Sparse Representations for the Cocktail Party Problem

Barak A. Pearlmutter,*1 Hiroki Asari,*2 and Anthony M. Zador*3

*1: Hamilton Institute, National University of Ireland Maynooth, Co. Kildare, Ireland. Ollscoil na hÉireann Mí Nuad, Éire.
*2: Cold Spring Harbor, Watson School of Biological Sciences, One Bungtown Road, Cold Spring Harbor, NY 11724, USA.
*3: Cold Spring Harbor, One Bungtown Road, Cold Spring Harbor, NY 11724, USA.

asari@cshl.edu http://zadorlab.cshl.edu/

Sparse Overcomplete Representation

Why does the cortex have so many neurons?

30,000 cochlear neurons

>100 million neurons in the auditory cortex

Intuition

Dense representation

Sparse representation

Problem formulation

Sparse Overcomplete Representation

Source Separation

Assumptions

• Sparse representations
• Prior knowledge of HRTF

Performance of different separation approaches

Context dependence of receptive field

Top-down receptive field modulation

Conclusions

• We propose that the cortex exploits the excess neural representations by selecting the sparsest representation within an overcomplete set of features.
• Sparseness can be used to separate sources perceived monaurally, by exploiting the differential filtering imposed by the HRTF.
• Our results support the idea that sparse representations may underlie efficient computations in the auditory cortex.

Predictions

Decoding is linear whereas encoding is nonlinear.

Context dependence of receptive field

Top-down receptive field modulation

Prediction Summary

• Neural representations should be sparse.
• Neural decoding is linear whereas neural encoding is nonlinear.
• Representations should be dynamically influenced by acoustic context (bottom-up).
• Representations should be dynamically modulated by top-down influences, including spatial expectation.

Cartoon of the model

Methods

Problem formulation

• Sound Source: \(x_i(t) = \sum_{j} c_{ij} u_j(t) \quad (i \neq j \text{: sparse})\)
• Coefficients: \(c_{ij}\)

Features

• Filter (Head-Related Transfer Function): \(f_j(t)\)
• Features: \(a_j(t) = f_j(t) * u_j(t)\)
• Filter (HRTF): \(f_j(t) = (HRTF)_j\)

Received signal:

\(y(t) = \sum_{i} a_i(t) * x_i(t) = \sum_{i} c_{ij} u_j(t)\)

Equations

\(x(t) = \sum_{i} a_i(t) * y(t) = \sum_{i} c_{ij} u_j(t)\)

L1-minimization:

\(\min \sum_{i} |c_{ij}| \text{ subject to } y(t) = \sum_{i} a_i(t)\)

L2-minimization (pseudo-inverse):

\(c_{ij} = (F^T F)^{-1} F^T y\)

NMF (non-negative matrix factorization)

Algorithm for factoring a data matrix under an elementwise non-negativity constraint

Barak A. Pearlmutter, Hiroki Asari, and Anthony M. Zador

Sparse Context 1

Sparse Context 2

STRF

Left HRTF

Back HRTF

Right HRTF

Expected azimuth: 180°

Expected azimuth: 200°

Normalized spike rate

Front HRTF

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)

Time (sec)

0 0.1 0.2 sec

Normalized spike rate

Time (sec)