



Linearly decodable functions from neural population codes

M. Brandon Westover^{a,*}, Chris Eliasmith^b, Charles H. Anderson^a

^a*Department of Anatomy and Neurobiology, Washington University, School of Medicine, St. Louis, MO 63110, USA*

^b*Department of Philosophy, University of Waterloo, Waterloo, Canada, ON N2L 3G1*

Abstract

The population vector is a linear *decoder* for an ensemble of neurons, whose response properties are nonlinear functions of the input vector. However, previous analyses of this decoder seem to have missed the observation that the population vector can also be used to estimate *functions* of the input vector. We explore the use of singular value decomposition to delineate the set of functions which are linearly decodable from a given population of noisy neurons. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Population codes; Singular value decomposition; Principal components

Many sensory systems utilize a large number of neurons to make measurements on inherently low-dimensional systems. Examples include the ~ 1000 neurons in the cricket cercal system that measure horizontal wind velocity [2]; and the much larger hair cell population in the mammalian otolith that senses linear acceleration of the head, a three-dimensional vector [1]. Although the responses of the individual neurons are highly nonlinear and heterogeneous, a linear weighted sum of these nonlinear measures often provides an exceptionally precise linear estimate of the underlying physical input. Linear or “population vector” decoding is typically used to explore how neural systems might implement a communication channel; that is, how downstream populations might *reconstruct* the value of an encoded physical variable by specific synaptic weighting and summing of afferent sensory neuronal signals [4]. However, many downstream populations may be more concerned with extracting *transformations* of the sensory

* Corresponding author.

E-mail addresses: westovem@medicine.wustl.edu (M.B. Westover), eliasmith@uwaterloo.ca (C. Eliasmith), cha@shifter.wustl.edu (C.H. Anderson).

input. Using the general framework we present below, such transformations can also be computed by linear decoding.

The approach taken here is to view the computation of the decoding vector as a linear combination of projections of the high-dimensional state vector of neuronal activities onto a *special* set of axes. Combinations of projections along other axes provide measures of nonlinear functions. Examples include the magnitude of the input vector, or bilinear forms, like the vector cross product. We discuss below how to determine the range of functions that can be robustly extracted in this manner.

Let \mathbf{x} be a D -dimensional random vector representing the input to a set of N neurons, with response curves $\{a_1(\mathbf{x}), a_2(\mathbf{x}), \dots, a_N(\mathbf{x})\}$. The response curves are a heterogeneous set of nonlinear, scalar-valued functions; and the input is physically restricted to a finite range R , e.g. $R = \{\mathbf{x} : \|\mathbf{x}\| \leq 1\}$. For simplicity, we assume that \mathbf{x} is uniformly distributed on R .

We specify a set of N orthonormal vectors $\{\hat{\mathbf{e}}_1, \hat{\mathbf{e}}_2, \dots, \hat{\mathbf{e}}_N\}$ (columns of the $N \times N$ identity matrix) to serve as a basis (axes) for the population's state space. In these coordinates, the state vector for a given input \mathbf{x} is

$$\mathbf{a}(\mathbf{x}) = a_1(\mathbf{x})\hat{\mathbf{e}}_1 + a_2(\mathbf{x})\hat{\mathbf{e}}_2 + \dots + a_N(\mathbf{x})\hat{\mathbf{e}}_N. \quad (1)$$

Thus, the neuronal population $\{a_i\}$ maps the range R onto a surface $\mathbf{a}(\mathbf{x})$ in the N -dimensional state space, parameterized by \mathbf{x} . Noise in the neural encoding process (e.g. jitter in spike times) introduces a cloud of uncertainty around each state point.

Next, let the desired transformation $f(\mathbf{x})$ be a vector in a generalized vector space \mathbf{F} with inner product

$$\langle f_1(\mathbf{x}), f_2(\mathbf{x}) \rangle = \int_R f_1(\mathbf{x})f_2(\mathbf{x}) \, d\mathbf{x}. \quad (2)$$

For example, we may take \mathbf{F} to be the set of continuous functions over R .

Finally, we specify how the desired transformation is to be extracted from the neural activities by defining a *decoding rule*

$$\hat{f}(x) = \sum_{i=1}^N a_i(\mathbf{x})\phi_i = \mathbf{a}'(\mathbf{x})\phi, \quad (3)$$

where $\phi = [\phi_1, \phi_2, \dots, \phi_N]'$ are the *decoding vectors*, or in the case of $D=1$, *decoding weights*. Thus, the extracted approximation to $f(\mathbf{x})$ will be a linear combination of the neural response functions. The set of all possible combinations defines a vector subspace $\hat{\mathbf{F}} \subset \mathbf{F}$. We seek an optimal projector ϕ for projecting points along $\mathbf{a}(\mathbf{x})$ from the state space onto $\hat{\mathbf{F}}$.

Having set the stage, the precise statement and solution of our problem becomes geometrically clear (see Fig. 1a):

Find the function $\hat{f}(\mathbf{x}) \in \hat{\mathbf{F}}$ which best approximates $f(\mathbf{x}) \in \mathbf{F}$; that is, the vector in $\hat{\mathbf{F}}$ which minimizes the norm of the approximation error $\varepsilon = f(\mathbf{x}) - \hat{f}(\mathbf{x})$.

According to the well-known Projection Theorem, a unique solution $\hat{f}_0(\mathbf{x})$ exists, and is simply the orthogonal projection of $f(\mathbf{x})$ onto $\hat{\mathbf{F}}$. Thus, the error will be orthogonal

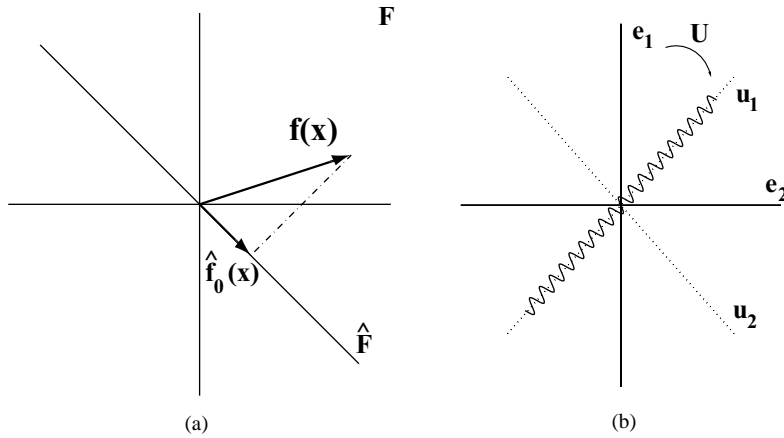


Fig. 1. (a) Projection theorem geometry. (b) State space rotation.

to the subspace spanned by the neural response functions; i.e.,

$$0 = \langle f(\mathbf{x}) - \hat{f}(\mathbf{x}), a_j(\mathbf{x}) \rangle = \left\langle f(\mathbf{x}) - \sum_{i=1}^N a_i(\mathbf{x}) \phi_i, a_j(\mathbf{x}) \right\rangle \quad (4)$$

for all $j = 1, \dots, N$. Solving, we obtain

$$\phi_i = \sum_{j=1}^N \Gamma_{ij}^{-1} \langle f(\mathbf{x}), a_j(\mathbf{x}) \rangle, \quad (5)$$

where

$$\Gamma_{ij} = \langle a_i(\mathbf{x}), a_j(\mathbf{x}) \rangle \quad (6)$$

is the gram matrix, whose entries represent the correlations between activities of neuron pairs, averaged over the input range R .

The gram matrix is usually ill-conditioned, so its inverse cannot be computed directly. This can be handled by adding statistically independent Gaussian noise to the neuronal responses [4]. Instead, we use the pseudoinverse based on singular value decomposition (SVD) [3, Chapter 2], because it reveals much about what functions can be extracted from a population's activities.

Using SVD, we decompose the gram matrix as

$$\Gamma = \mathbf{U}' \mathbf{S}' \mathbf{U}, \quad (7)$$

where $\mathbf{S} = \text{diag}\{\omega_1, \omega_2, \dots, \omega_N\}$ is the diagonal matrix of singular values in order of decreasing magnitude; and \mathbf{U} is a rotation matrix whose columns are the corresponding eigenvectors. The columns $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N\}$ of \mathbf{U} define a new coordinate system for the state space.

In rotated coordinates, the state vector $\mathbf{a}(\mathbf{x})$ and the orthogonal projector ϕ become $\mathbf{U}\mathbf{a}(\mathbf{x}) \doteq \chi(\mathbf{x}) = [\chi_1(\mathbf{x}), \chi_2(\mathbf{x}), \dots, \chi_N(\mathbf{x})]'$; and $\mathbf{U}\phi \doteq \Phi$. The function approximation can be reexpressed in rotated coordinates as

$$\hat{f}(\mathbf{x}) = \mathbf{a}'(\mathbf{x})\phi = \mathbf{a}'(\mathbf{x})\mathbf{U}'\mathbf{U}\phi = [\mathbf{U}\mathbf{a}(\mathbf{x})]'[\mathbf{U}\phi] = \chi'(\mathbf{x})\Phi = \sum_{i=1}^N \chi_i(\mathbf{x})\Phi_i. \quad (8)$$

Each of the curves $\chi_i(\mathbf{x})$ represents the projection of the rotated state path $\chi(\mathbf{x})$ onto a coordinate axis \mathbf{u}_i . As before, $\hat{f}(\mathbf{x})$ is a linear combination of functions spanning the subspace $\hat{\mathbf{F}}$.

Crucial differences distinguish the new and old representations. First, in the original coordinates, the ordering of axes $\hat{\mathbf{e}}_i$ is arbitrary. However, projections along the rotated axes \mathbf{u}_i are naturally ordered by the singular values of the gram matrix: Each singular value measures the second moment of the state path $\chi(\mathbf{x})$ about its associated axis. Second, the spanning set $\{\chi_i(\mathbf{x})\}$ is now *orthogonal*:

$$\langle \chi_i(x), \chi_j(x) \rangle = \omega_i \delta_{ij},$$

which means that the $\chi_i(\mathbf{x})$'s contribute *uncorrelated* information to the sum in (8), in decreasing order of importance from $i = 1$ to N .

Fig. 1b illustrates these points for a hypothetical case, where $D = 1$, $N = 2$. Before rotation, neither axis is indispensable to a representation of points on the curve $\mathbf{a}(x)$. After rotation, the projection along axis \mathbf{u}_1 clearly captures most of the information. Accordingly, truncating the expansion introduces only small changes in the error ε . In fact, noise in neuronal responses will destroy information along axes with small singular values.

By examining the first several χ functions (i.e., the robust projections of $\chi(\mathbf{x})$) we gain qualitative information about the nature and range of functions $\hat{\mathbf{F}}$ supported by the population.

We first examine a population with monotonic saturating response functions that encode a scalar x (see Fig. 2a). One such well studied example is the neural population that encodes horizontal eye position in the neural integrator [5].

In Fig. 2b we plot the five $\chi_i(\mathbf{x})$ functions corresponding to the highest singular values of the gram matrix. We see that this population supports an approximation basis which looks roughly like a “rotated” set of orthogonal polynomials over the interval $-1 \leq x \leq 1$.

As expected, linear transformations $f : x \rightarrow \alpha x$ are well supported by the population, in addition to functions well approximated by low order polynomials (Fig. 2c). We emphasize *low* order, because projections onto later axes are small and thus sensitive to neuronal noise. For this population, we observe an approximately exponential decrease in magnitude of the singular values with increasing polynomial order (not shown). The number of $\chi_i(x)$'s with significant singular values, hence, the range of transformations extractable from the population's neural activities, can be expanded by increasing the number of neurons or, perhaps more interestingly, by adjusting parameters governing the encoding of \mathbf{x} .

Note that this analysis fails if the nonlinear responses of the neurons in the population are *homogeneous*. Clearly, in this case, linear weighted sums will simply reflect the

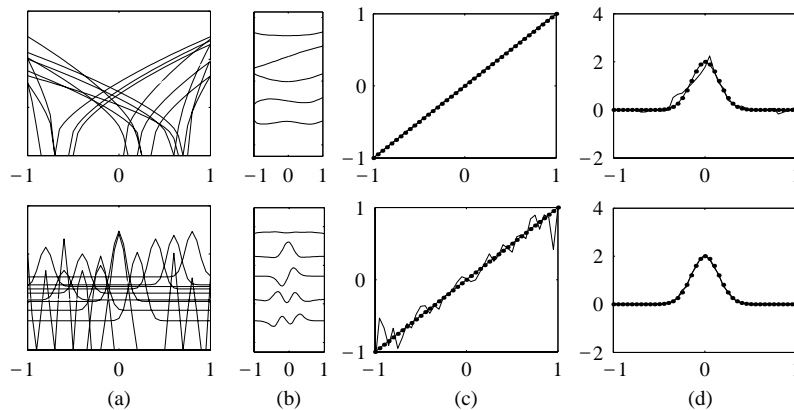


Fig. 2. Top: Monotonic, broadly tuned population. Bottom: Gaussian-tuned population. (a) Tuning curves, (b) first five $\chi_i(x)$'s, (c) decoding results for $f(x) = x$, (d) decoding results for a localized function $f(x)$. Dotted line is the desired function; solid line is the actual result. For (a), (c), and (d), $N = 20$. For (b), $N = 200$.

nonlinear responses of the individual neurons. This suggests that the diversity of real neuronal responses is computationally important.

Next, for comparison, we consider a population of neurons having Gaussian tuning curves (see Fig. 2a), modeled after ensembles frequently encountered in mammalian cortical populations. Again we plot the first five projections of $\chi(x)$ along the eigenvectors of this population's gram matrix.

Several striking differences stand out. First, note the absence of a linear term. Predictably, attempts to extract linear transformations from this population are less successful (Fig. 2c). However, the localized nature of these $\chi_i(x)$'s renders them a reasonable basis for approximating localized functions (Fig. 2d).

Thus, through the lens of the rotated state space, the differences between the response functions of these populations are seen to imply qualitatively different computational potentialities.

We have described how the population vector can be generalized to compute functions of input variables with linear decoding. The class of functions that can be decoded with significant robustness to noise depends on the form and diversity of neuronal responses in the population code. We are presently applying this observation to examine the properties of neuron populations in several primary sensory systems to obtain insight into the form of nonlinear transformations they can efficiently supply to downstream neurons.

References

- [1] D.E. Angelaki, M.Q. McHenry, J.D. Dickman, S.D. Newlands, B.J.M. Hess, Computation of inertial motion: neural strategies to resolve ambiguous otolith information, *J. Neurosci.* 19 (1999) 316–327.

- [2] G.A. Jacobs, F.E. Theunissen, Extraction of sensory parameters from a neural map by primary sensory interneurons, *J. Neurosci.* 20 (2000) 2934–2943.
- [3] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical recipes in C*, Cambridge University Press, New York, 1992.
- [4] E. Salinas, L.F. Abbott, Vector reconstruction from firing rates, *J. Comput. Neurosci.* 1 (1994) 89–107.
- [5] H.S. Seung, D.D. Lee, B.Y. Reis, D.W. Tank, Stability of the memory of eye position in a recurrent network of conductance-based model neurons, *Neuron* 26 (2000) 259–271.