which corresponds to either the monotonically increasing or monotonically decreasing cases of figure 11 and 12. If any further increase of τ_x would simply keep pushing ρ_* in the same direction, it would mean that the error curve would have no maximum within the bounds $\rho_* \in [-1, 1]$. However, 30 predicts something different. From our example, as soon as the increase in tau_x makes it that $|r_x| > |r_y|$, ρ_* is given by the inverse of the expression used before *i.e* $\rho_* = \frac{r_y}{r_x}$. Any further increase in τ_x would therefore have the inverse effect it had up to that point and would move back the peak of error towards a correlation of zero.

We will formalize this phenomenon, by thinking of all parameters as existing within the set G. From this set, we are able to freely manipulate a sub-set of parameters g. Within the expression for ρ_*, the sub-set of parameters g would appear naturally in the ratio $\frac{r_x}{r_y}$ (eq. 30) as a function $f(g)$ and all other parameters as a single constant c_g. For example, we can choose to manipulate the set of parameters $\{\Delta\nu, \beta_x, \alpha_y\}$. This would result in the function $f(\Delta\nu, \beta_x, \alpha_y) = \frac{\Delta\nu\sqrt{\alpha_y}}{\beta_x}$ and constant $c_{\Delta\nu, \beta_x, \alpha_y} = \frac{\beta_y\sqrt{\tau_x}}{\Delta\nu\sqrt{\tau_y\alpha_x}}$. From this definition, we can rewrite ρ_* from equation 30 as:

$$\rho_* = \begin{cases} f(g)c_g & if|r_x| < |r_y| \\ f(g)^{-1}c_g^{-1} & if|r_x| > |r_y| \end{cases} \tag{32}$$

We know from before our previous expression that the point of discontuinity would arrive at a value of $|\rho_*| = 1$. This latest expression however, also helps us map that discontinuity to our subset of parameters g through $f(g)$. Indeed,

this can only happen when:

$$f(g)c_g = f(g)^{-1}c_g^{-1}$$
$$\Leftrightarrow |f(g)| = |c_g|^{-1} \tag{33}$$

We can explore this by considering a subset of parameters more interesting than the previous two examples. We first manipulate the value of the noise gain of both units, β_x, β_y by a common factor from their original values to new values β'_x, β'_y and look at the resulting value for ρ_* and $\varepsilon_* = \varepsilon(\rho_*)$. For a section of the explored values of noise gains, we know that:

$$\rho_* = f(\beta_y, \beta_x)c_{\beta_y,\beta_x} = \frac{\beta_y}{\beta_x}\left(\frac{\sqrt{\tau_x\alpha_y}}{\sqrt{\tau_y\alpha_x}}\right)$$

Since, we impose here that both gains are modified equally and all other parameters are held constant:

$$f(\beta_y, \beta_x) = \frac{\beta_x}{\beta_y} = \frac{\beta'_x}{\beta'_y} = f(\beta'_y, \beta'_x)$$

Therefore, we can expect the value of ρ_* to be equal to that of ρ'_* for all values of $\frac{\beta_x}{\beta_y}$. This does note hold true for the maximum error ε_*, since it does not involve the noise gains exclusively as a ratio. Therefore, increasing noise gains would be expected to only multiplicatively increase the maximum error, but not the position at which it happens (ρ_*) and this is exactly what happens when tested numerically, as seen on figure 13. The left side panel shows that as β_y is increased and β_x is tuned accordingly to keep an equal ratio $\frac{\beta_x}{\beta_y}$, the position of peak error ρ_* stays constant. On the right panel, four error curves computed from sampled values of β_y and β_x are plotted to compare the behaviour of the error curve. We see that as the noise gains are increased, the peak error grows

larger.

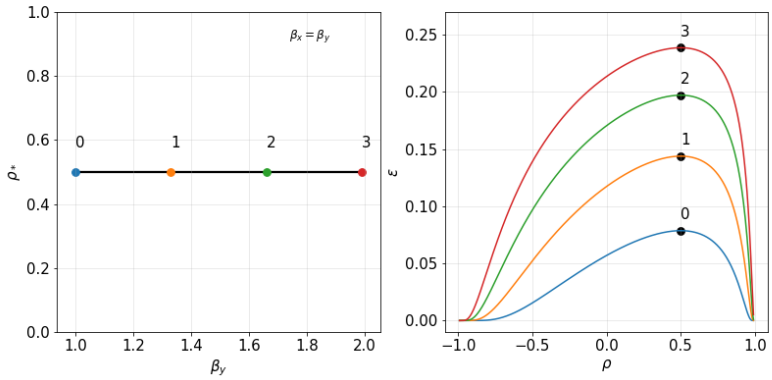

Figure 13: Correlation of peak error from both noise gains. Left panel, position of maximal error ρ_* as a function of noise gain β_x, while keeping the ratio $\frac{\beta_x}{\beta_y}$ constant. Coloured points, represent values of noise gain used in right panel. Right panel, samples of error curve with noise gains as shown in left panel.

This last result might seem trivial, however, it is not as such when only one noise gain is increased. If we were to include only the noise gain β_x in the subset g, then $f(\beta_x) = \frac{1}{\beta_x}$ and $c_{\beta_x} = \frac{\beta_y\sqrt{\tau_x\alpha_y}}{\sqrt{\tau_y\alpha_x}}$. For greater values of β_x, we would expect the magnitude of correlation to decrease proportionally to $\frac{1}{\beta_x}$ and for lower values of β_x, to increase proportionally to β_x. The discontinuity would happen at the value specified by c_{β_x} (resulting in $|\rho_*| = 1$). This is precisely the behaviour observed in figure 14. The left panel shows that in the case where the noise gain β_y is fixed, ρ_* (not its magnitude) decreases linearly with β_x until the discontinuity at $f(\beta_x) = c_{\beta_x}$ shown as a vertical dashed line. After this point, it increases as a function of $\frac{1}{\beta_x}$. When looking at the error curves resulting from sample β_x values on the right panel, we see that error still increases. However, one can follow the order in which β_x was increased by following the number indeces and the black dashed line and it would become apparent that first the peak error moves towards lower correlation values, and then reverses towards

higher correlation values.

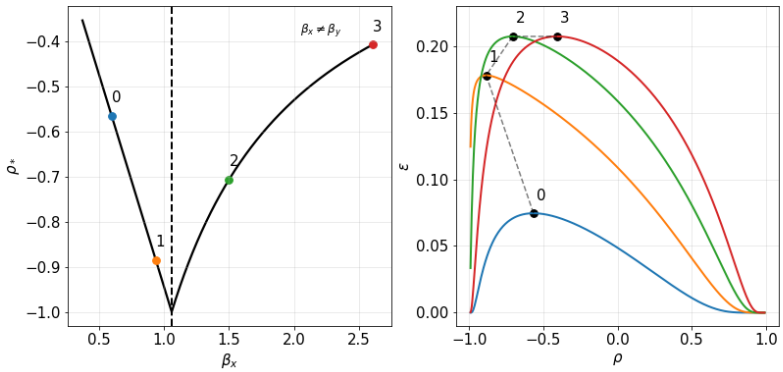

Figure 14: Correlation of peak error when increasing only the noise gain β_x. Left panel, position of maximal error ρ_* as a function of noise gain β_x, while keeping the value of β_y constant. Coloured points, represent values of noise gain used in right panel. Right panel, samples of error curve with noise gains as shown in left panel. Black dashed line showing the non-monotonic behaviour of rho_* in the error curves.

5.2 Translations of peak error

Hidden within this latest result is a counter-intuitive property of the model that is worth exploring. If one were to compare the curves associated with the sampled β_xs 1, 2 and 3, even though the peak error increases as β_x is increased, error throughout the whole domain of correlations is not always higher for higher noise gains. There are regions for which the error of the red curve (associated with the highest noise gain) falls under that of the green and yellow curves. To explore this latests result under better controlled conditions, we will first develop the conditions that would give rise to horizontal translations of the peak error, while conserving its height. That is, the conditions for which given an initial value $\rho_*^{(1)}$ it is possible to tune the network parameters such that the peak error happens at a new value $\rho_*^{(2)} \neq \rho_*^{(1)}$, yet $\varepsilon(\rho_*^{(2)}) = \varepsilon(\rho_*^{(1)})$.

We first define the translation to be given by $\rho_*^{(2)} = m\rho_*^{(1)}$ whit m being a non-null scalar. From the definition of ε in equation 28, the condition on the error peak height, is equivalent to a condition of equality between the squared Mahalanobis distances of the initial and modified error curves, $d_1^2 = d_2^2$. It is important here to not confuse the indexes 1 and 2 with indexes for different stimuli categories. The Mahalanobis distance, through the ratios r_u already contain the information for both the stimuli categories involved in the calculation of error rate. Here, the indexes simply refer to an original (1) and a modified value (2). The change from one condition to the other has to be dependent exclusively on the ratios r_u, so we set the final conditions that $r_{x2} = ar_{x1}$ and $r_{y2} = br_{y1}$, where a and b are also non-null scalars which cannot be simultaneously equal to one, since that would result in no change of correlation. For clarity, we list all four conditions here:

$$\begin{array}{ll} \rho_*^{(2)} = m\rho_*^{(1)} & d_1^2 = d_2^2 \\ r_{x2} = ar_{x1} & r_{y2} = br_{y1} \end{array}$$

Because of the picewise definition of ρ_*, four cases naturally arise from the combination of r_{uj} magnitudes. The full derivations can be found in annex A. First, we show the particular case where given a different set of network parameters, the error curve is conserved, *i.e.* how for arbitrary values of a and b, $m = 1$. The conditions for each case are presented on table 2, where the first column indicates the combination of ratio magnitudes, the second column and third columns, the values of the modified r_u from their original values, the fourth column, the relationship between m and the scalars a and b and the fifth columns, any extra necessary conditions for the case to be valid.

Figure 15 shows the application of conditions for the arbitrarily chosen third (green) case in table 2. The blue curve shows an initial curve for error rate derived from conditions r_{x1}, r_{y1}. The curve derived from modified conditions is

Table 2: Parameter relationship for a constant error curve

Case	r_{x2}	r_{y2}	m	Extra conditions
$r_{x1}^2 < r_{y1}^2,\ r_{x2}^2 < r_{y2}^2$	ar_{x1}	br_{y1}	$a = b$	$\|b\| = 1$
$r_{x1}^2 > r_{y1}^2,\ r_{x2}^2 > r_{y2}^2$	ar_{x1}	br_{y1}	$a = b$	$\|a\| = 1$
$r_{x1}^2 < r_{y1}^2,\ r_{x2}^2 > r_{y2}^2$	$\|r_{x2}\| = \|r_{y1}\|$	br_{y1}	$a = b^{-1}$	$\frac{r_{x1}^2}{r_{y1}^2} = \frac{1}{a^2}$
$r_{x1}^2 > r_{y1}^2,\ r_{x2}^2 < r_{y2}^2$	ar_{x1}	$\|r_{y2}\| = \|r_{x1}\|$	$a = b^{-1}$	$\frac{r_{y1}^2}{r_{x1}^2} = \frac{1}{b^2}$

shown in orange. While the two curves are derived from different ratios r_u, thus from different network parameters, their error rate as a function of correlation perfectly overlaps.

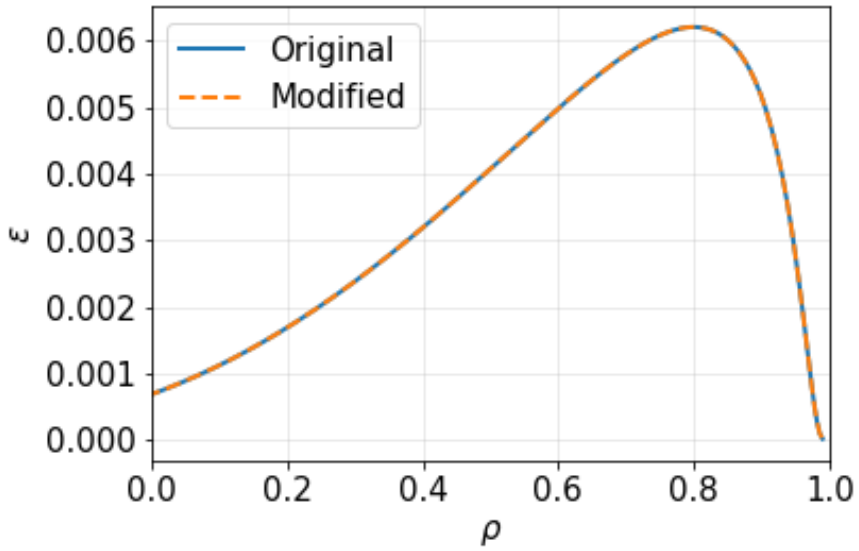

Figure 15: Constant value for peak error from change of parameters. Blue curve shows the error curve from original parameters. Orange dashed line shows it is possible to conserve the shape of the error curve after parameter changes under controlled conditions from table 2

More interestingly however, we can look at the more general case of horizontal translations by a factor m. The equivalent conditions for each of the four cases are presented in table 3.

Table 3: Conditions for horizontal translation of the error maximum

Case	r_{x2}	r_{y2}	m	Extra conditions
$r_{x1}^2 < r_{y1}^2,\ r_{x2}^2 < r_{y2}^2$	ar_{x1}	br_{y1}	$m = \frac{a}{b}$	$\lvert b\rvert = 1$
$r_{x1}^2 > r_{y1}^2,\ r_{x2}^2 > r_{y2}^2$	ar_{x1}	br_{y1}	$m = \frac{b}{a}$	$\lvert a\rvert = 1$
$r_{x1}^2 < r_{y1}^2,\ r_{x2}^2 > r_{y2}^2$	$\lvert r_{x2}\rvert = \lvert r_{y1}\rvert$	br_{y1}	$m = ab$	$\frac{r_{x1}^2}{r_{y1}^2} = \frac{1}{a^2}$
$r_{x1}^2 > r_{y1}^2,\ r_{x2}^2 < r_{y2}^2$	ar_{x1}	$\lvert r_{y2}\rvert = \lvert r_{x1}\rvert$	$m = ab$	$\frac{r_{y1}^2}{r_{x1}^2} = \frac{1}{b^2}$

Figure 16, presents translations for different values of m from an original distribution. We see that indeed, it is possible to control the position of the error peak on the horizontal axis, without altering its height, by tuning network parameters in controlled conditions.

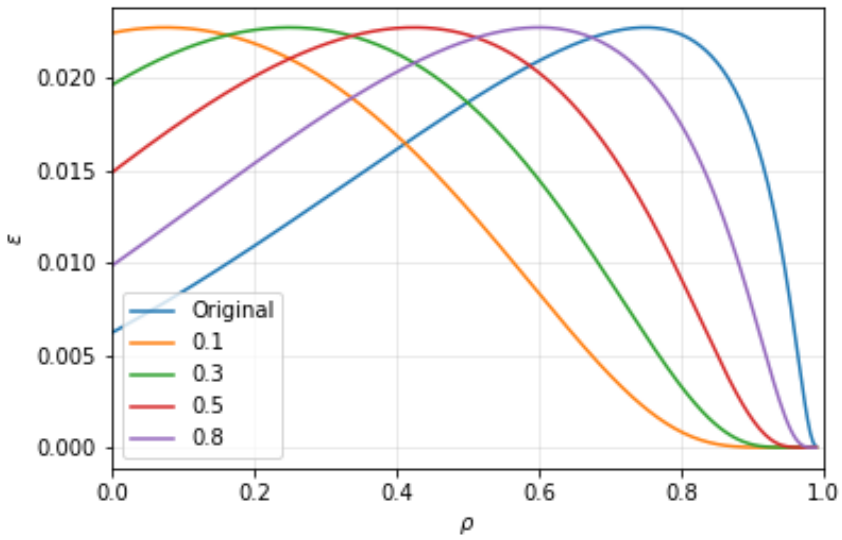

Figure 16: Translations of peak error on the dimension of correlations. Blue curve shows the error curve from original parameters. Coloured curves show translations by the factor shown in the legend respecting conditions in table 3
.

While this latest results can be seen mainly as mathematical oddities of our model, they help better visualize the aforementioned counter-intuitive results on

noise gain. As a tunable parameter, we can increase a noise gain β while keeping all other parameters constant and making sure to respect the conditions set by table 3. Figure 17 shows the resulting translated curve when increasing the noise gain from a value of 1 to a noise gain 10 times as high. It is possible to see that for all values to the left of the intersection point, the error rate of the curve derived from a higher noise gain is higher than the other curve's. Again, this should not be surprising, since intuitively, one would expect that adding noise would hinder classification. However, for the range of correlations to the right of the intersection point, the error curve associated with higher noise has a lower error rate. It means that for a certain regime, higher noise is advantageous to stimulus discrimination. To explain this behaviour, we can refer back to the geometric explanations offered in section 4. Geometrically, if we were to increase the noise gain of both units, both multivariate distributions of would extend in all directions, thus increasing the overlap between classes. When only one noise gain is increased, this causes an artificial stretch of the distributions on an individual dimension, which, if not compensated by the stretch generated by correlations, could make the distributions become more parallel and overlap less. In terms of Mahalanobis distance, the additional stretch of the distributions can make it so the variance in the direction of a vector joining the means would be reduced, which by definition increases the Mahalanobis distance between means and therefore reduces the error rate.

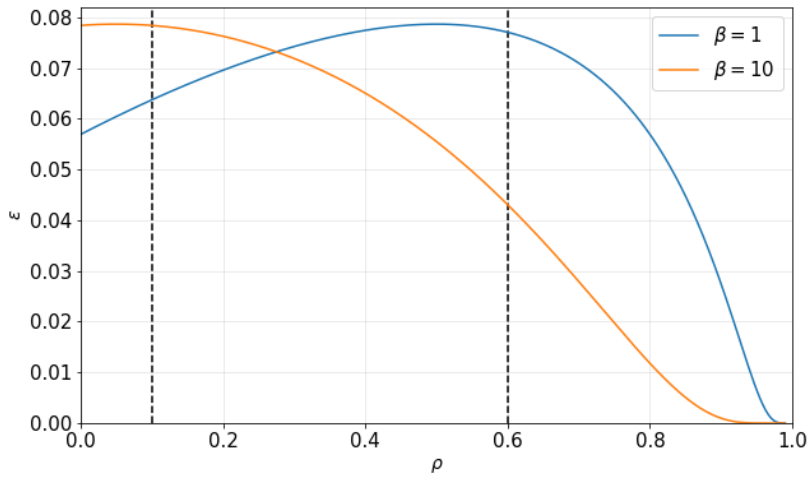

Figure 17: Samples of advantageous noise. Blue line, error curve for a noise gain of $\beta_x = 1$. Orange line, error curve for a relatively higher noise gain of $\beta_x = 10$. Left black dashed line shows a correlation level for which higher noise equates in higher error rate. Right black dashed line, correlation level at which a higher error rate equates in lower error rate.

6 Discussion

6.1 Summary of results

In this work, we show that the the effect of noise correlation on decoding error is highly dependent on the fine tuning of dynamic parameters of the network. Different parameter combinations can lead to qualitatively different decoding profiles thus providing a well controlled basis for exploration of the many contradictory results on the beneficial and detrimental effects of noise correlation in neural decoding.

We accomplished this goal by developing a model of network activity with few assumptions that could be decoded by a simple, biologically plausible linear classifier thus providing an upper bound on decoding capacity. Although the model itself was designed to be simple to use, the insights that can be derived from it can be rich and varied in that, not only it serves to account for expected behaviour, but also predicts the presence of more complex and somewhat counter-intuitive phenomena.

We began by developing an abstracted, two-unit neuronal network model derived from neural integrator models. These units were fully described by their mean rate, which was shown to be Gaussian distributed with finite mean and variance that could be calculated from network parameters. The activity of the two units thus jointly generated a generated multivariate Gaussian distribution whose mean was dependent on the stimulus and whose covariance matrix allowed the introduction of correlation in the noise between each component.

The joint activity of the integrator units would be read by a downstream unit able to classify the different distributions by stimuli as a perceptron whose weights were determined within an LDA framework. With closed-form solutions for the optimal categorization weights, we derived an expression for error rate that would be dependent on network parameters through the Mahalanobis

distance between means of distributions for different stimuli.

Being able to perform controlled tuning and exploration of parameters, we showed that the effect of neural correlation on decoding rate is generally non-monotonic, but for different conditions, error can also become monotonically decreasing or increasing for the full domain of correlations. Each of these conditions was representative of qualitatively different cases reported elsewhere [11].

Finally, we extended the results on the behaviour of the error curve to the implication of sub-sets of parameters in regulating the behaviour of the peak error rate. Of particular interest, we showed that the effect of noise gain on linear decoding can be more nuanced than would be intuitively thought. Indeed, we show that under certain conditions noise can multiplicatively increase the height of the error peak and the correlation level at which it happens. Further, under certain conditions, the increase in noise of the integrator units can lead to a reduction in error by the decoder unit.

6.2 Relevance

The principal empirical approach to the study of noise correlations is to measure the amount of information lost when decoding from biological networks using classifiers that ignore correlations by shuffling the signal against those that do not. An important issue with such an approach is that it is particularly hard to isolate network parameters in such a way that the full range of parameter space can be explored. This can result in a relatively limited view on the full extent of the effect of correlations, and its transition between qualitatively different states. It is unsurprising then that such a wide array of contradictory results on the effect of noise correlations have been reported, including beneficial, detrimental and null effects [11]. This is particularly true, when comparing across animal models, brain regions and sub-networks since their specific tuning might differ

greatly, which we have shown can have drastic effects on the conclusions one can derive from the model.

On this subject, in comparison to previous theoretical work, the main addition of our model is that the study of noise correlations is grounded on a less mathematically abstract scenario that emphasises on the dynamical properties of the network. Most often, when discussing neuronal decoding (as opposed to encoding), the quantifiable measure of decoding performance used is the difference in information, such as Shannon or Fisher information or distance measures such as differentiability (Mahalanobis distance in this text) and Bhattacharyya distance that can be decoded from shuffled and unshuffled neuronal responses. Most often, these types of approaches are not concerned with how such measures would come to be or could be represented as an embedded function of a biological system. In this respect, the closest theoretical studies to our derivations are those of [29], which are information-theoretic approaches for most parts, except that decoding is quantified by distance measures between multivariate distributions. An important distinction of our model however is how it manages to formalise the concept of distance between distributions within the framework of a biologically plausible network. Such change of paradigm taking the question of "what is the effect of noise correlations on decoding?" to "how is the read-out from a downstream unit within a biological network affected by noise correlations?" grounds the problem, pragmatically allowing it to explore the effect and role of biologically meaningful parameters such as stimulus magnitudes (ν_i) and integration time-scales (τ) in decoding rate. In practical terms, this allowed us to explain all possible qualitative different cases reported in the literature as naturally emerging from interactions between tunable parameters within a continuous space, which goes beyond simply considering shuffled versus unshuffled activity distributions. Furthermore, our results could extend to predictions

beyond the role of correlation in error curve by the study of the peak error. Our model allowed us to make important predictions on the role of noise in neural decoding, which could be some-what counter-intuitive at face-value and that could be extended to other relevant network parameters.

6.3 Limitations

6.3.1 Connectivity Weights

An important limitation concerning the generalizability of the model is the lack of explicit connectivity between units. Most commonly, when modelling interconnected units, one would rewrite equation 8 as equation 34, with explicit weighted connections.

$$\begin{aligned} \tau_x \frac{dx}{dt} &= -\alpha_x x + \nu_i + \beta_x \xi_x(t) + \omega_{xy} y(t) \\ \tau_y \frac{dy}{dt} &= -\alpha_y y + \nu_i + \beta_y \xi_y(t) + \omega_{yx} x(t) \end{aligned} \tag{34}$$

The problem with such a formulation is that it fails to completely isolate the effect of noise correlations from other network dynamics. Firstly, explicit connectivity as the one in 34 does not introduce any noise correlations by itself. Indeed, the source of noise is still only dependent on the gain β. What such connectivity does is increase the signal correlation, which is a subject of itself altogether and beyond the scope of this work. As 34 stands, one could sample the noise term for the whole length of a simulation and simply superpose it to the numerical solution of the ODE with no difference in resulting activity. Such an often proposed model, would therefore be inadequate to the subject matter.

A much better candidate for an interconnected higher dimensional model would unsurprisingly be the multi-dimensional Ornstein-Uhlenbeck process de-

fined as 35.

$$d\boldsymbol{x} = \boldsymbol{\alpha}\boldsymbol{x}dt + \boldsymbol{\sigma}\boldsymbol{dW} \tag{35}$$

where $\boldsymbol{x}$ and $\boldsymbol{dW}$ are $N \times 1$ vectors and $\boldsymbol{\alpha}$ and $\boldsymbol{\sigma}$ are $N \times N$ square matrices. As with univariate O-U processes, one can solve the coupled stochastic differential equations, resulting in expressions for mean and covariance across units. The product $\boldsymbol{\sigma}\boldsymbol{dW}$ introduces noise correlations between random variables since it effectively results in a a vector of normally distributed linear combinations of normal random variables. Further work on the subject would benefit greatly from adopting this approach for more naturally interconnected units. One potential issue would be the addition of the coupling of unit rates through the product $\boldsymbol{\alpha}\boldsymbol{x}$. Firstly, this would again fail to isolate noise from signal correlations. Secondly, for different ranges of parameter values, this could cause the model to fail to reach a stationary state, thus making it less feasible to respect basic assumptions for LDA in terms of the long-term distribution of firing rates. This limitation could very easily be adressed by making the $N \times N$ matrix $\boldsymbol{\alpha}$ diagonal. This would result in uncoupled stochastic equations that would be solved as the one presented originally in this text while creating noise correlations between units. From there, it would only be necessary to create an easily tunable function that would map noise parameters $\boldsymbol{\sigma}$ and its equivalent covariance matrix.

6.3.2 Dimensionality and linear decoding

The model, as it has been presented in this document, potentially suffers from a problem of low-dimensionality. This can be seen in two different features of the model: the limited number of distinct inputs (*i.e.* ν_1 and ν_2 exclusively) and the limited number of processing sub-networks or units (*i.e* x and y). With respect

to the former, ideally, we would want to extend the model's conclusions to multi-class classification from a vector of inputs $\overrightarrow{\nu} = [\nu_1, \nu_2, ..., \nu_I]^T$ of arbitrary length I. For the latter, the results presented in this document should simply be the special two-dimensional case of an N-dimensional multivariate distribution generated from N different units $x^{(n)}$ (where $x^{(1)}$ and $x^{(2)}$ are respectively units x and y in this document). We address this limitations separately.

First, we consider the case of multi-class classification. For simplicity and ease of visualization, we consider the case of only two processing units or sub-networks, *i.e* a two-dimensional feature-space and classification surface. LDA is inherently a binary classifier, meaning that it can define a linear boundary that separates only two statistical representations of the sampled distributions. In terms relating to the perceptron rule, LDA can only determine if the weighted sum of inputs is above or below threshold, with no further graduation. Given this limitation, it is still possible to perform multi-class classification task by adopting one of two strategies: "one-vs-rest" (1vR) or "one-vs-one" (1v1) classification. The first, 1vR would consider during training each individual distribution against the combined distribution of all other individual distributions. This would mean that to determine whether a new sample of network activity corresponds to input ν_1, it would attempt to discriminate between ν_1 against the mixture of all other $N - 1$ distributions. This approach would be particularly weak, given that the mixture of multiple Gaussian is (most likely) not a Gaussian distribution itself so the "mixture" class against one would compare ν_1 would violate one of the key assumptions of LDA. On the other hand, a 1v1 classification seems like a more likely candidate. The 1v1 strategy addresses the multi-class problem by building $\frac{I(I-1)}{2}$ pairwise classifiers, *i.e.* a classifier per pair of stimuli, and combining all classifications into a final decision. This could be done either as a "voting" strategy or a probability-weighted summed

strategy. As a multi-layered perceptron, this could be adapted as in figure 18, where, trained to classify among three different inputs, the network divides into three different classifiers. From there, this could forward-feed to a new layer whose units perform the summation of activations from different binary classifiers, and this final binary vector is deconstructed by another layer, capable of detecting the dimension with most activity. While possible for such a limited range of inputs, such architecture would require prior knowledge of all possible inputs or recruiting, reorganizing the network to accommodate for new inputs as well as very fine tuning for the limited discrimination task.

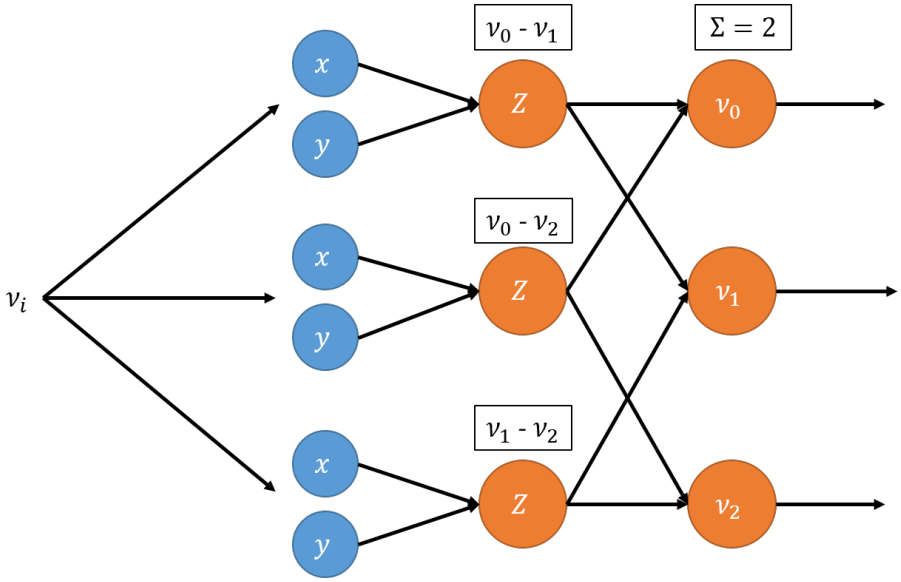

Figure 18: Schematic of a multi-layered perceptron architecture capable of performing 1v1 type classification. The common input is treated by a series of linear classifiers which feed into a layer tuned for the summation of "votes". The comparison of votes is done by a final layer capable of dimension reduction

Second, we consider the case of a higher dimensional feature space, composed of N units $x^{(n)}$. Linear discriminant analysis is inherently a high-dimensional classifier. This is particularly evident when looking at its Bayesian definition on equation 4 which explicitly depends on the dimensionality of feature space N.

In that sense, the results presented here should easily scale to an N dimensional space representation of the network as a whole. However, this "Bayesian-black-box" solution or explanation is not satisfactory in the context of the present document. Indeed, while computations in the brain can be explained and described through Bayesian theory [30, 31], in relation to linear classifiers, it appears as a more engineered approach than the perceptron inspired approach used throughout the analysis presented here. It would be useful then, to provide a higher dimensional extension to our model that would be equally inspired by a perceptron architecture with biologically plausible learning mechanisms. This could be achieved by simple redefinitions. As it is, the multivariate distributions that need to be classified can be described as:

$$Z_i = X_i = [x_i, y_i]^T \tag{36}$$

Where X_i would be more generally written as $X_i = [x_i^{(1)}, ..., x_i^{(N)}]$ in N dimensional notation. However, one could also change paradigm away from the one-to-one mapping between sub-network activity and read-out input. Instead of having to linearly discriminate within N-dimensional space, one could make it so the linear-readout is done on an arbitrarily high-dimensional space, two-dimensional for practical reasons for example, where each dimension of the input-space into the readout unit is a linear combination of the activity of N units. Mathematically, this is equivalent to rewriting equation 36 as:

$$Z_i = MX_i \tag{37}$$

Where M is a rectangular $D \times N$ weight matrix with N being the number of units or sub-networks $x_i^{(n)}$ and D the dimensionality of the input-space into the read-out unit. The whole architecture of the model could then be re-conceptualized

as a feed-forward multi-layer perceptron as shown in figure 19.

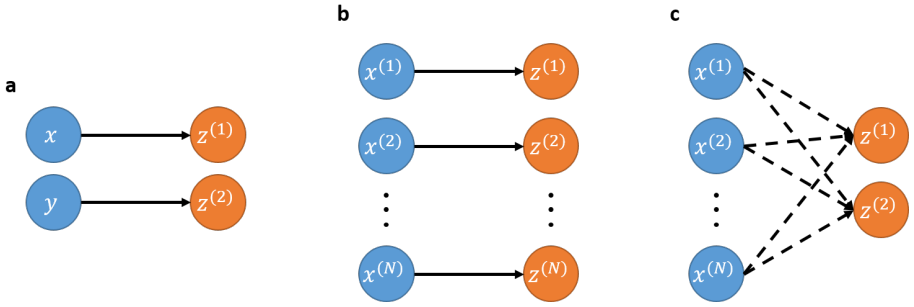

Figure 19: Alternative architectures to incorporate higher dimensional inputs into the readout. Blue circles represent input units. Orange circles represent dimensions of the read-out unit. Solid black arrows represent one-to-one mappings. Dashed arrows represent weighted connections. (a) shows the original model discussed in the document. (b) shows one-to-one extended model to higher-dimensional input. (c) shows weighted integration of inputs to integrate higher-dimensional inputs.

With the new paradigm, one could move from the original model's architecture seen in panel (a) of figure 19 to that of panel (c) without needing to extend the model to higher dimensional feature space as that of (b). The linear combination of activity of all sub-networks is not only more practical given the main analysis of this document, but it is more biologically relevant than a fully parallelized, one-to-one mapping conceptualisation shown in (b). Furthermore, a crucial advantage of such an approach is that by Central Limit Theorem, as long as the individual contributions of each unit is stable in time, we can expect the resulting read-out distributions to be multivariate Gaussian, irrespective of the distributions of the original units. This in turn provides a way to expand the model not only to higher dimensions, but to a broad range of distributions of activity.

One caution to remark from model (c) of figure 19 with respect to the main analysis is that the new dimensions of the read-out unit Z are no longer independent, meaning that the parameter ρ used in the main analysis to tune

correlations would have to be modified to accommodate for the introduced dependencies.

6.3.3 Unequal class covariances

An important assumption of the original model is that, while the variance of each unit can be unique ($\sigma_x = \frac{\beta_x^2}{2\tau_x \alpha_x}$), the class covariance matrices are equal ($\Sigma_1 = \Sigma_2 = \Sigma$). As the indexes of the covariance matrix indicates, this would be different if any of the elements of the covariance matrix was input-dependent, or simpler, if any of the variances σ_x^2, σ_y^2 was input dependent. From the expression of variance as a function of network parameters, we see that this condition does not necessarily mean that no netwrok parameter can be input dependent. Indeed, if more than one of them would be a function of input, it would be possible that they be defined in such a way that they would cancel each others effect and the variance itself of each unit was not dependent on the input value. For example, if $\beta^2 = f(\nu_i), \alpha = f(\nu_i)^2$, this would mean that $\sigma = \frac{1}{2\tau}$ which is not input dependent, thus would not affect the covariance matrix while allowing a model that that could prove quite interesting, described as:

$$\tau_x \frac{dx}{dt} = -f(\nu_i)^2 x + \nu_i + f(\nu_i)\xi_x(t) \tag{38}$$

This condition for the class covariance matrix is however quite restrictive, and even if it was respected, it does not mean that the resulting distributions would always exhibit mean-reverting, Gaussian distributed steady regimes, a necessary condition to use LDA.

For Gaussian distributed classes with different class covariances, it is necessary to consider interaction effects between variables. One way to work around this is to extend the classification model with N units to one that includes the $\frac{N^2}{2} + N$ interaction effects, a technique reminiscent of polynomial regression as

an extension to linear regression. In terms of topology, this could be achieved by using higher-order units such as a sigma-pi instead of linear perceptron units.

$$[x, y] \; becomes \; [x, y, x^2, y^2, xy] \tag{39}$$

Such a restructuring of feature-space would then be able to accomodate for the different class covariances in that classes in the new feature-space would become linealry-separable again.

Alternatively, one could implement Quadratic Discrimnant Analysis (QDA) instead of LDA (which is indirectly what was done in the previous alternative). QDA, as opposed to LDA has a decision boundary that is given by 40.

$$X^T AX + b^T X + c \tag{40}$$

More simply, in Bayesian terms, the probability of a given point to be part of class k is given by the same equation presented earlier for LDA (equations 3, 3), meaning that a simlar analysis could be performed as long as the data distributions remain Gaussian and the priors can be calculated.

6.3.4 LDA comparison with PCA

As discussed previously, in terms of classification LDA has a clear advantage compared to more powerful algorithms such as support-vector-machines in that it has a closed-form solution for its weights. However, It would have been equally possible to first use a principal component analysis (PCA) on the mixture covariance matrix, which at low-dimension might allow for analytical eigenvalue decomposition and perform the projection on the eigenvector that explained the highest percentage of the variance. Particularly, this would have lead to dimensionality reduction unbound by the assumption of normality and shared

covariance, and then the same logic applied to the LDA method, namely, the integration of the overlap of density functions, could have been applied to find an estimate of the error. While this might be a valid line of inquiry in other cases, in the context analysed here, it might have not been an adequate one. This is because a defining feature of LDA is how it performs dimensionality reduction on the feature space prior to classification. Given a set of N-dimensional distributions (*i.e.* N distinct sub-networks feeding into the linear-readout), it performs discrimination on a projected $N-1$ dimensional space, using as a projection line, the dimension that would result in the biggest distance between distribution means when scaled by their variance. On the other hand, PCA reduces feature-space dimensionality by an arbitrary (or at least, not algorithmically or mathematically chosen) factor by projecting against the dimension with the largest spread of data. This difference can lead to very different separation of classes on the projected space. We consider a two-dimensional case that represents the advantage of LDA's dimensionality reduction in figure 6.3.4.

It can be seen in figure 6.3.4, that under conditions which are normal in the context of this document, the projection resulting from LDA creates a much better separation than PCA. Indeed, when we compare the resulting projections from the first distributions, those that stretch over a common line, we see that both methods create similar projections, with as much overlap between projected distributions. This is because the direction of largest separation happens to be aligned with that of biggest variance. On the other hand, when looking at results on the right column of figure 6.3.4, we see that as soon as a slight distance is inserted between distributions, LDA is able to find a projection of data that greatly reduces the overlap between projected distributions, since the optimal projection line is no longer aligned with that of the largest spread of the data as with PCA.

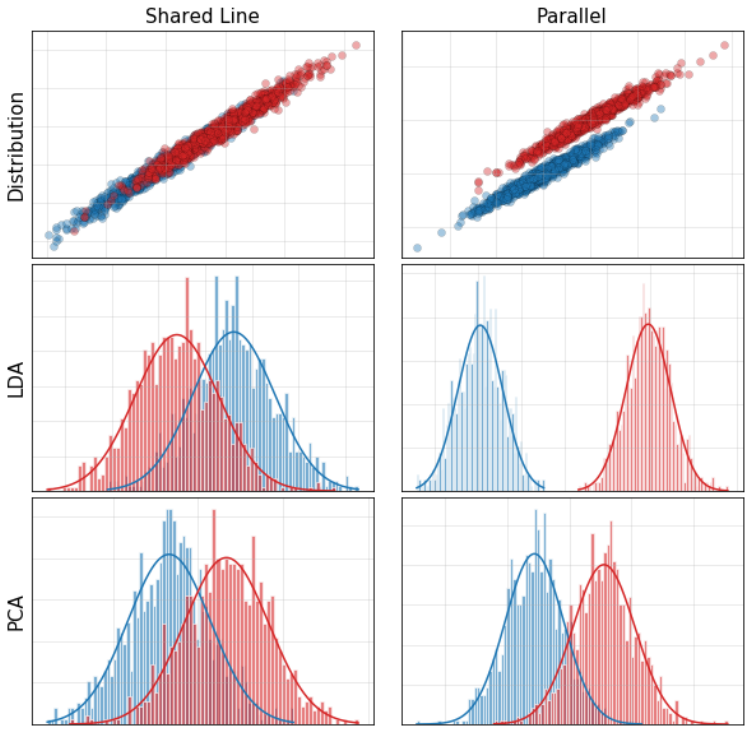

Figure 20: Comparison of dimensionality reduction performed by LDA and PCA. First row, two different pairs of distributions to be classified with the class of each data point being represented by its color. Second row, lower-level projections of the classes given by LDA. Third row, lower-level projections of the classes given by PCA. Right column, the two distributions stretch along a common line. Left column, the two distributions stretch along parallel lines.

7 Conclusion

The contribution of noise correlations to neural decoding still is a subject of debate both empirically and theoretically. The subject is particularly important in that an effect of correlation, and more specifically dependence across units of a statistical system signifies that the individual unit cannot be assumed to be the basic processing element of the network. Practically, this means that results from single-cell or observations done under the assumption of independence across neurons cannot be accurately extrapolated to the scale of the network.

From a simple network model being read-out solely by a linear decoder, we have shown that any conclusion on the beneficial or detrimental effect of noise correlation must include a thorough exploration of parameter space. This space must cover size effects and network heterogeneity, such as previously discussed work, as well as dynamic properties of the network.

Further work should focus on more detailed descriptions of neuronal networks. The model presented here should be considered and used as a behavioural framework to study network activity. This means that while the starting point in this work was the integrator model, the analysis of decoding only requires the statistical parameters, *i.e.* the mean and the variance of the activity of each unit. In that sense, the analytical stage is already set to test any arbitrary model of network activity through the same procedure described in this work. In this respect, we hope to emphasize, not only the ensemble of results which have proven to be informative in unexpected ways, but also on the method itself, which could be promising as a cornerstone for further behavioural implementations. Future work should firstly involve the expansion of the model to higher dimensional space. Secondly, with the goal of more thorough exploration of parameter space, it would prove useful to replace the mean rate model by a biophysical network model from which the mean and the variance of the rate

can be calculated and the influence or relevant parameters can be studied. This could, for example include single cell dynamical parameters as well as network properties such as the balance of excitation and inhibition.

In sum, the approach has not only shown promise in this initial work but is designed in such a way that can be readily extended to future work. It it is the hope of the author that this work will simultaneously shed light on the role of noise correlations as well as opening a new line of inquiry into the theoretical study in the context of a biologically plausible task.

8 References

[1] C. F. Stevens, "Inferences about Membrane Properties from Electrical Noise Measurements," *Biophysical Journal*, vol. 12, no. 8, pp. 1028–1047, 1972.

[2] M. Voelker and P. Fromherz, "Nyquist noise of cell adhesion detected in a neuron-silicon transistor," *Physical Review Letters*, vol. 96, no. 22, 2006.

[3] T. Prodromakis, R. W. Berg, A. Parihar, A. Raychowdhury, M. Jerry, and S. Datta, "Stochastic IMT (Insulator-Metal-Transition) Neurons: An Interplay of Thermal and Threshold Noise at Bifurcation," *Front. Neurosci*, vol. 12, p. 210, 2018.

[4] A. F. Strassberg and L. J. DeFelice, "Limitations of the Hodgkin-Huxley Formalism: Effects of Single Channel Kinetics on Transmembrane Voltage Dynamics," *Neural Computation*, vol. 5, no. 6, pp. 843–855, 1993.

[5] Lockery and Goodman, "Action Potential in C.Elegans," *Nature Neurosci*, vol. 12, 2009.

[6] M. Güler, "Dissipative stochastic mechanics for capturing neuronal dynamics under the influence of ion channel noise: Formalism using a special membrane,"

[7] A. A. Faisal and J. A. White, "Ion-Channel Noise Places Limits on the Miniaturization of the Brain's Wiring," *Current Biology*, vol. 15, pp. 1143–1149, 2005.

[8] R. F. Pena, M. A. Zaks, and A. C. Roque, "Dynamics of spontaneous activity in random networks with multiple neuron subtypes and synaptic noise: Spontaneous activity in networks with synaptic noise," *Journal of Computational Neuroscience*, vol. 45, no. 1, 2018.

[9] T. Womelsdorf, B. Lima, M. Vinck, R. Oostenveld, W. Singer, S. Neuenschwander, and P. Fries, "Orientation selectivity and noise correlation in awake monkey area V1 are modulated by the gamma cycle," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 11, pp. 4302–4307, 2012.

[10] M. R. Cohen and A. Kohn, "Measuring and interpreting neuronal correlations," 2011.

[11] B. B. Averbeck, P. E. Latham, and A. Pouget, "Neural correlations, population coding and computation," 2006.

[12] E. Zohary, M. N. Shadlen, and W. T. Newsome, "Correlated neuronal discharge rate and its implications for psychophysical performance," *Nature*, vol. 370, no. 6485, pp. 140–143, 1994.

[13] P. Alexandre, D. Peter, and Z. Richard, "Information processing with population codes," *Nature Reviews Neuroscience*, vol. 1, no. November, pp. 125–132, 2000.

[14] B. Cybernetics, "Paradiso , M . A . A theory of the use of visual orientation information which exploits the columnar structure of striate cortex . Biol . Biological Cybernetics," *Biol. Cybern*, vol. 58, no. January, pp. 35–49, 2016.

[15] E. Salinas, "Vector Reconstruction from Firing Rates," tech. rep., 1994.

[16] E. C. Chang and W. G. Lewellen, "Market Timing and Mutual Fund Investment Performance Author (s): Eric C . Chang and Wilbur G . Lewellen Published by : The University of Chicago Press Stable URL : http://www.jstor.org/stable/2352888 Accessed : 11-04-2016 10 : 16 UTC

Your use of the JST," *The Journal of Business*, vol. 57, no. 1, pp. 57–72, 1984.

[17] C. J. Mcadams and J. H. R. Maunsell, "Effects of Attention on the Reliability of Individual Neurons in Monkey Visual Cortex proportionally and does not improve the selectivity of single neurons, as measured by the width of their tuning curve (Vogels and Orban," tech. rep., 1999.

[18] A. Schoups, R. Vogels, N. Qian, and G. Orban, "Practising orientation identification improves orientation coding in V1 neurons," *Nature*, vol. 412, no. 6846, pp. 549–553, 2001.

[19] T. Yang and J. H. R. Maunsell, "The effect of perceptual learning on neuronal responses in monkey visual area V4.," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 24, no. 7, pp. 1617–26, 2004.

[20] A. Pouget, S. Deneve, and P. E. Latham, "Reading population codes: a neural implementation of ideal observers," *Nature Neuroscience*, vol. 2, no. 8, pp. 740–745, 1999.

[21] N. Berberian, A. MacPherson, E. Giraud, L. Richardson, and J. P. Thivierge, "Neuronal pattern separation of motion-relevant input in lip activity," *Journal of Neurophysiology*, vol. 117, no. 2, pp. 738–755, 2017.

[22] J. D. Touboul and G. B. Ermentrout, "Finite-size and correlation-induced effects in mean-field dynamics," *Journal of Computational Neuroscience*, vol. 31, no. 3, pp. 453–484, 2011.

[23] R. Moreno-Bote, J. Beck, I. Kanitscheider, X. Pitkow, P. Latham, and A. Pouget, "Information-limiting correlations," 2014.

[24] D. V. Buonomano and W. Maass, "State-dependent computations: Spatiotemporal processing in cortical networks," 2009.

[25] P. D'Souza, S. C. Liu, and R. H. Hahnloser, "Perceptron learning rule derived from spike-frequency adaptation and spike-time-dependent plasticity," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 107, no. 10, pp. 4722–4727, 2010.

[26] Fisher A R, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[27] N. Cain, A. K. Barreiro, M. Shadlen, and E. Shea-Brown, "Neural integrators for decision making: a favorable tradeoff between robustness and sensitivity," *J Neurophysiol*, vol. 109, pp. 2542–2559, 2013.

[28] D. A. Gutnisky, C. B. Beaman, S. E. Lew, and V. Dragoi, "Spontaneous fluctuations in visual cortical responses influence population coding accuracy," *Cerebral Cortex*, vol. 27, no. 2, pp. 1409–1427, 2017.

[29] B. B. Averbeck and D. Lee, "Effects of Noise Correlations on Information Encoding and Decoding," *Journal of Neurophysiology*, vol. 95, no. 6, pp. 3633–3644, 2006.

[30] R. Moreno-Bote, D. C. Knill, and A. Pouget, "Bayesian sampling in visual perception," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 108, no. 30, pp. 12491–12496, 2011.

[31] C. Bielza and P. Larrañaga, "Bayesian networks in neuroscience: A survey," 2014.

A Appendix: Mathematical supplement

A.1 Solving the Integrator model as a linear differential equation

We attempt to solve the equation for the firing rate of the integrator model. To alleviate the notation we drop the unit indexes.

$$\begin{aligned}
&\tau\frac{dx}{dt} = -\alpha x + \nu_i + \beta\xi(t) \\
\Leftrightarrow &\frac{dx}{dt} = -\frac{\alpha}{\tau}x + \frac{\nu_i + \beta\xi(t)}{\tau} \\
\Leftrightarrow &\frac{dx}{dt} + \frac{\alpha}{\tau}x = \frac{\nu_i + \beta\xi(t)}{\tau} \\
\Leftrightarrow &\frac{dx}{dt} + p(t)x = r(t)
\end{aligned}$$

With $p(t) = \frac{\alpha}{\tau}$, $r(t) = \frac{\nu_i + \beta\xi(t)}{\tau}$. From there, we define a function $u(t)$ such that:

$$\begin{aligned}
u(t) &= e^{\int p(t)dt} \\
&= e^{\int \frac{\alpha}{\tau}dt} \\
&= e^{\frac{\alpha}{\tau}t}
\end{aligned}$$

From there, then:

$$\begin{aligned}
&u(t)\left(\frac{dx}{dt} + p(t)x\right) = u(t)r(t) \\
\Leftrightarrow\ &e^{\frac{\alpha}{\tau}t}\frac{dx}{dt} + e^{\frac{\alpha}{\tau}t}\frac{\alpha}{\tau}x = e^{\frac{\alpha}{\tau}t}r(t)
\end{aligned}$$

From the chain rule, this means that:

$$\begin{aligned}
\frac{d}{dt}\left(u(t)x\right) &= u(t)r(t) \\
\Leftrightarrow d\left(u(t)x\right) &= u(t)r(t)dt \\
\Leftrightarrow \int_0^t d\left(u(s)x\right) &= \int_0^t u(s)r(s)ds \\
\Leftrightarrow u(s)x|_0^t &= \int_0^t u(s)r(s)ds \\
\Leftrightarrow u(t)x - u(0)x_0 &= \int_0^t u(s)r(s)ds \\
\Leftrightarrow x &= u(t)^{-1}\left(x_0 + \int_0^t u(s)r(s)ds\right) \\
&= e^{-\frac{\alpha}{\tau}t}\left(x_0 + \int_0^t e^{\frac{\alpha}{\tau}s}\frac{\nu_i + \beta\xi(s)}{\tau}ds\right) \\
&= e^{-\frac{\alpha}{\tau}t}\left(x_0 + \frac{1}{\tau}\int_0^t e^{\frac{\alpha}{\tau}s}\left(\nu_i + \beta\xi(s)\right)ds\right) \\
&= e^{-\frac{\alpha}{\tau}t}\left(x_0 + \frac{1}{\tau}\int_0^t e^{\frac{\alpha}{\tau}s}\nu_i ds + \frac{1}{\tau}\int_0^t e^{\frac{\alpha}{\tau}s}\beta\xi(s)ds\right) \\
&= e^{-\frac{\alpha}{\tau}t}\left(x_0 + \frac{1}{\tau}\frac{\tau\nu_i}{\alpha}e^{\frac{\alpha}{\tau}s}|_0^t + \frac{\beta}{\tau}\int_0^t e^{\frac{\alpha}{\tau}s}\xi(s)ds\right) \\
&= e^{-\frac{\alpha}{\tau}t}\left(x_0 + \frac{\nu_i}{\alpha}\left[1 - e^{\frac{\alpha}{\tau}t}\right] + \frac{\beta}{\tau}\int_0^t e^{\frac{\alpha}{\tau}s}\xi(s)ds\right) \\
&= x_0 e^{-\frac{\alpha}{\tau}t} - \frac{\nu_i}{\alpha}\left[1 - e^{-\frac{\alpha}{\tau}t}\right] + e^{-\frac{\alpha}{\tau}t}\frac{\beta}{\tau}\int_0^t e^{\frac{\alpha}{\tau}s}\xi(s)ds \\
x &= \frac{\nu_i}{\alpha} + \left[x_0 - \frac{\nu_i}{\alpha}\right]e^{-\frac{\alpha}{\tau}t} + \frac{\beta}{\tau}\int_0^t e^{-\frac{\alpha}{\tau}(t-s)}\xi(s)ds
\end{aligned}$$

A.2 Expected value and Variance

We are interested in finding the mean and variance of the random variable x such that:

$$x = \mu + \left(x_0 - \mu\right)e^{-\theta t} + \sigma\int_0^t e^{-\theta(t-s)}dB_s$$

The mean first:

$$\begin{aligned} E[x] &= E\left[\mu + (x_0 - \mu)\, e^{-\theta t} + \sigma \int_0^t e^{-\theta(t-s)} dB_s\right] \\ &= E[\mu] + E\left[(x_0 - \mu)\, e^{-\theta t}\right] + E\left[\sigma \int_0^t e^{-\theta(t-s)} dB_s\right] \end{aligned}$$

The zero mean property of of Ito integrals of simple adapted processes stipulates that:

$$E\left[\sigma \int_0^t e^{-\theta(t-s)} dB_s\right] = 0$$

Therefore:

$$E[x] = \mu + (x_0 - \mu)\, e^{-\theta t} \tag{41}$$

Now, the variance:

$$\begin{aligned} var(x) &= var\left(\mu + (x_0 - \mu)\, e^{-\theta t} + \sigma \int_0^t e^{-\theta(t-s)} dB_s\right) \\ &= \sigma^2 var\left(\int_0^t e^{-\theta(t-s)} dB_s\right) \\ &= \sigma^2 \left(E\left[\left(\int_0^t e^{-\theta(t-s)} dB_t\right)^2\right] - E\left[\int_0^t e^{-\theta(t-s)} dB_s\right]^2\right) \\ &= \sigma^2 E\left[\left(\int_0^t e^{-\theta(t-s)} dB_s\right)^2\right] \end{aligned}$$

By Ito isometry:

$$\sigma^2 E\left[\left(\int_0^t e^{-\theta(t-s)} dB_s\right)^2\right] = \sigma^2 E\left[\int_0^t \left(e^{-\theta(t-s)}\right)^2 ds\right]$$

Therefore, the variance:

$$
\begin{aligned}
var(x) &= \sigma^2 \int_0^t e^{-2\theta(t-s)} ds \\
&= \sigma^2 \left[-\frac{e^{-2\theta(t-s)}}{-2\theta} \right]_0^t \\
&= \frac{\sigma^2}{2\theta} \left[e^{-2\theta(t-t)} - e^{-2\theta(t-0)} \right] \\
var(x) &= \frac{\sigma^2}{2\theta} \left[1 - e^{-2\theta t} \right]
\end{aligned}
$$

A.3 Error integral

The classification error as a function of integrals is given by:

$$
\begin{aligned}
\varepsilon &= \frac{1}{2} \int_{-\infty}^{0} \frac{1}{\sqrt{2\varsigma^2\pi}} e^{\frac{-(w-\eta_1)^2}{2\varsigma^2}} dw + \frac{1}{2} \int_{0}^{\infty} \frac{1}{\sqrt{2\varsigma^2\pi}} e^{\frac{-(w-\eta_0)^2}{2\varsigma^2}} dw \\
&= \frac{1}{2} \frac{1}{\sqrt{2\varsigma^2\pi}} \left[\sqrt{\frac{\pi\varsigma^2}{2}} erfc\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) + \sqrt{\frac{\pi}{2}} \left(\sqrt{\varsigma^2} erf\left(\frac{\eta_0}{\sqrt{2\varsigma^2}} \right) + \sqrt{\varsigma^2} \right) \right] \\
&= \frac{1}{2} \frac{1}{\sqrt{2\varsigma^2\pi}} \sqrt{\frac{\pi\varsigma^2}{2}} \left[erfc\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) + erf\left(\frac{\eta_0}{\sqrt{2\varsigma^2}} \right) + 1 \right] \\
&= \frac{1}{4} \left[1 - erf\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) + erf\left(\frac{\eta_0}{\sqrt{2\varsigma^2}} \right) + 1 \right] \\
&= \frac{1}{4} \left[1 - erf\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) + erf\left(\frac{\eta_0}{\sqrt{2\varsigma^2}} \right) + 1 \right] \\
&= \frac{1}{4} \left[2 - erf\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) + erf\left(\frac{-\eta_1}{\sqrt{2\varsigma^2}} \right) \right] \\
&= \frac{1}{4} \left[2 - erf\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) - erf\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) \right] \\
&= \frac{1}{2} \left[1 - erf\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right) \right] \\
&= \frac{1}{2} erfc\left(\frac{\eta_1}{\sqrt{2\varsigma^2}} \right)
\end{aligned}
$$

When replacing the mean and variance values from section A.4, this is equivalent to:

$$\begin{aligned}\varepsilon &= \frac{1}{2} erfc\left(2\frac{d^2}{4\sqrt{2d^2}}\right) \\ &= \frac{1}{2} erfc\left(\frac{1}{2\sqrt{2}}\sqrt{d^2}\right)\end{aligned}$$

A.4 Shifted means and variance to mahalanobis distance

We start with the following definitions:

$$\begin{aligned}W &= (2\Sigma)^{-1}\Delta\boldsymbol{\mu} \\ c &= W \cdot \frac{1}{2}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0}) + b \\ \eta_i &= W \cdot \boldsymbol{\mu_i} + b - c \\ \varsigma^2 &= W^T \Sigma W\end{aligned}$$

We first take η_i and replace c by its full definition:

$$\begin{aligned}\eta_i &= W \cdot \boldsymbol{\mu_i} + b - \left(W \cdot \frac{1}{2}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0}) + b\right) \\ &= W \cdot \boldsymbol{\mu_i} + b - W \cdot \frac{1}{2}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0}) - b \\ &= W \cdot \left(\boldsymbol{\mu_i} - \frac{1}{2}(\boldsymbol{\mu_1} + \boldsymbol{\mu_0})\right)\end{aligned}$$

From here, we can determine that:

$$\begin{aligned}\eta_1 &= \frac{1}{2} W \cdot (\boldsymbol{\mu_1} - \boldsymbol{\mu_0}) \\ &= \frac{1}{2} W \cdot \Delta\boldsymbol{\mu} \\ &= -\eta_0\end{aligned}$$

We then expand W from its full definition and use the property $u \cdot v = u^T v$:

$$\begin{aligned}\eta_1 &= \frac{1}{2} \left[(2\Sigma)^{-1}\Delta\boldsymbol{\mu}\right] \cdot \Delta\boldsymbol{\mu} \\ &= \frac{1}{4} \left[(\Sigma)^{-1}\Delta\boldsymbol{\mu}\right]^T \Delta\boldsymbol{\mu} \\ &= \frac{1}{4} \Delta\boldsymbol{\mu}^T {\Sigma^{-1}}^T \Delta\boldsymbol{\mu} \\ &= \frac{1}{4} \Delta\boldsymbol{\mu}^T \Sigma^{-1} \Delta\boldsymbol{\mu}\end{aligned}$$

We introduce here the squared-mohalanobis distance between means defined as:

$$d^2 = \Delta\boldsymbol{\mu}^T \Sigma^{-1} \Delta\boldsymbol{\mu}$$

The η_i can therefore be concisely written as:

$$\eta_1 = \frac{1}{4} d^2 = -\eta_0$$

We can do similarly for the variance ς^2:

$$\begin{aligned}\varsigma^2 &= \left[(2\Sigma)^{-1}\Delta\boldsymbol{\mu}\right]^T \Sigma(2\Sigma)^{-1}\Delta\boldsymbol{\mu} \\ &= \frac{1}{4}\Delta\boldsymbol{\mu}^T(\Sigma)^{-1}\Sigma\Sigma^{-1}\Delta\boldsymbol{\mu} \\ &= \frac{1}{4}\Delta\boldsymbol{\mu}^T\Sigma^{-1}\Delta\boldsymbol{\mu} \\ &= \frac{1}{4}d^2\end{aligned}$$

A.5 Derivative of Error

We can analyse the extrema of the error function as a function of correlation by taking its first derivative through the chain rule and determining when it is equal to zero:

$$\begin{aligned}\frac{d\varepsilon}{d\rho} &= \frac{d}{dp}\left(\frac{1}{2}erfc\left(\frac{1}{2\sqrt{2}}\sqrt{d^2}\right)\right) = \frac{1}{2}\frac{d}{dp}\left(erfc\left(z\right)\right) \\ &= \frac{1}{2}\frac{d}{dz}\left(erfc\left(z\right)\right)\frac{dz}{dd^2}\frac{dd^2}{d\rho}\end{aligned} \tag{42}$$

With:

$$d^2 = \frac{1}{1-\rho^2}\left[r_x^2 + r_y^2 - 2\rho r_x r_y\right] \tag{43}$$

We start with the first derivative:

$$\begin{aligned}\frac{d}{dz}\left(erfc\left(z\right)\right) &= \frac{-2e^{-z^2}}{\sqrt{\pi}} \\ &= \frac{-2e^{-\left(\frac{1}{2\sqrt{2}}\sqrt{d^2}\right)^2}}{\sqrt{\pi}} \\ &= \frac{-2e^{-\frac{1}{8}d^2}}{\sqrt{\pi}}\end{aligned} \tag{44}$$

Then the second derivative:

$$\begin{aligned}\frac{dz}{dd^2} &= \frac{d}{dd^2}\frac{1}{2\sqrt{2}}\sqrt{d^2} \\ &= \frac{1}{4\sqrt{2d^2}}\end{aligned} \tag{45}$$

The third derivative:

$$\begin{aligned}\frac{dd^2}{d\rho} &= \frac{d}{d\rho}\left(\frac{1}{1-\rho^2}\left[r_x^2 + r_y^2 - 2\rho r_x r_y\right]\right) \\ &= \left[r_x^2 + r_y^2 - 2\rho r_x r_y\right]\frac{d}{d\rho}\frac{1}{1-\rho^2} + \frac{1}{1-\rho^2}\frac{d}{d\rho}\left[r_x^2 + r_y^2 - 2\rho r_x r_y\right] \\ &= \left[r_x^2 + r_y^2 - 2\rho r_x r_y\right]\frac{2\rho}{\left(1-\rho^2\right)^2} + \frac{1}{1-\rho^2}\left[-2r_x r_y\right] \\ &= \frac{1}{\left(1-\rho^2\right)^2}\left[\left(r_x^2 + r_y^2 - 2\rho r_x r_y\right)2\rho + \left(1-\rho^2\right)\left(-2r_x r_y\right)\right] \\ &= \frac{1}{\left(1-\rho^2\right)^2}\left[\left(2\rho r_x^2 + 2\rho r_y^2 - 2\rho 2\rho r_x r_y\right) + \left(-2r_x r_y + 2r_x r_y \rho^2\right)\right] \\ &= \frac{-2}{\left(1-\rho^2\right)^2}\left[\rho^2 r_x r_y - \rho\left(r_x^2 + r_y^2\right) + r_x r_y\right]\end{aligned} \tag{46}$$

A.6 Extrema of Error

We can evaluate the extrema by finding the points where equations 44, 45, 46 are equal to 0:

$$\begin{aligned}0 = \frac{d}{dz}\left(erfc\left(z\right)\right) \Leftrightarrow 0 &= \frac{-2e^{-\frac{1}{8}d^2}}{\sqrt{\pi}} \\ d^2 &\to \infty\end{aligned} \tag{47}$$

We assume the ratios r_x, r_y to be finite and the euclidean distance between the distribution means to be finite and non-null. In other words, if $d^2 \to \infty$ it is

exclusively due to the correlation coefficient. Given the definition of d^2 then:

$$d^2 \to \infty \Leftrightarrow |\rho| \to 1 \tag{48}$$

We proceed similarly for the second derivative (equation 45):

$$\begin{aligned} 0 = \frac{dz}{dd^2} \Leftrightarrow 0 &= \frac{1}{4\sqrt{2d^2}} \\ d^2 \to \infty &\Leftrightarrow |\rho| \to 1 \end{aligned} \tag{49}$$

Finally, for the third derivative (equation 46):

$$\begin{aligned} 0 = \frac{dd^2}{d\rho} \Leftrightarrow 0 &= \frac{-2}{\left(1-\rho^2\right)^2}\left[\rho^2 r_x r_y - \rho\left(r_x^2 + r_y^2\right) + r_x r_y\right] \\ \Leftrightarrow 0 &= \rho^2 r_x r_y - \rho\left(r_x^2 + r_y^2\right) + r_x r_y \end{aligned} \tag{50}$$

Two cases naturally arise from network parameters. First, if one of the ratios $r_u = 0$. This would happen if the mean activity of a unit is equal across inputs. If the mean activity of both units remained unchanged, the resulting multivariate distributions would overlap which breaks the basic assumptions justifiying the choice of LDA. In this first case then:

$$0 = \frac{dd^2}{d\rho} \Leftarrow 0 = \rho \quad if \;\; r_x = 0 \quad or \;\; r_y = 0 \tag{51}$$

The second case happens when neither ration r_x, r_y is null:

$$
\begin{aligned}
0 = \frac{dd^2}{d\rho} \Leftarrow 0 &= \rho^2 - \rho\frac{r_x^2 + r_y^2}{r_x r_y} + 1 \\
&\Leftrightarrow \rho = \frac{\frac{r_x^2+r_y^2}{r_x r_y} \pm \sqrt{\frac{\left(r_x^2+r_y^2\right)^2}{r_x^2 r_y^2} - 4}}{2} \\
&= \frac{r_x^2 + r_y^2 \pm \sqrt{\left(r_x^2 + r_y^2\right)^2 - 4r_x^2 r_y^2}}{2r_x r_y} \\
&= \frac{r_x^2 + r_y^2 \pm \sqrt{r_x^4 + r_y^4 - 2r_x^2 r_y^2}}{2r_x r_y} \\
&= \frac{r_x^2 + r_y^2 \pm \sqrt{\left(r_x^2 - r_y^2\right)^2}}{2r_x r_y} \\
&= \frac{r_x^2 + r_y^2 \pm |r_x^2 - r_y^2|}{2r_x r_y} \\
&= \frac{r_x^2 + r_y^2 \pm \left[max\left(r_x^2, r_y^2\right) - min\left(r_x^2, r_y^2\right)\right]}{2r_x r_y}
\end{aligned}
$$

For simplicity, we can again split the last equation in three cases. The first, when $r_x \to r_y$:

$$\rho \to \frac{r_y^2 + r_y^2}{2r_y r_y} \to 1 \tag{52}$$

The second, when $r_x \to -r_y$:

$$\rho \to \frac{r_y^2 + r_y^2}{-2r_y r_y} \to -1 \tag{53}$$

To evaluate the more general case, *i. e.* $r_x \neq r_y$ it is simpler to look separately

at the positive and negative roots. First, the positive one:

$$\begin{aligned}\rho_+ &= \frac{r_x^2 + r_y^2 + max\left(r_x^2, r_y^2\right) - min\left(r_x^2, r_y^2\right)}{2r_x r_y} \\ &= \frac{max\left(r_x^2, r_y^2\right)}{r_x r_y}\end{aligned} \tag{54}$$

However, $|max\left(r_x^2, r_y^2\right)| > |r_x r_y|$ from the assumption that one ratio is smaller than the other (or unequal, non-null). This means that $|\rho_+| > 1 \forall r_x, r_y$. Since the correlation is bound in the range $]-1, 1[$, the positive root must be rejected. The negative root on the other hand does not suffer from the same problem:

$$\begin{aligned}\rho_- &= \frac{r_x^2 + r_y^2 - max\left(r_x^2, r_y^2\right) + min\left(r_x^2, r_y^2\right)}{2r_x r_y} \\ &= \frac{min\left(r_x^2, r_y^2\right)}{r_x r_y}\end{aligned} \tag{55}$$

The summed up conditions for extrema are therefore:

	Condition	Position of Extrema
1.	Irrespective of network parameters	$\rho \to 1$
2.	$r_x \to r_y$	$\rho \to 1$
3.	$r_x = 0$ or $r_y = 0$	$\rho = 0$
4.	$r_x \neq 0$ and $r_y \neq 0$	$\rho = \frac{min\left(r_x^2, r_y^2\right)}{r_x r_y}$
5.	$r_x \to -r_y$	$\rho \to -1$
6.	Irrespective of network parameters	$\rho \to -1$

A.7 Minima and Maxima

Calculating the second derivative of the error function would be lengthy considering that each of the first derivatives calculated previously is a function of

ρ. Instead, we can determine upwards and downwards trends of the error curve by determinng the sign of the derivative between the three potential maxima (considering two of them are mutually exclusively). We take equations 44-46 and introduce them to equation 42.

$$\begin{aligned}
\frac{d\varepsilon}{d\rho} &= \frac{1}{2}\frac{d}{dz}\left(erfc\left(z\right)\right)\frac{dz}{dd^2}\frac{dd^2}{d\rho} \\
&= \frac{1}{2}\frac{-2e^{-\frac{1}{8}d^2}}{\sqrt{\pi}}\frac{1}{4\sqrt{2d^2}}\frac{-2}{\left(1-\rho^2\right)^2}\left[\rho^2 r_x r_y - \rho\left(r_x^2+r_y^2\right)+r_x r_y\right] \\
\Leftrightarrow sign\left(\frac{d\varepsilon}{d\rho}\right) &= sign\left(\rho^2 r_x r_y - \rho\left(r_x^2+r_y^2\right)+r_x r_y\right) \qquad (56)
\end{aligned}$$

We first consider condition (2) for extrema assuming that $r_x = 0$ (for this part of the analysis, the choice is arbitrary and makes no difference):

$$\begin{aligned}
\Leftrightarrow sign\left(\frac{d\varepsilon}{d\rho}\right) &= sign\left(\rho^2 r_x r_y - \rho\left(r_x^2+r_y^2\right)+r_x r_y\right) \\
&= sign\left(-\rho r_y^2\right) \\
&= -sign\left(\rho\right) \qquad (57)
\end{aligned}$$

We then consider condition (3) for extrema. In this case, we have already found the zeros of $\rho^2 r_x r_y - \rho\left(r_x^2+r_y^2\right)+r_x r_y$, *i.e.* ρ_- and ρ_+. Then to determine $sign\left(\frac{d\varepsilon}{d\rho}\right)$ we simply need to know if the extremum of the parabola is a minimum or a maximum:

$$\frac{d^2}{d\rho^2}\left(\rho^2 r_x r_y - \rho\left(r_x^2+r_y^2\right)+r_x r_y\right) = 2 > 0$$

Since $\rho_+ > 1$ this means that:

$$\frac{d\varepsilon}{d\rho} > 0 \Leftrightarrow p \in [-1, \rho_-] \tag{58}$$

$$\frac{d\varepsilon}{d\rho} < 0 \Leftrightarrow p \in [\rho_-, 1] \tag{59}$$

Together, from equations 57, 58, 59, we can see that for all cases, the error curve as a function of correlation increases from $\rho = -1$ until its maximum, found at a value $\rho_* = 0$ or $\rho_* = \rho_-$ and then decreases until $\rho = 1$.

A.8 Translations of maximum error

The conditions for ρ_* to happen is that no ratio r_u is equal to zero and that $r_x^2 \neq r_y^2$. From here, we can look at how we can control the value of ρ_* and d^2 to generate precise translations from an initial state 1 to a modified state, indexed with 2. These indexes are not to be confused with the indexes for the different input, since the ratios already encompass both inputs ν_1 and ν_2. From these definitions, this is equivalent to say that we want to find the values of a, b, m, n such that:

$$\rho_2 = m\rho_1$$

$$d_2^2 = nd_1^2$$

$$r_{x2} = ar_{x1}$$

$$r_{y2} = br_{y1}$$

With:

$$\rho_i = \frac{min\left(r_{xi}^2, r_{yi}^2\right)}{r_{xi}r_{yi}}$$

$$d_i^2 = \frac{1}{1-\rho_i^2}\left[r_{xi}^2 + r_{yi}^2 - 2r_{xi}r_{yi}\rho_i\right]$$

A.8.1 Correlation

$$
\begin{aligned}
m\rho_1 &= \rho_2 \\
m\frac{min\left(r_{x1}^2, r_{y1}^2\right)}{r_{x1}r_{y1}} &= \frac{min\left(r_{x2}^2, r_{y2}^2\right)}{r_{x2}r_{y2}}
\end{aligned}
$$

Case 1, $r_{x1}^2 < r_{y1}^2$, $r_{x2}^2 < r_{y2}^2$:

$$
\begin{aligned}
m\frac{r_{x1}^2}{r_{x1}r_{y1}} &= \frac{r_{x2}^2}{r_{x2}r_{y2}} \\
m\frac{r_{x1}}{r_{y1}} &= \frac{r_{x2}}{r_{y2}} \\
m\frac{r_{x1}}{r_{y1}} &= \frac{ar_{x1}}{br_{y1}} \\
m &= \frac{a}{b}
\end{aligned} \tag{60}
$$

Case 2, $r_{x1}^2 > r_{y1}^2$, $r_{x2}^2 > r_{y2}^2$:

$$
\begin{aligned}
m\frac{r_{y1}^2}{r_{x1}r_{y1}} &= \frac{r_{y2}^2}{r_{x2}r_{y2}} \\
m\frac{r_{y1}}{r_{x1}} &= \frac{r_{y2}}{r_{x2}} \\
m\frac{r_{y1}}{r_{x1}} &= \frac{br_{y1}}{ar_{x1}} \\
m &= \frac{b}{a}
\end{aligned} \tag{61}
$$

Case 3, $r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$:

$$\begin{aligned} m\frac{r_{x1}^2}{r_{x1}r_{y1}} &= \frac{r_{y2}^2}{r_{x2}r_{y2}} \\ m\frac{r_{x1}}{r_{y1}} &= \frac{r_{y2}}{r_{x2}} \\ m\frac{r_{x1}}{r_{y1}} &= \frac{br_{y1}}{ar_{x1}} \\ \frac{r_{x1}^2}{r_{y1}^2} &= \frac{b}{ma} \end{aligned} \tag{62}$$

Case 4, $r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$:

$$\begin{aligned} m\frac{r_{y1}^2}{r_{x1}r_{y1}} &= \frac{r_{x2}^2}{r_{x2}r_{y2}} \\ m\frac{r_{y1}}{r_{x1}} &= \frac{r_{x2}}{r_{y2}} \\ m\frac{r_{y1}}{r_{x1}} &= \frac{ar_{x1}}{br_{y1}} \\ \frac{r_{y1}^2}{r_{x1}^2} &= \frac{a}{mb} \end{aligned} \tag{63}$$

A.8.2 Distance

$$
\begin{aligned}
d_1^2 &= \frac{1}{1-\rho_1^2}\left[r_{x1}^2 + r_{y1}^2 - 2r_{x1}r_{y1}\rho_1\right] \\
&= \frac{1}{1-\rho_1^2}\left[r_{x1}^2 + r_{y1}^2 - 2r_{x1}r_{y1}\frac{min\left(r_{x1}^2, r_{y1}^2\right)}{r_{x1}r_{y1}}\right] \\
&= \frac{1}{1-\rho_1^2}\left[r_{x1}^2 + r_{y1}^2 - 2min\left(r_{x1}^2, r_{y1}^2\right)\right] \\
&= \frac{1}{1-\rho_1^2}\left[max\left(r_{x1}^2 + r_{y1}^2 - 2r_{x1}^2, r_{x1}^2 + r_{y1}^2 - 2r_{y1}^2\right)\right] \\
&= \frac{max\left(r_{y1}^2 - r_{x1}^2, r_{x1}^2 - r_{y1}^2\right)}{1-\rho_1^2} \\
&= \frac{max\left(r_{y1}^2 - r_{x1}^2, r_{x1}^2 - r_{y1}^2\right)}{1-\left(\frac{min\left(r_{x1}^2, r_{y1}^2\right)}{r_{x1}r_{y1}}\right)^2} \\
&= \frac{max\left(r_{y1}^2 - r_{x1}^2, r_{x1}^2 - r_{y1}^2\right)}{1-\frac{min\left(r_{x1}^4, r_{y1}^4\right)}{r_{x1}^2 r_{y1}^2}} \\
&= \frac{max\left(r_{y1}^2 - r_{x1}^2, r_{x1}^2 - r_{y1}^2\right)}{1-min\left(\frac{r_{x1}^2}{r_{y1}^2}, \frac{r_{y1}^2}{r_{x1}^2}\right)} \\
&= \frac{max\left(r_{y1}^2 - r_{x1}^2, r_{x1}^2 - r_{y1}^2\right)}{max\left(1-\frac{r_{x1}^2}{r_{y1}^2}, 1-\frac{r_{y1}^2}{r_{x1}^2}\right)} \\
&= \frac{max\left(r_{y1}^2 - r_{x1}^2, r_{x1}^2 - r_{y1}^2\right)}{max\left(\frac{1}{r_{y1}^2}\left[r_{y1}^2 - r_{x1}^2\right], \frac{1}{r_{x1}^2}\left[r_{x1}^2 - r_{y1}^2\right]\right)}
\end{aligned}
$$

As before, we separate it in 4 cases. Case 1, $r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$:

$$d_1^2 = d_2^2 \tag{64}$$

$$\frac{r_{y1}^2 - r_{x1}^2}{\frac{1}{r_{y1}^2}\left[r_{y1}^2 - r_{x1}^2\right]} = \frac{r_{y2}^2 - r_{x2}^2}{\frac{1}{r_{y2}^2}\left[r_{y2}^2 - r_{x2}^2\right]}$$

$$r_{y1}^2 = r_{y2}^2$$

$$r_{y1}^2 = b^2 r_{y1}^2$$

$$|b| = 1 \tag{65}$$

Case 2, $r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$:

$$d_1^2 = d_2^2 \tag{66}$$

$$\frac{r_{x1}^2 - r_{y1}^2}{\frac{1}{r_{x1}^2}\left[r_{x1}^2 - r_{y1}^2\right]} = \frac{r_{x2}^2 - r_{y2}^2}{\frac{1}{r_{x2}^2}\left[r_{x2}^2 - r_{y2}^2\right]}$$

$$r_{x1}^2 = r_{x2}^2$$

$$r_{x1}^2 = a^2 r_{x1}^2$$

$$|a| = 1 \tag{67}$$

Case 3, $r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$:

$$d_1^2 = d_2^2 \tag{68}$$

$$\frac{r_{y1}^2 - r_{x1}^2}{\frac{1}{r_{y1}^2}\left[r_{y1}^2 - r_{x1}^2\right]} = \frac{r_{x2}^2 - r_{y2}^2}{\frac{1}{r_{x2}^2}\left[r_{x2}^2 - r_{y2}^2\right]}$$

$$r_{y1}^2 = r_{x2}^2$$

$$r_{y1}^2 = a^2 r_{x1}^2$$

$$\frac{r_{x1}^2}{r_{y1}^2} = \frac{1}{a^2} \tag{69}$$

$$\tag{70}$$

Case 4, $r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$:

$$d_1^2 = d_2^2 \tag{71}$$

$$\frac{r_{x1}^2 - r_{y1}^2}{\frac{1}{r_{x1}^2}\left[r_{x1}^2 - r_{y1}^2\right]} = \frac{r_{y2}^2 - r_{x2}^2}{\frac{1}{r_{y2}^2}\left[r_{y2}^2 - r_{x2}^2\right]}$$

$$r_{x1}^2 = r_{y2}^2$$

$$r_{x1}^2 = b^2 r_{y1}^2$$

$$\frac{r_{y1}^2}{r_{x1}^2} = \frac{1}{b^2} \tag{72}$$

A.8.3 Correlation and Distance together

We can now unify the conditions for each case. Case 1, $r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$:

$$m = \frac{a}{b},\ \ |b| = 1 \tag{73}$$

Case 2, $r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$:

$$m = \frac{b}{a},\ \ |a| = 1 \tag{74}$$

Case 3, $r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$:

$$\frac{r_{x1}^2}{r_{y1}^2} = \frac{b}{ma},\quad \frac{r_{x1}^2}{r_{y1}^2} = \frac{1}{a^2}$$

$$\frac{b}{ma} = \frac{1}{a^2}$$

$$ab = m \tag{75}$$

Case 4, $r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$:

$$\frac{r_{y1}^2}{r_{x1}^2} = \frac{a}{mb},\ \ \frac{r_{y1}^2}{r_{x1}^2} = \frac{1}{b^2}$$
$$\frac{a}{mb} = \frac{1}{b^2}$$
$$ab = m \tag{76}$$

Table 4: Conditions for translations of error maximum

	$r_{x1}^2 < r_{y1}^2$	$r_{x1}^2 > r_{y1}^2$
$r_{x2}^2 < r_{y2}^2$	$m = \frac{a}{b},\ \ \lvert b\rvert = 1$	$ab = m$
$r_{x2}^2 > r_{y2}^2$	$ab = m$	$m = \frac{b}{a},\ \ \lvert a\rvert = 1$

A.8.4 Putting it together

We start with a known r_{x1}, r_{y1} and m, we want to know the values of r_{x2}, r_{y2} with as much control on a and b as possible.

Table 5: General parameter relationships for translations of error maximum

Case	r_{x2}	r_{y2}	m	Extra conditions
$r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$	ar_{x1}	br_{y1}	$m = \frac{a}{b}$	$\lvert b\rvert = 1$
$r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$	ar_{x1}	br_{y1}	$m = \frac{b}{a}$	$\lvert a\rvert = 1$
$r_{x1}^2 < r_{y1}^2,\ \ r_{x2}^2 > r_{y2}^2$	$\lvert r_{x2}\rvert = \lvert r_{y1}\rvert$	br_{y1}	$m = ab$	$\frac{r_{x1}^2}{r_{y1}^2} = \frac{1}{a^2}$
$r_{x1}^2 > r_{y1}^2,\ \ r_{x2}^2 < r_{y2}^2$	ar_{x1}	$\lvert r_{y2}\rvert = \lvert r_{x1}\rvert$	$m = ab$	$\frac{r_{y1}^2}{r_{x1}^2} = \frac{1}{b^2}$