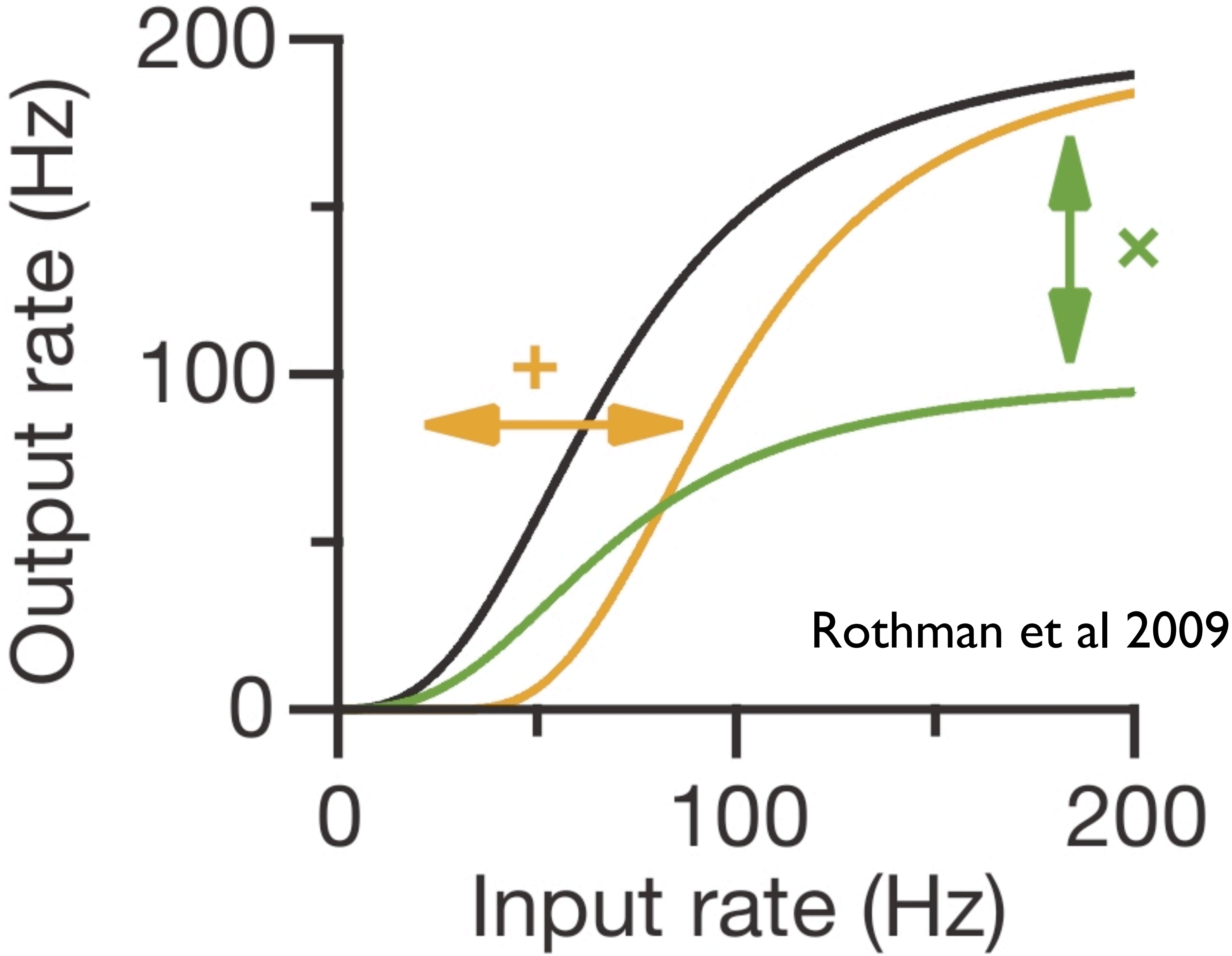
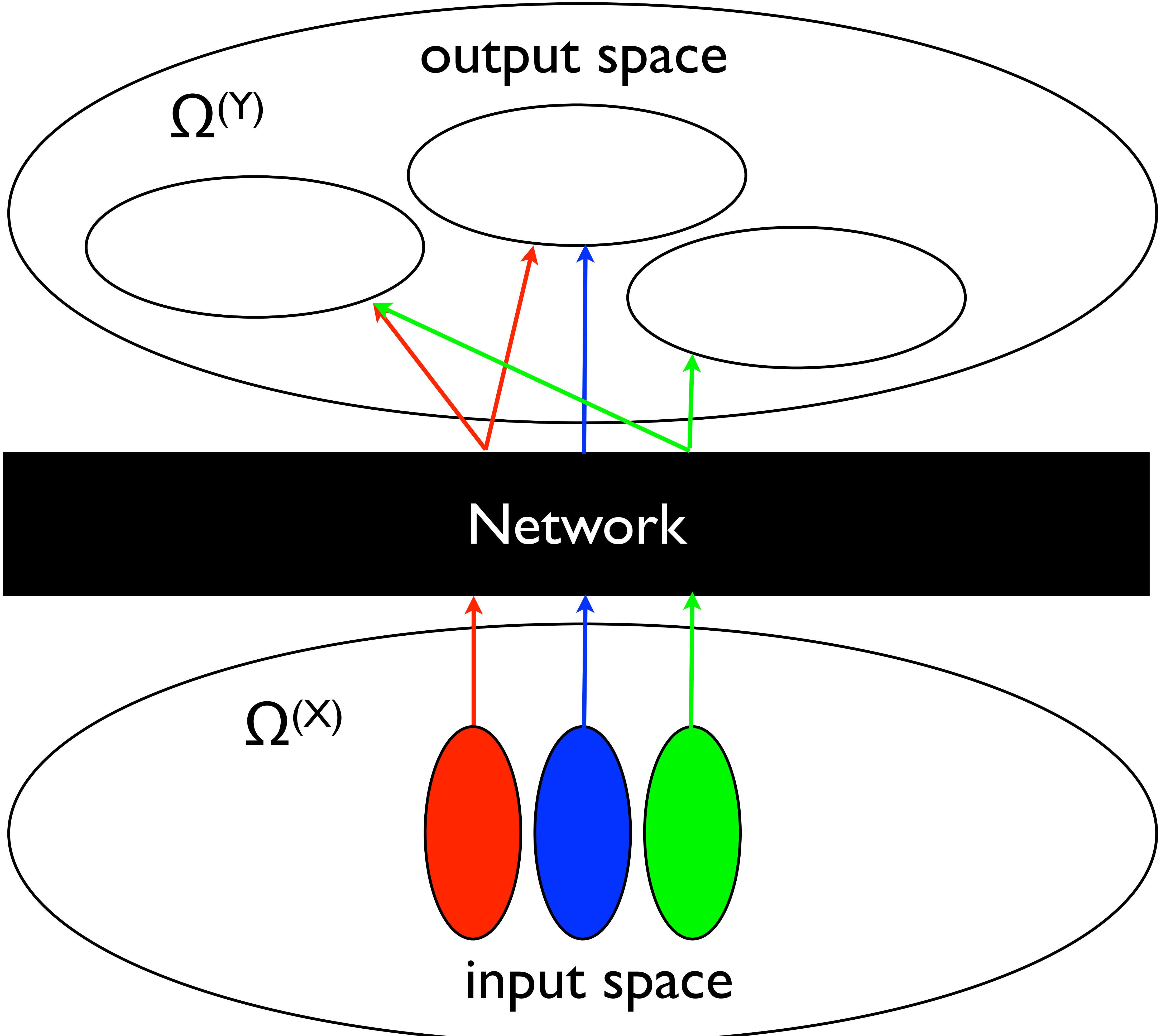
Network Transformations on Code Space

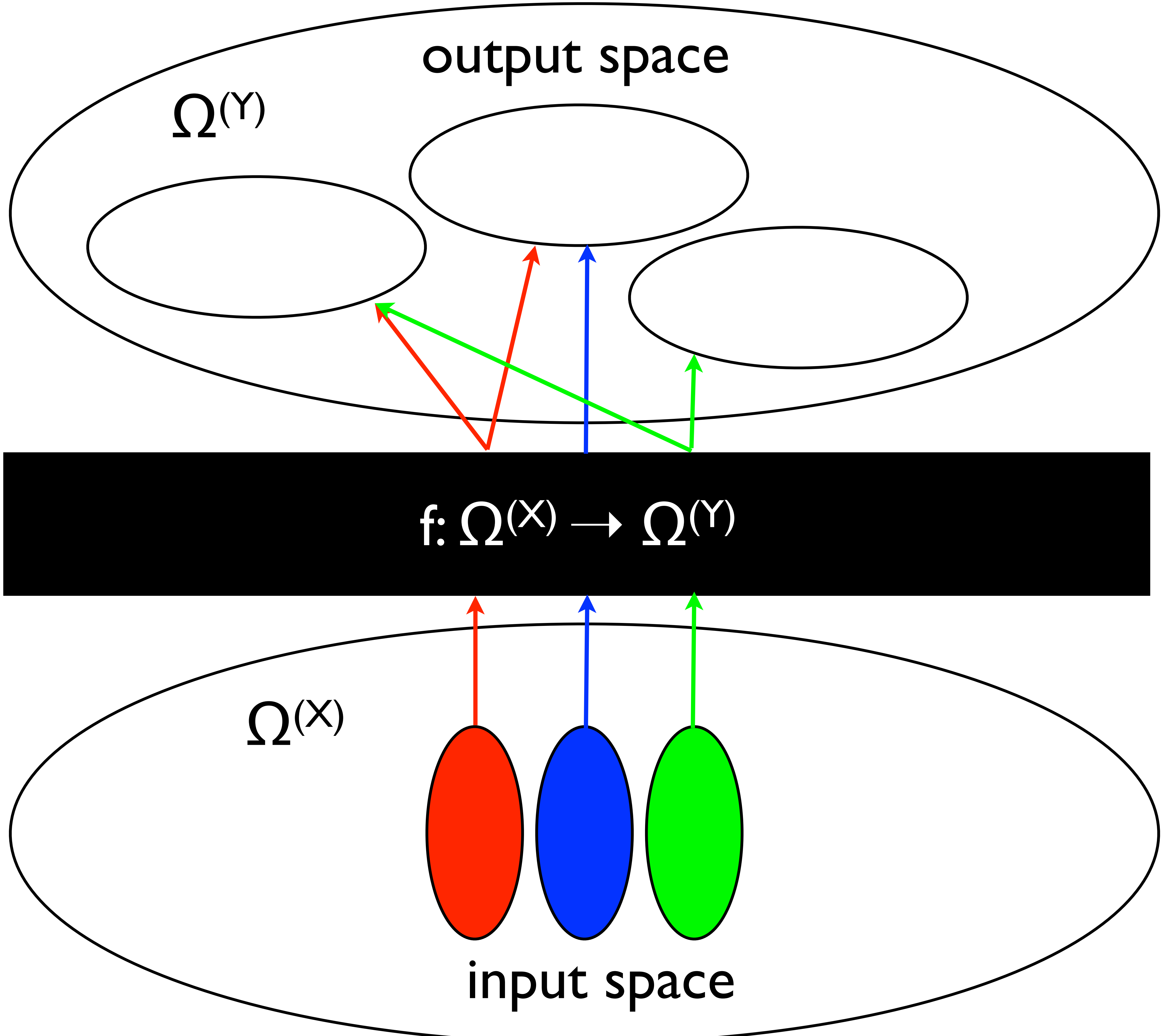
Guy Billings
Data Club 02/08/2010



200
100
0
Output rate (Hz)
+
×
Rothman et al 2009
0
100
200
Input rate (Hz)

y=f(x)


output space
Ω(Y)
Network
Ω(X)
input space


output space
$\Omega^{(Y)}$
$f: \Omega^{(X)} \rightarrow \Omega^{(Y)}$
$\Omega^{(X)}$
input space

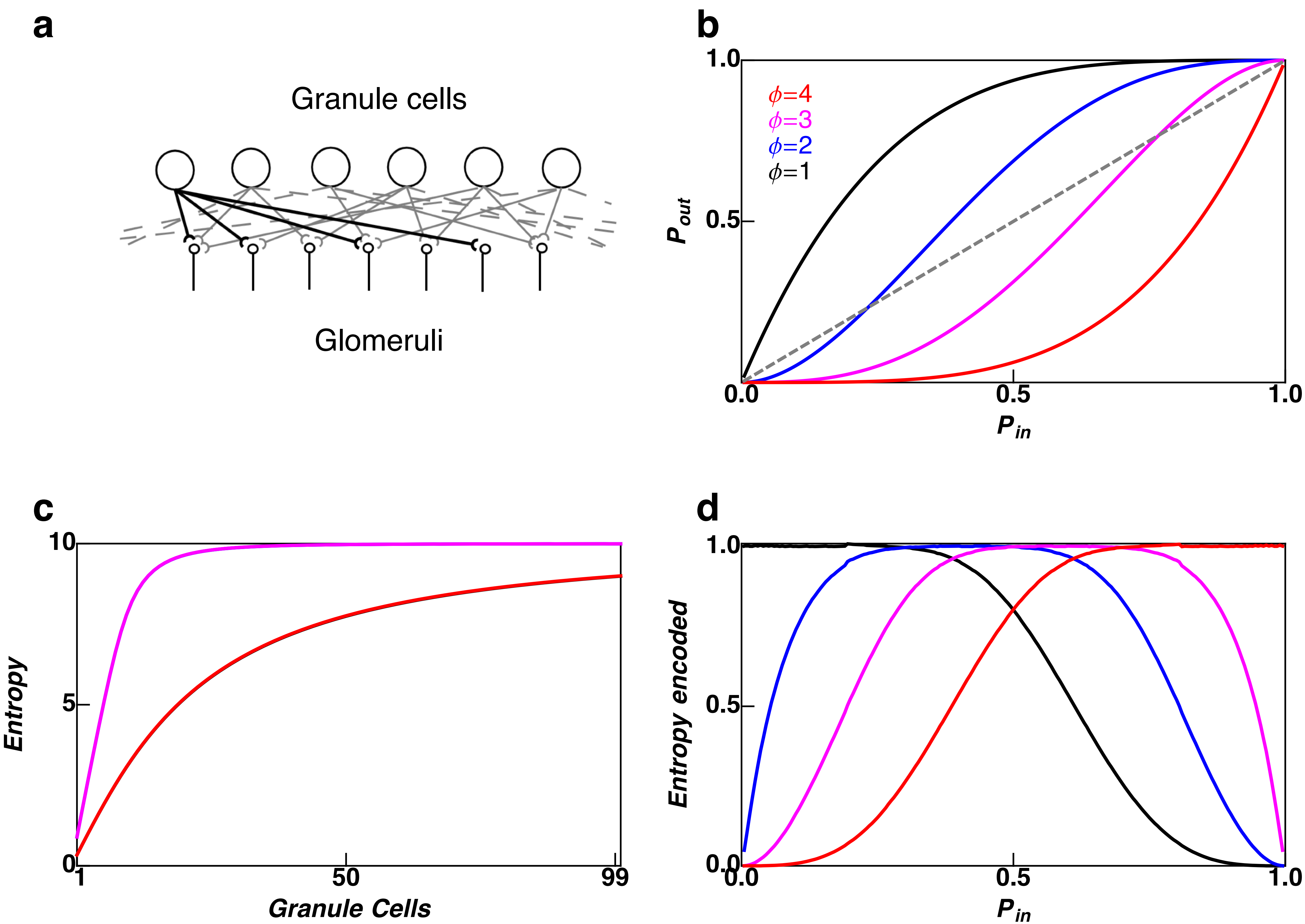
Introduction

Case study: Binary model of the granule cell layer

Spike train space

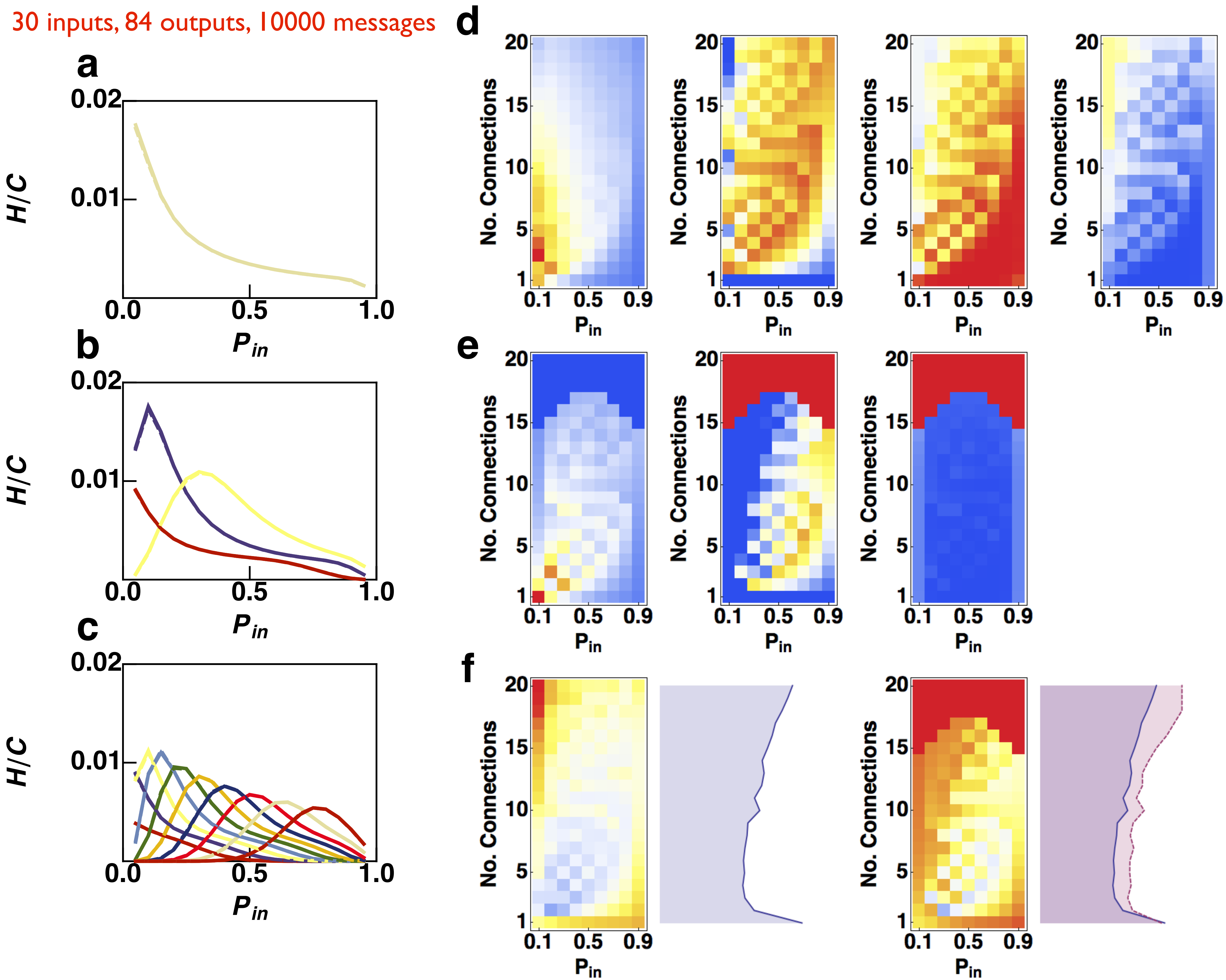
Hierarchical clustering

Simple model of the cerebellar granule cell layer

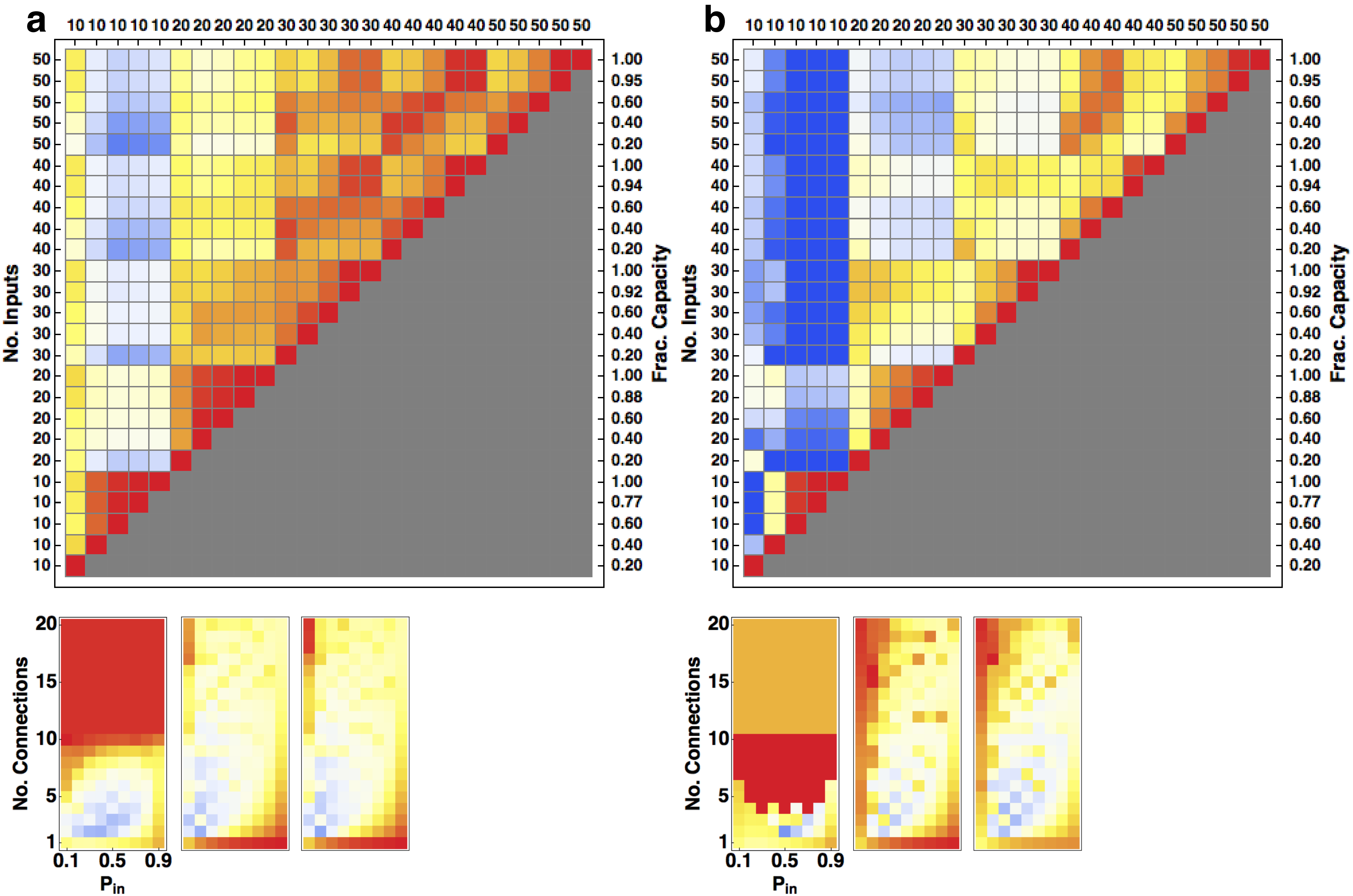


3 connections maximises coding efficiency

30 inputs, 84 outputs, 10000 messages



Scaling of encoding



Granule cell layer encoding

3 connections optimises energy efficiency of encoding

Overall, having a small number of connections (<10) minimises the performance distance between the network and the perfect network

Primary determinant of the nature of encoding (lossy/lossless, sparsity) is thresholding (tonic inhibition)

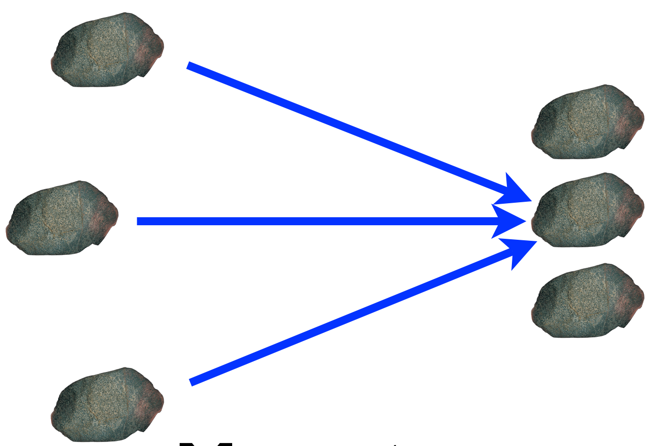
Further questions:

What is the optimal distribution?
Inclusion of metric pattern separation
Effect of correlations

NTCS perspective

$f: \Omega^{(X)} \rightarrow \Omega^{(Y)}$

Lossy



Many to one
Onto

Output space: Discrete topology
Define: pre-images of open sets as open
Input space is not separable
Metric space not possible (pseudo-metric space)
Efficient (compression?)

Lossless

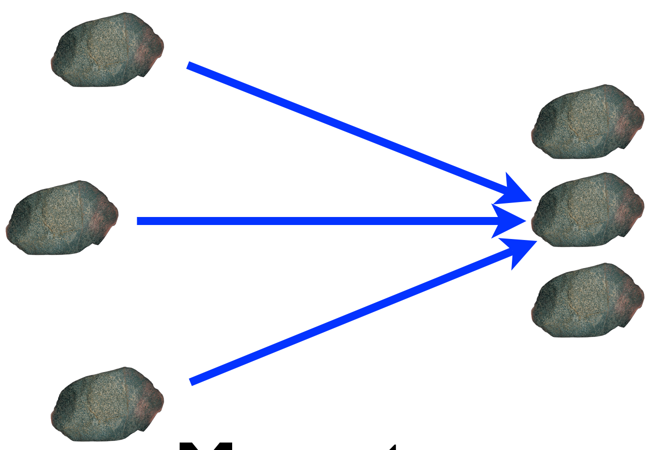
One to one
Not onto

Output space: Discrete topology
Define: pre-images of open sets as open
Input space is separable (has discrete topology)
Can be metrized
Costly (separation?)

NTCS perspective

$f: \Omega^{(X)} \rightarrow \Omega^{(Y)}$

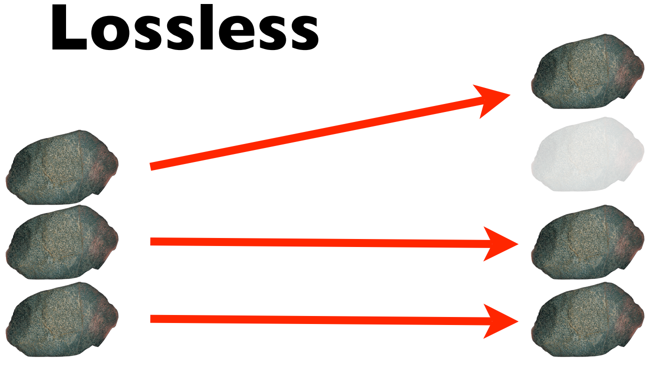
Lossy



Many to one
Onto

Output space: Discrete topology
Define: pre-images of open sets as open
Input space is not separable
Metric space not possible (pseudo-metric space)
Efficient (compression?)

Lossless



One to one
Not onto

Output space: Discrete topology
Define: pre-images of open sets as open
Input space is separable (has discrete topology)
Can be metrized
Costly (separation?)

In analogy to the transfer function: Statements about the transformation are restricted
Restrictions directly related to our **choice of code space** (domain)

The obvious choice of code space....

SPIKE TRAIN SPACE!

SPIKE TRAIN SPACE!

But this introduces extreme case of 'curse of dimensionality'

SPIKE TRAIN SPACE!

But this introduces extreme case of 'curse of dimensionality'

Need a way of controlling the complexity of the problem

Ways forward

Let $\Omega^{(X)}$ be 'spike train space'. Embed $\Omega^{(X)}$ in an 'ambient' space ($\mathbb{R}^n$).

1) Define a metric
   Embedded spike train space
   Clustering/dimensionality-reduction/other analysis based on chosen metric
   (Victor Purpura 1996, van Rossum 2001, Houghton 2010)

2) 'Connect the dots'
   Simplical complex
   Computational topology to find Homology group of the complex (i.e. its 'shape')
   (Carlsson 2008, Singh et al 2008)

Ways forward

Let $\Omega^{(X)}$ be 'spike train space'. Embed $\Omega^{(X)}$ in an 'ambient' space ($\mathbb{R}^n$).

1) Define a metric

Embedded spike train space

Clustering/dimensionality-reduction/other analysis based on chosen metric

(Victor Purpura 1996, Houghton 2010)

2) 'Connect the dots'

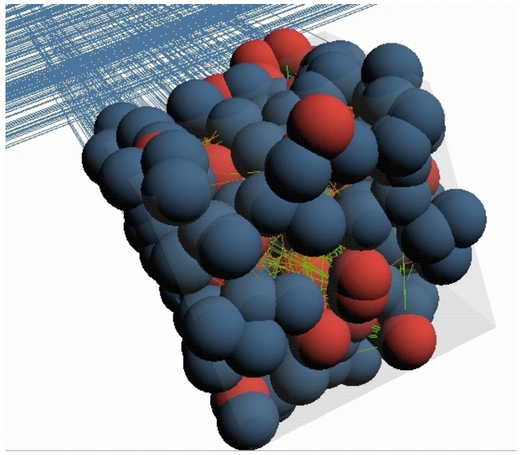
Simplical complex

Computational topology to find Homology group of the complex (i.e. its 'shape')

(Carlsson 2008, Singh et al 2008)

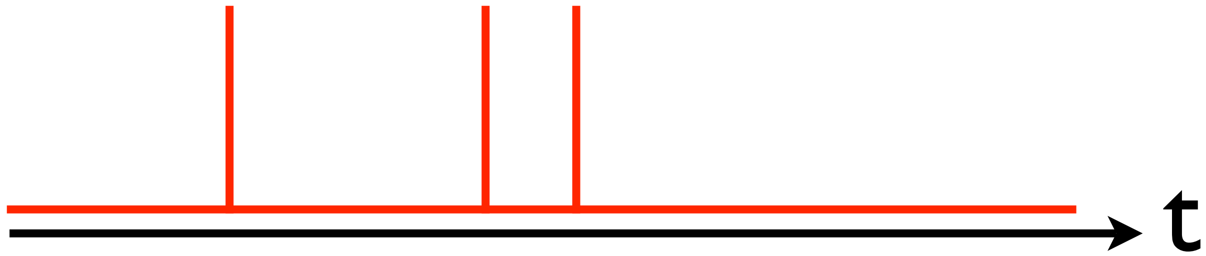
30 glomeruli, 84 granule cells

Input patterns: Inputs randomly activated (fire at 30Hz) with prob. 0.1
10 patterns, each with 10 repetitions
0Hz otherwise

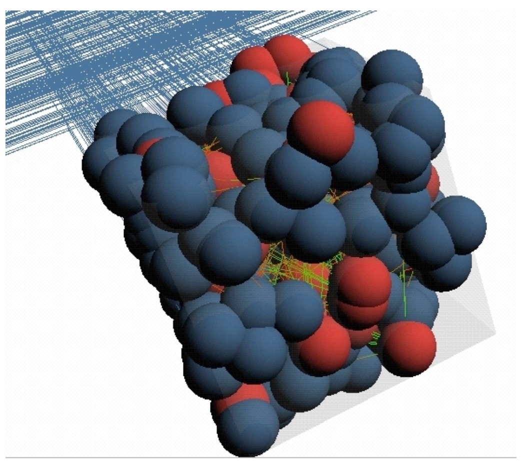


Output patterns: Resulting spike trains from granule cells
Weights adjusted to match experiment

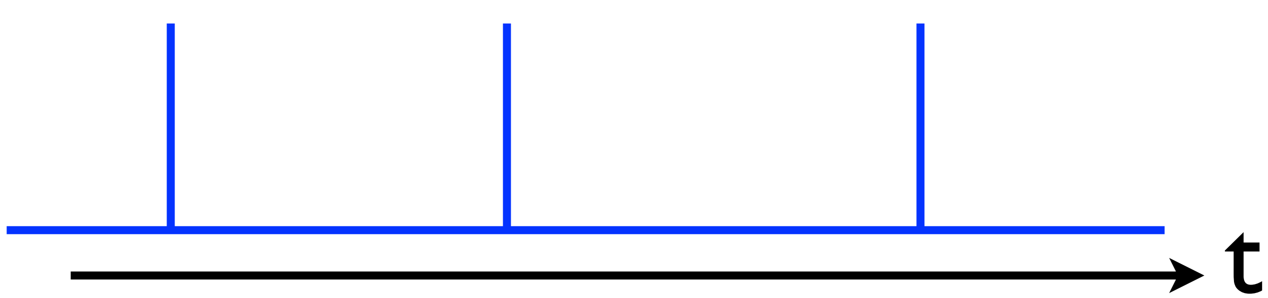
Embed $\Omega^{(X)}$ in an 'ambient' space ($\mathbb{R}^n$)



Output

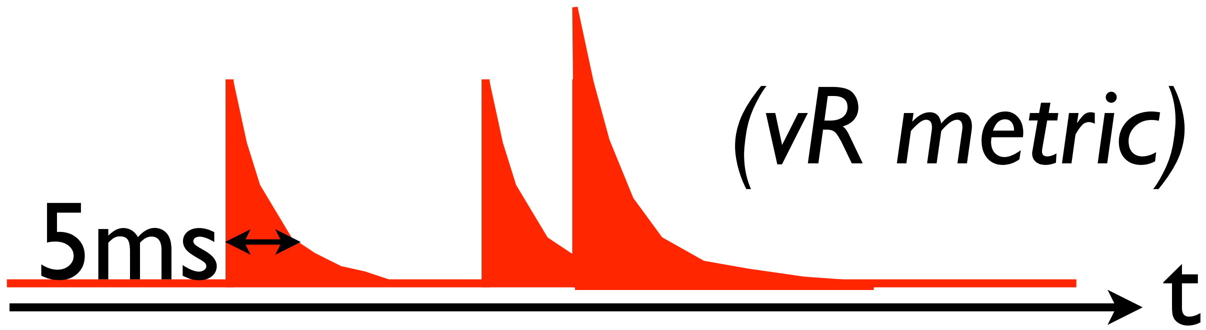


Input

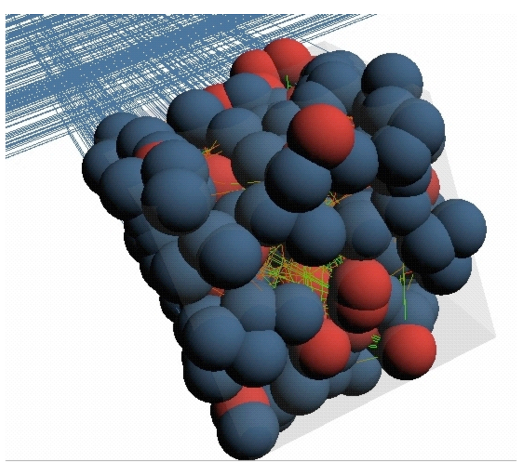


Embed $\Omega^{(X)}$ in an ‘ambient’ space ($\mathbb{R}^n$)

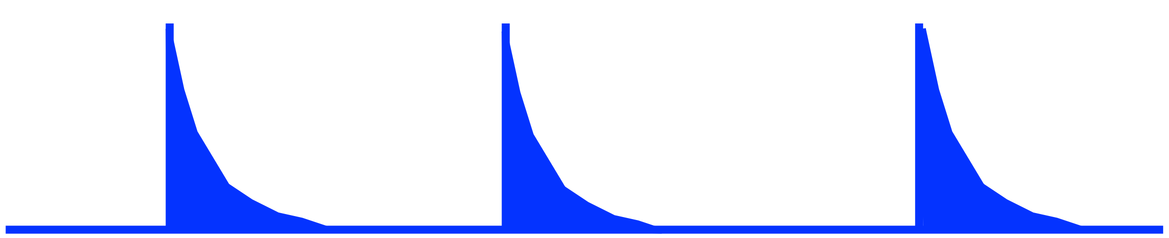
(vR metric)



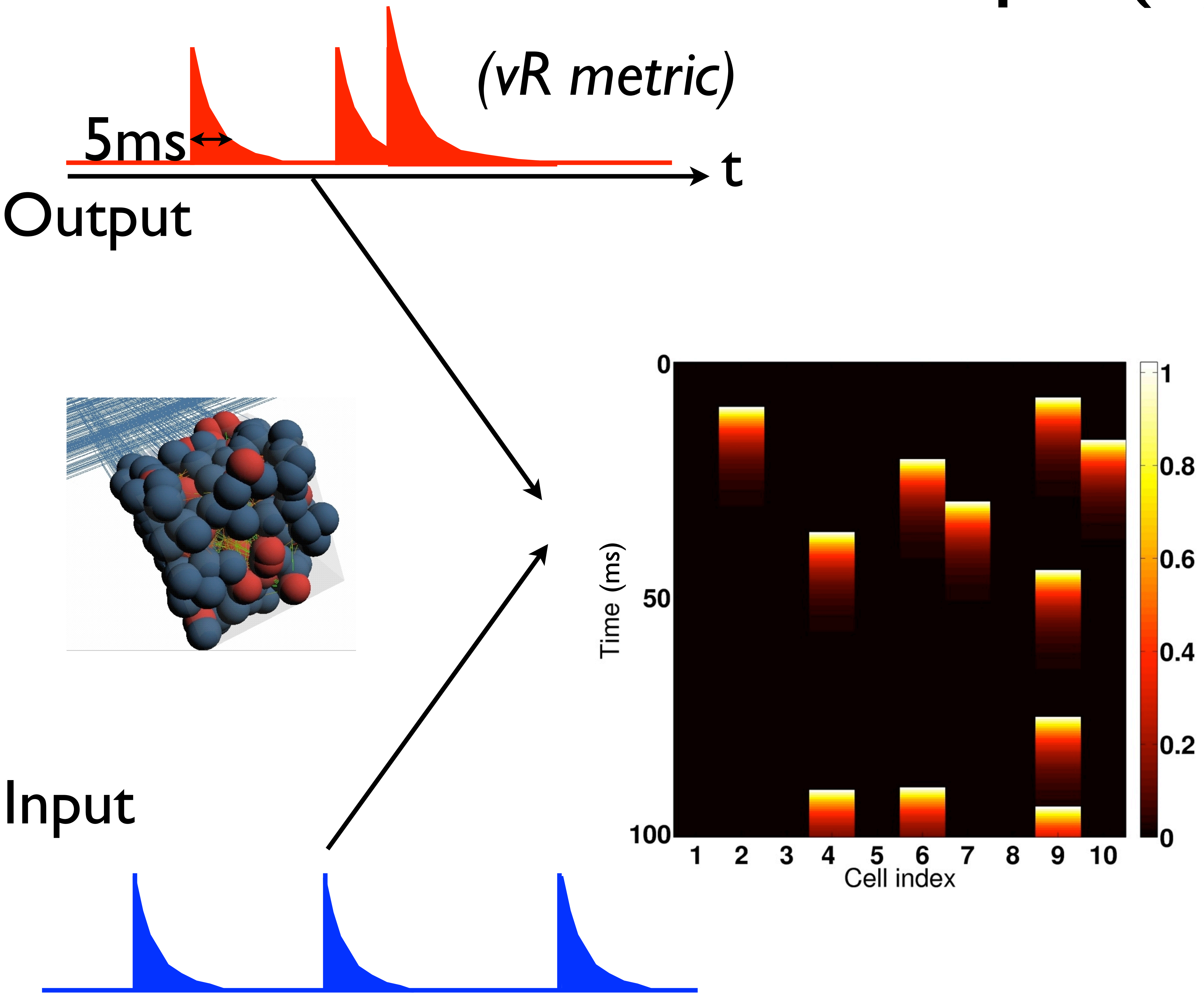
Output



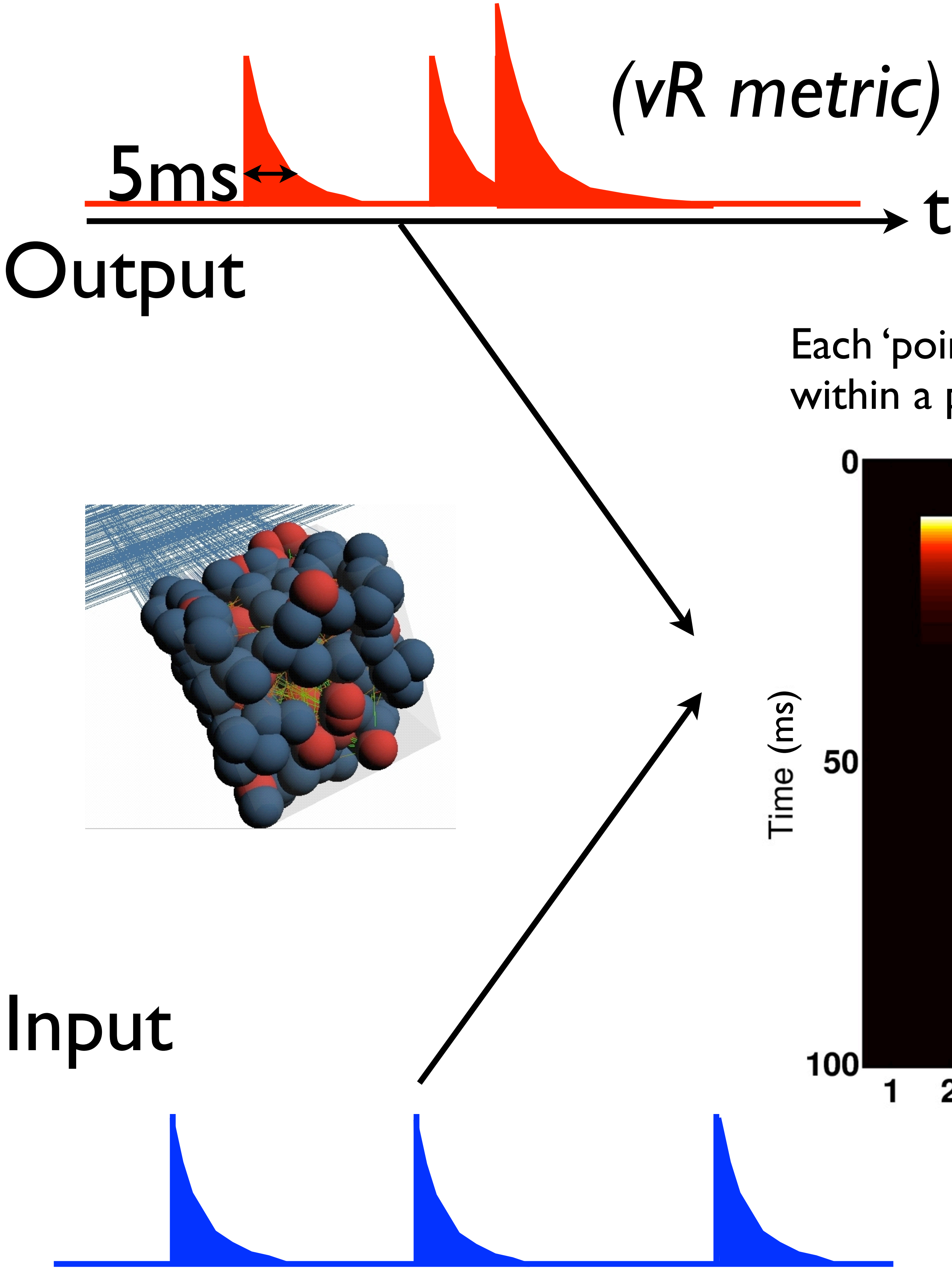
Input

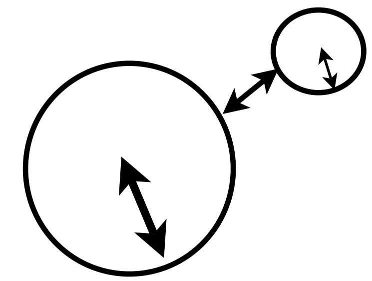


Embed $\Omega^{(X)}$ in an 'ambient' space ($\mathbb{R}^n$)

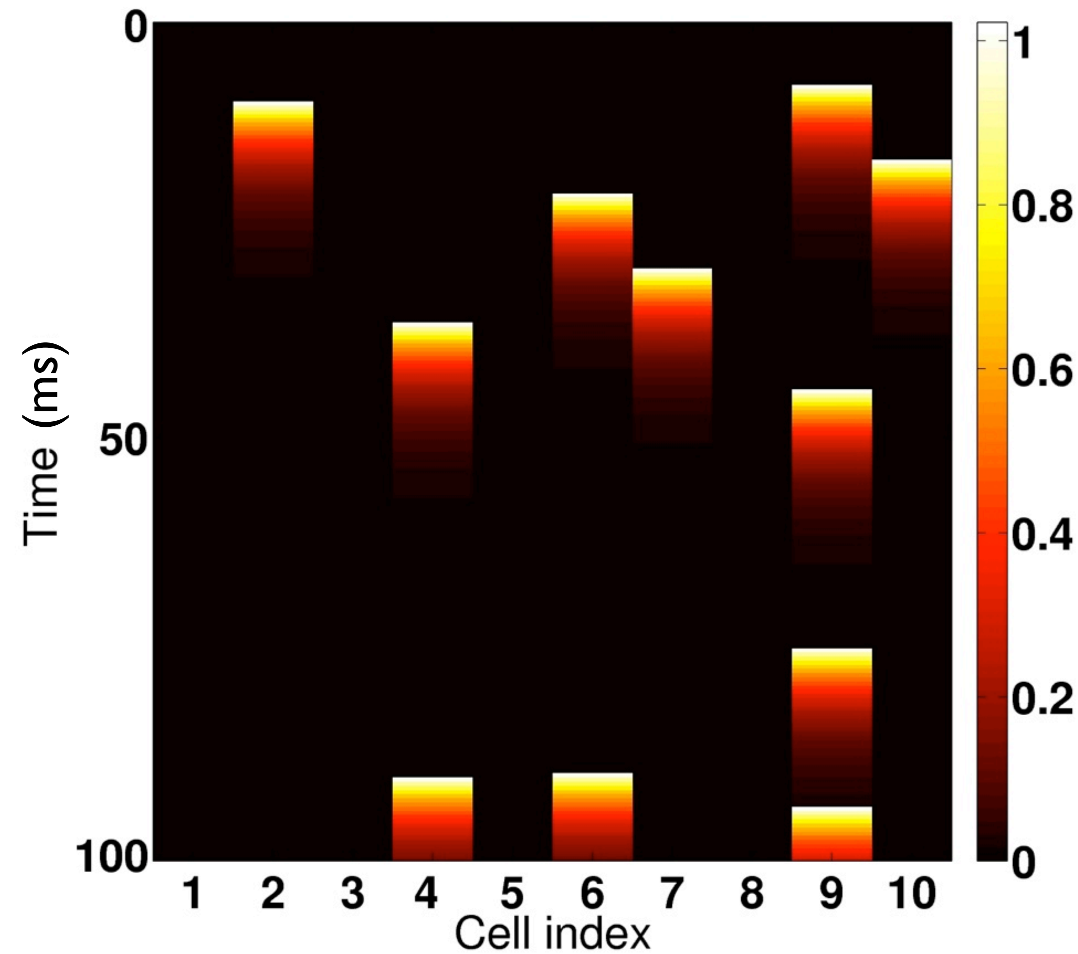


Embed $\Omega^{(X)}$ in an 'ambient' space ($\mathbb{R}^n$)

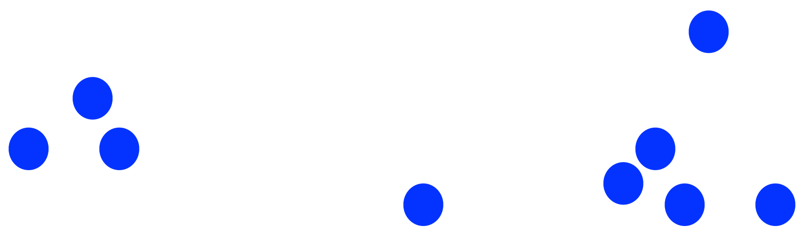




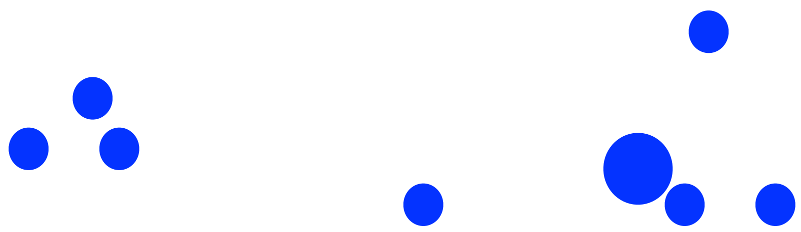
Each 'point' in the space represents activity of all cells within a particular time segment



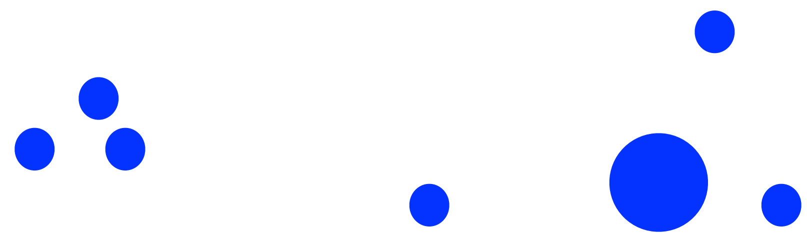
Hierarchical clustering



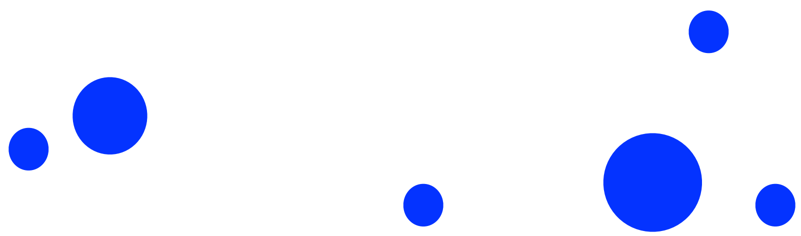
Hierarchical clustering



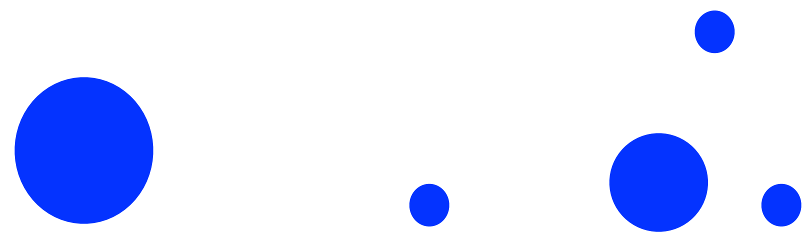
Hierarchical clustering



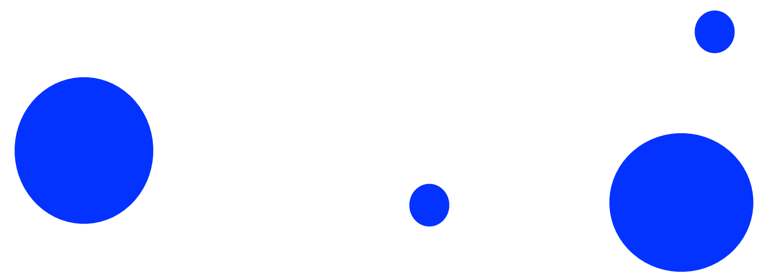
Hierarchical clustering



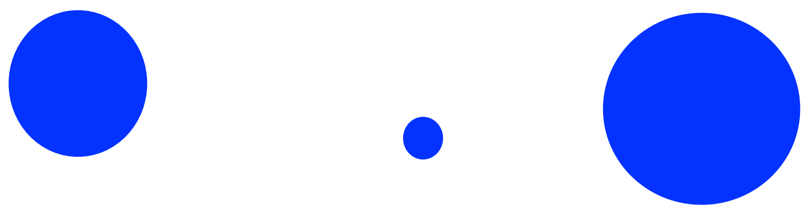
Hierarchical clustering



Hierarchical clustering

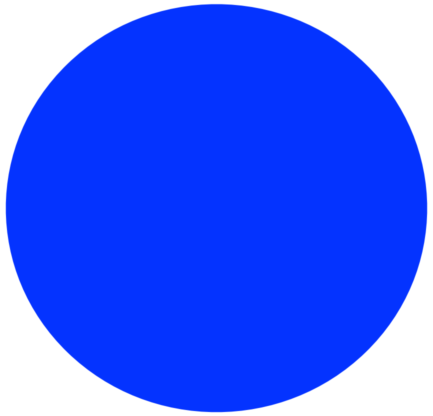


Hierarchical clustering



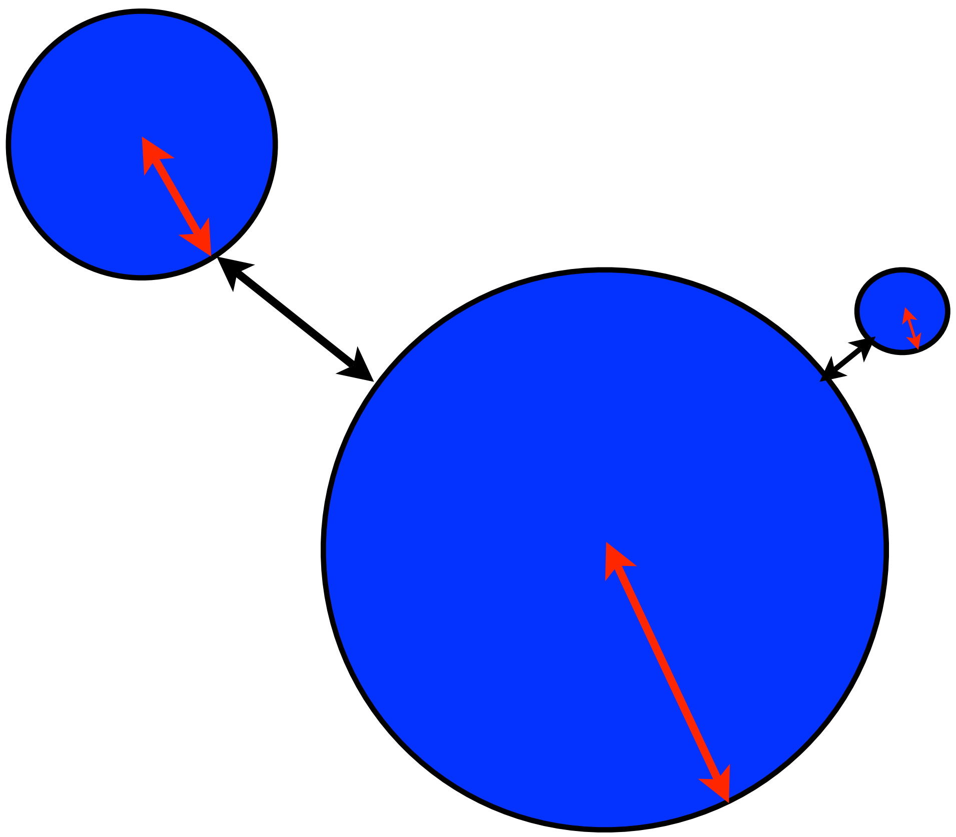
Hierarchical clustering

Hierarchical clustering



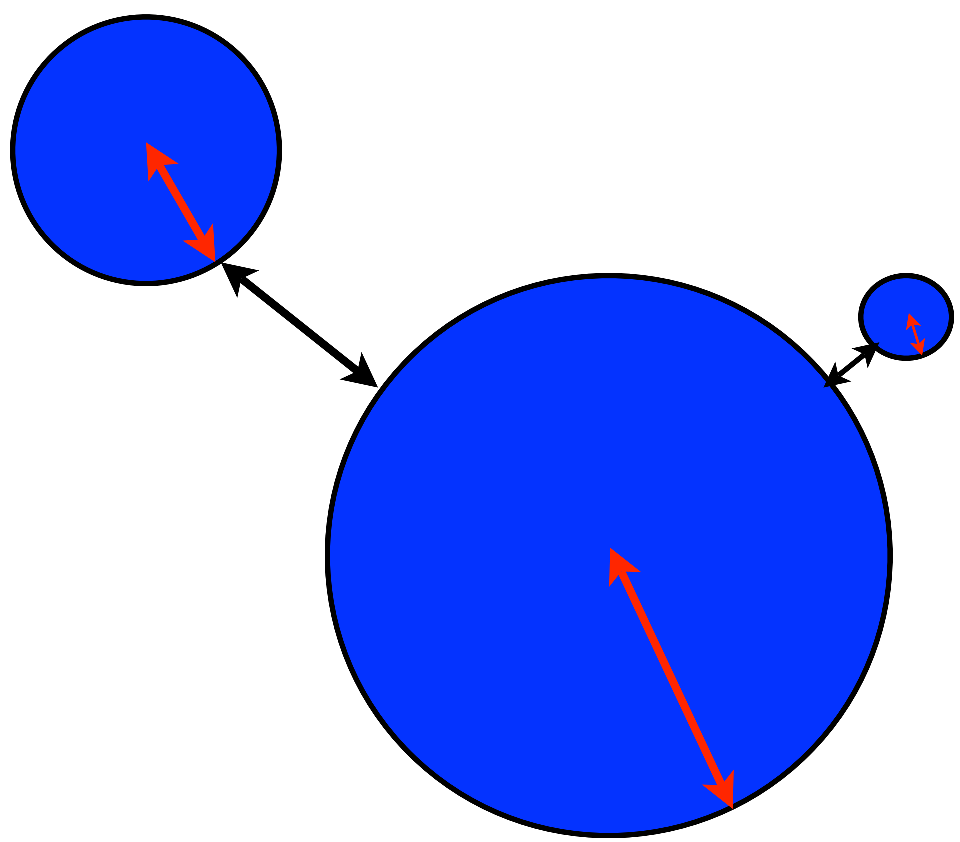
Hierarchical clustering

Clusters data into groups by merging successive closest points



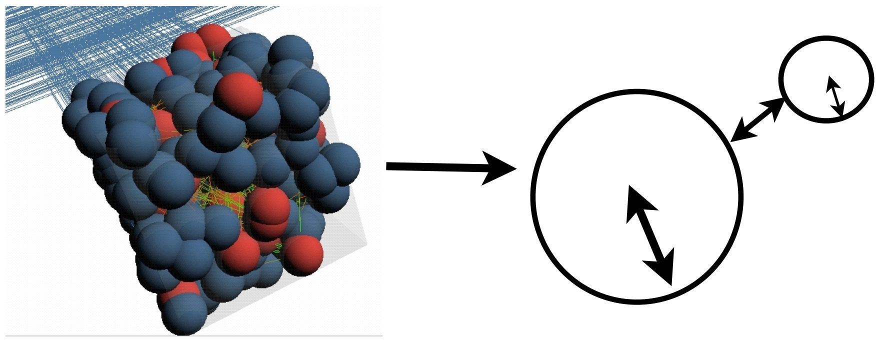
Hierarchical clustering

Clusters data into groups by merging successive closest points



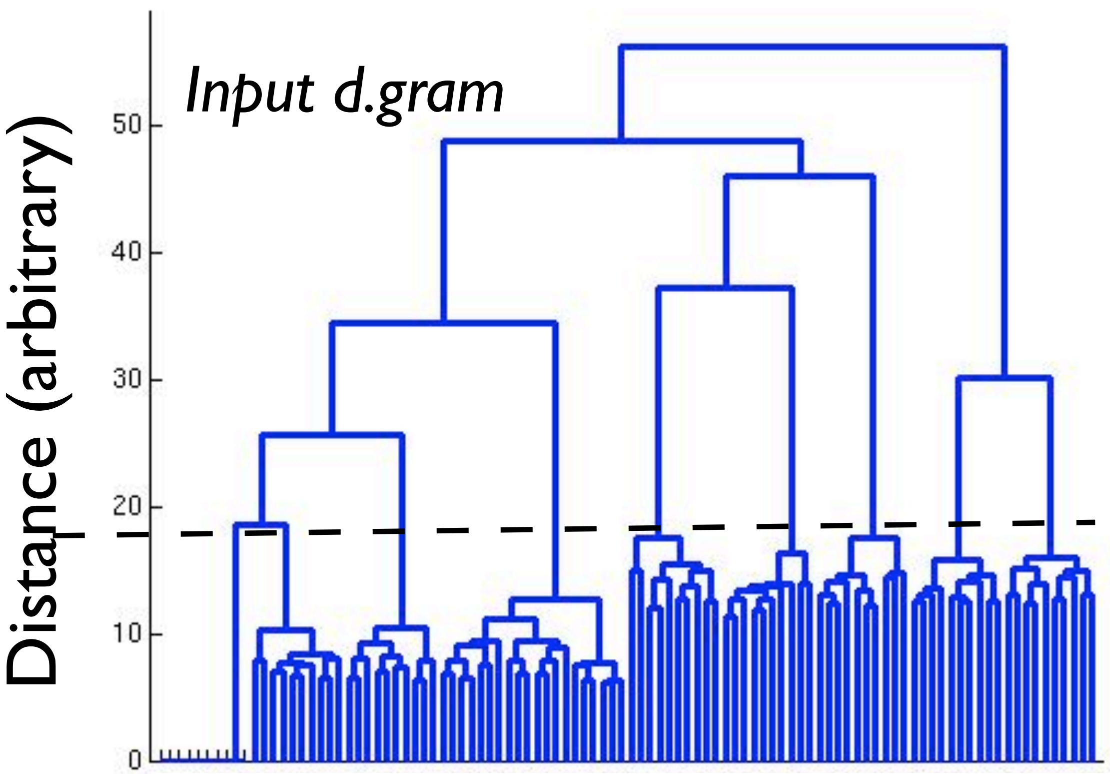
Use ‘ward’s linkage’ which clusters on the basis of squared distance

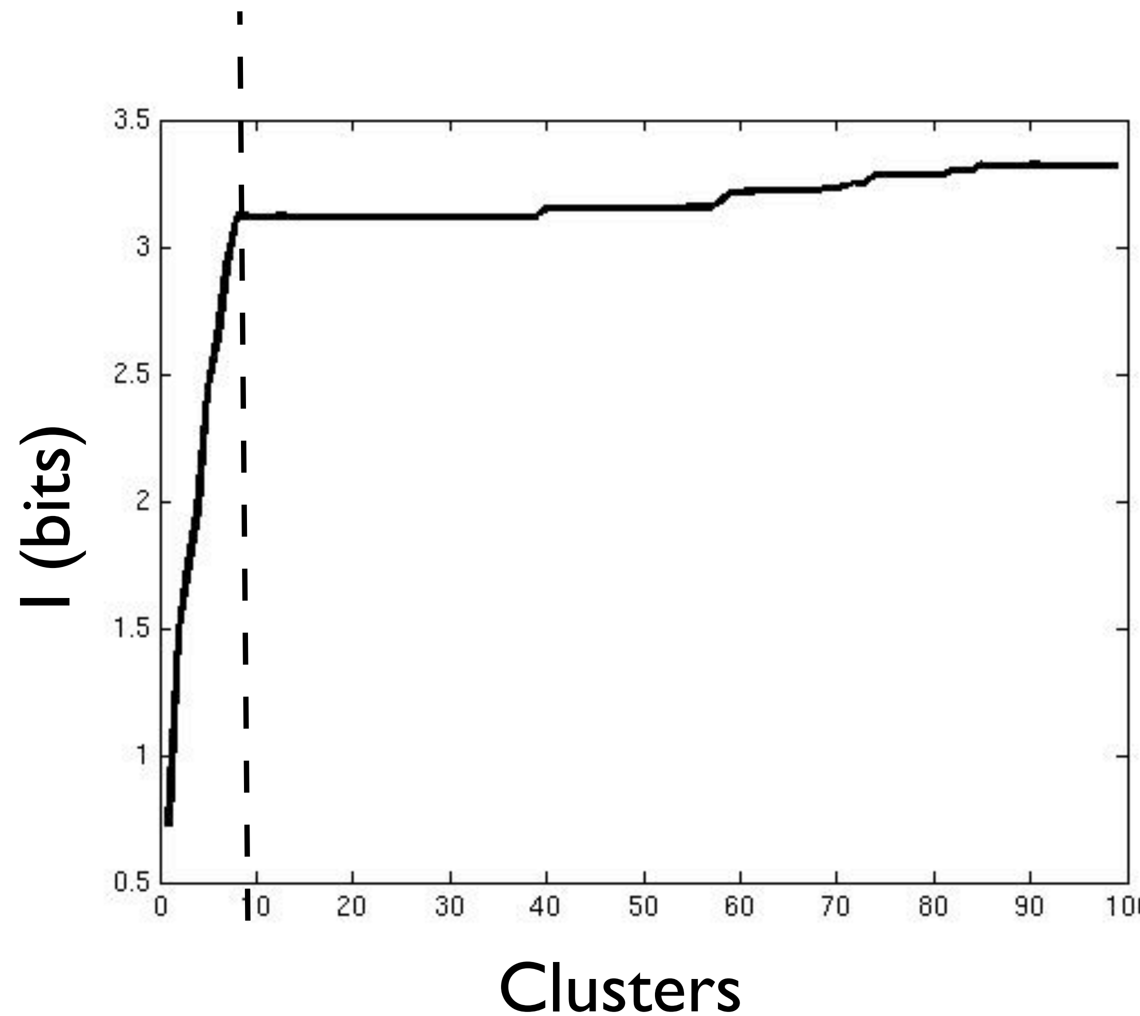
Hierarchical clustering gives variable resolution information measure



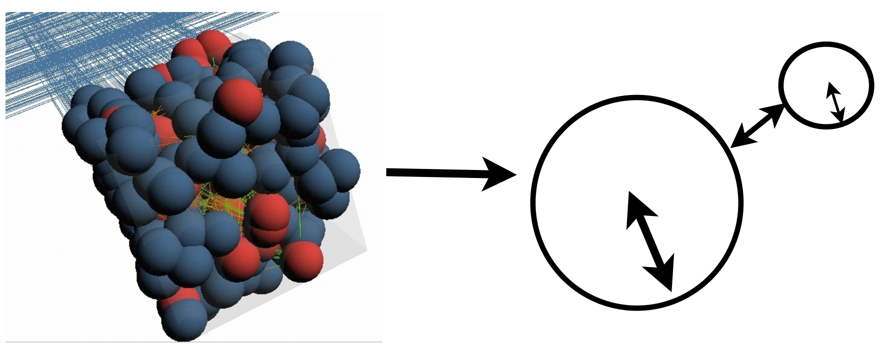
$$I(d,c) = \sum_{p \in P} \sum_{c \in C} p(d,c) \log[\frac{p(d,c)}{p(d)p(c)}]$$

Source entropy=3.32 bits



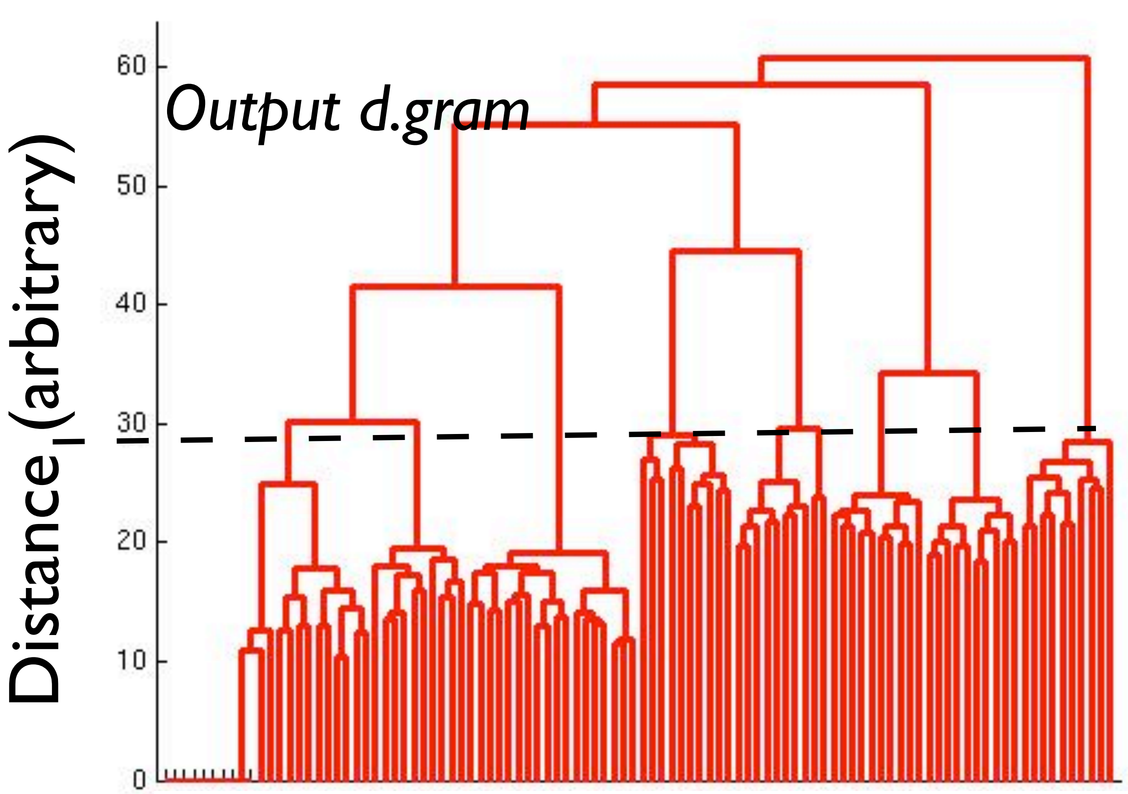


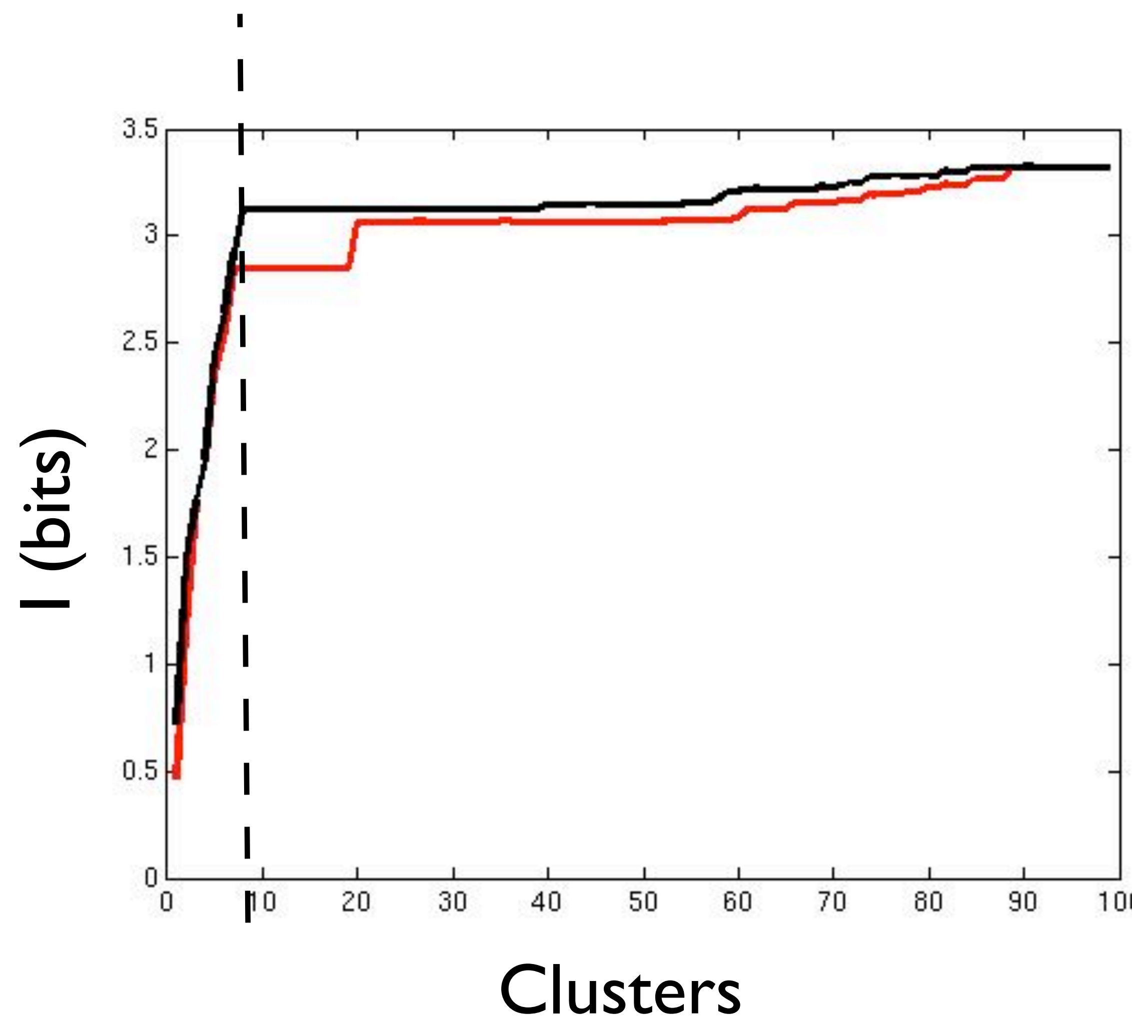
Hierarchical clustering gives variable resolution information measure



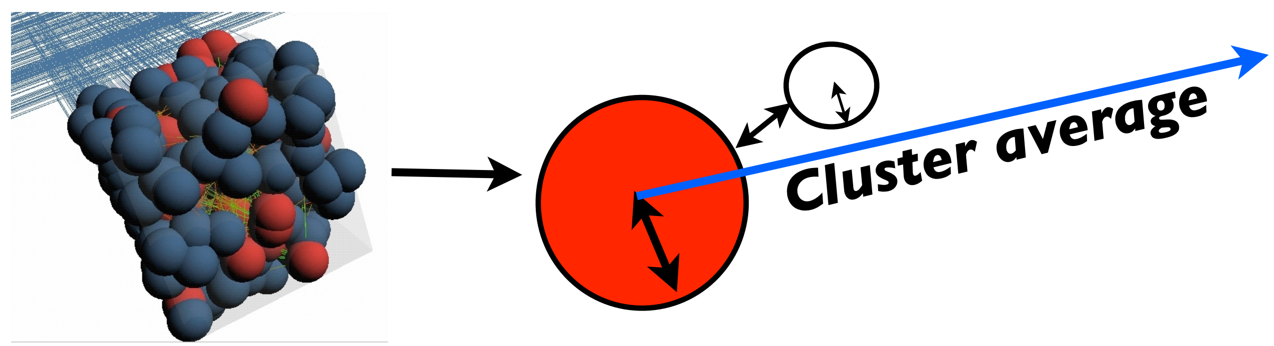
$$I(d,c) = \sum_{p \in P} \sum_{c \in C} p(d,c) \log[\frac{p(d,c)}{p(d)p(c)}]$$

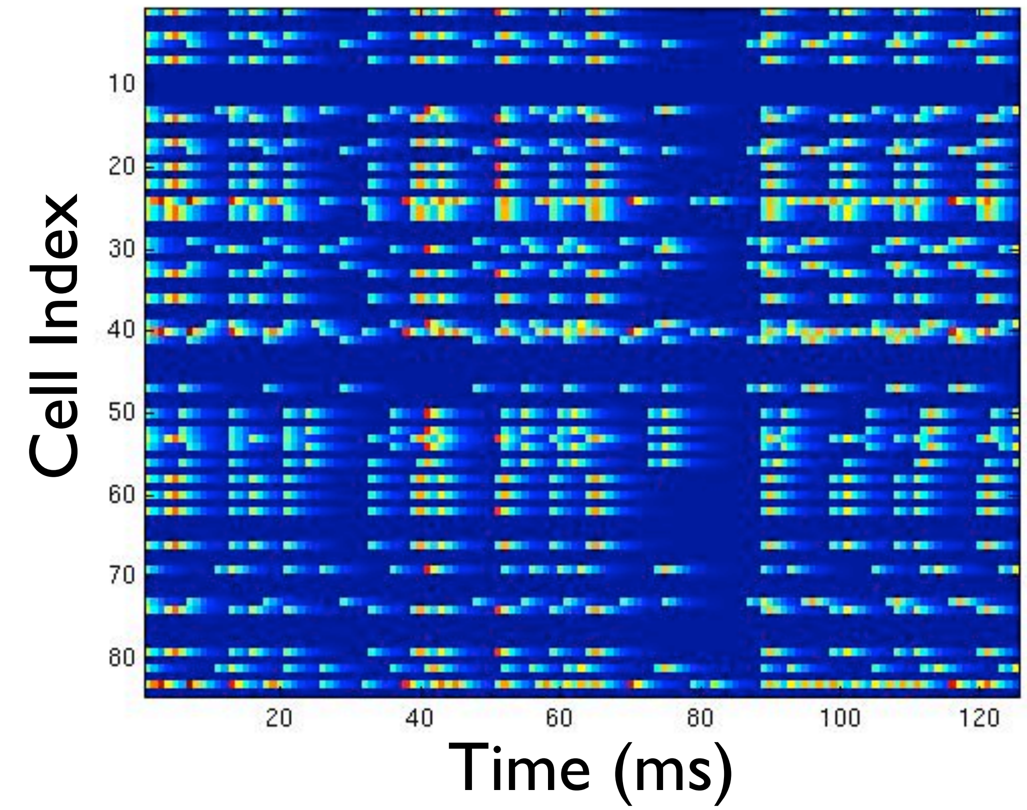
Source entropy=3.32 bits



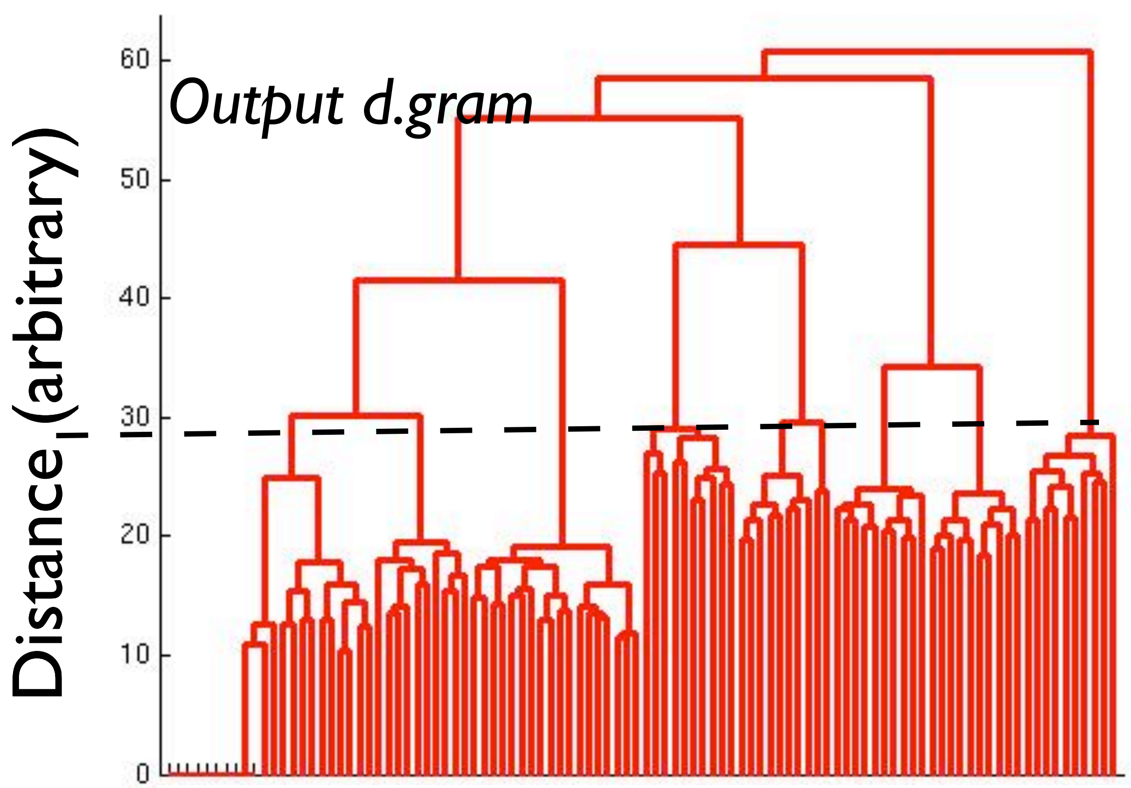


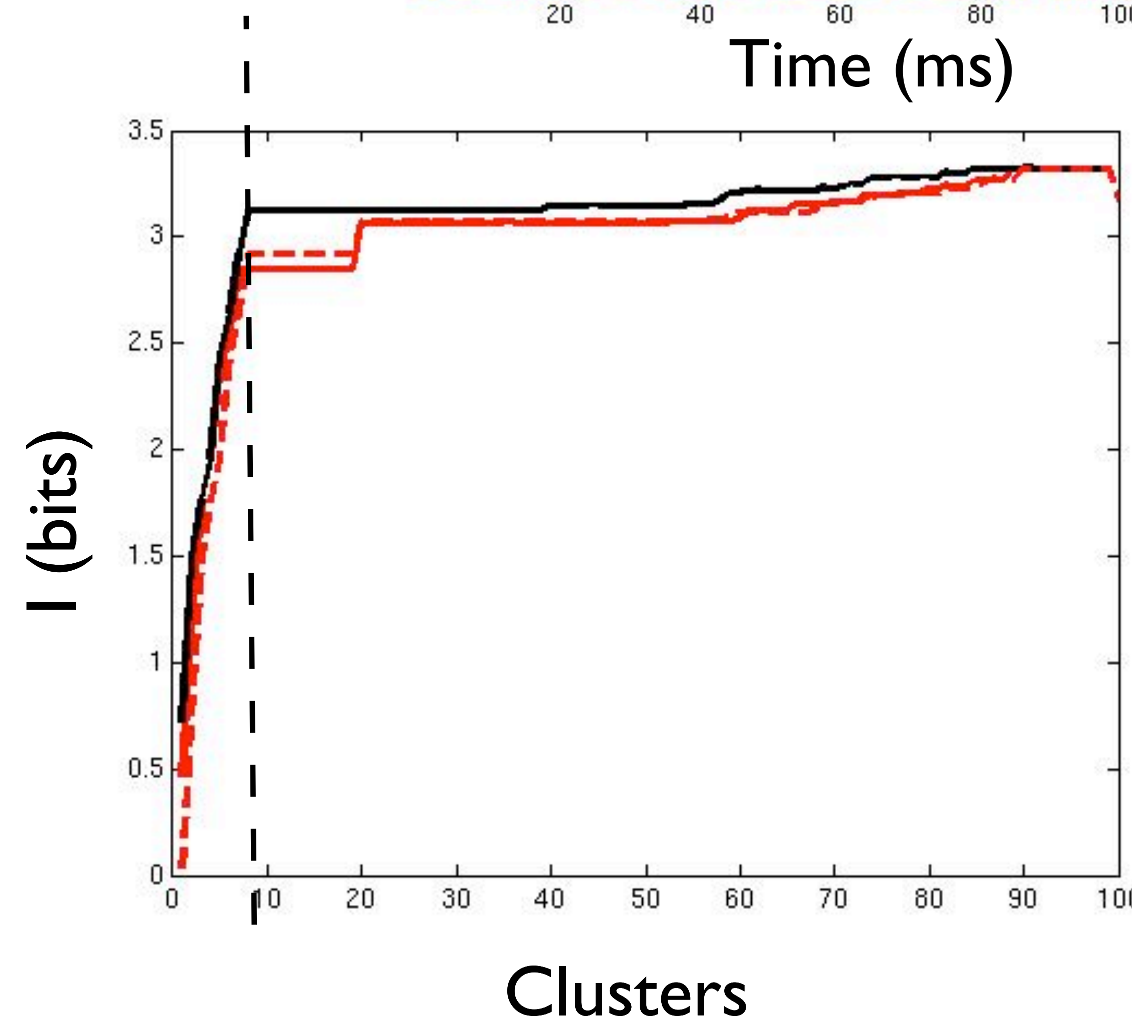
Building a 'decoder'



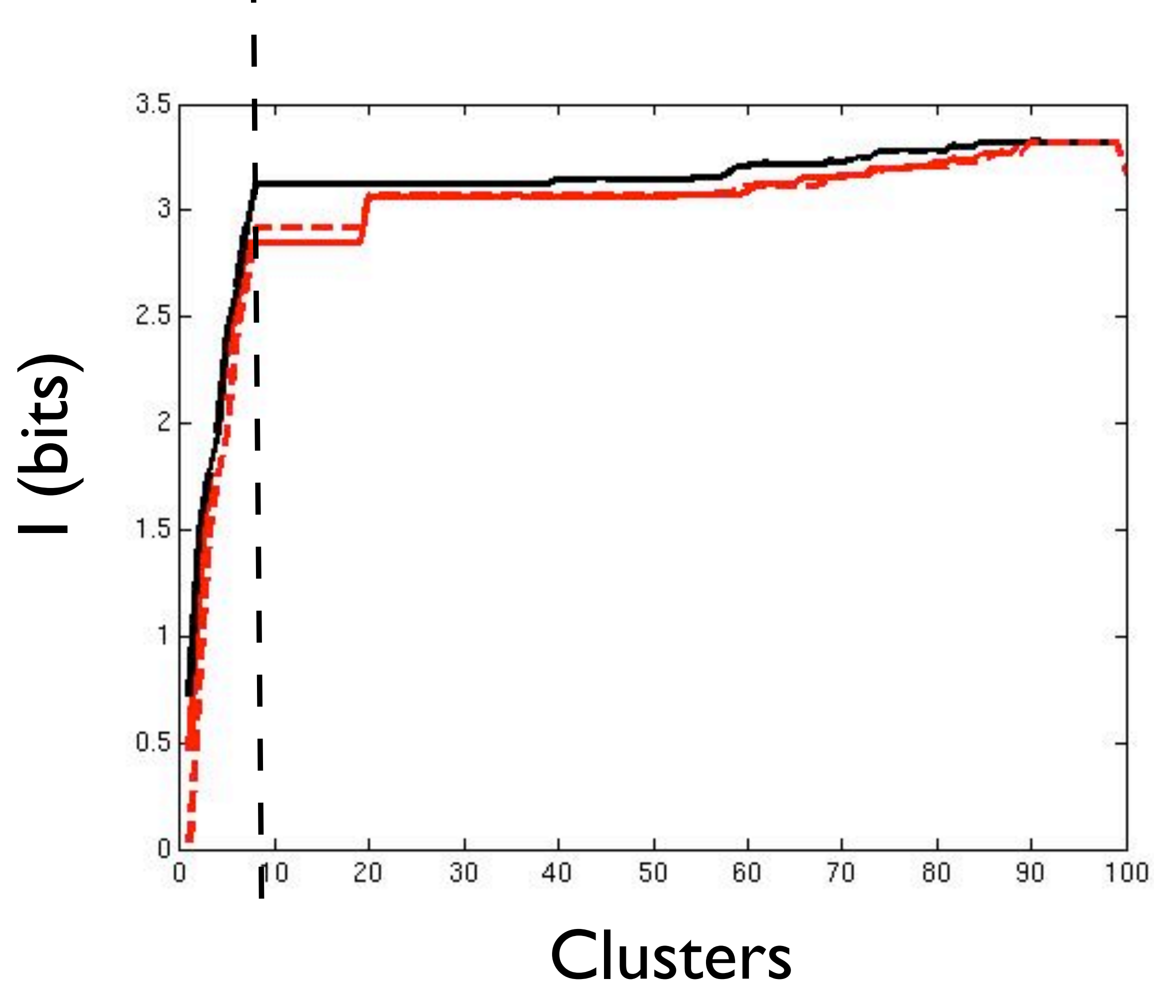


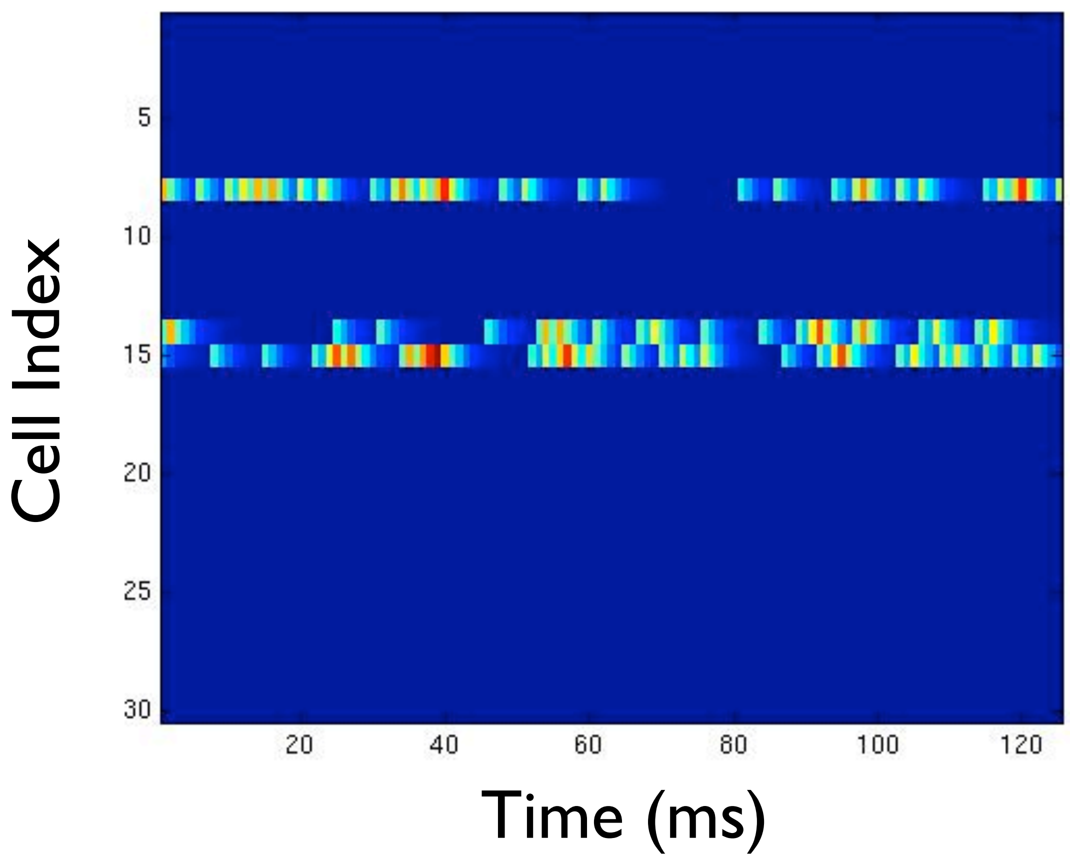
Source entropy=3.32 bits

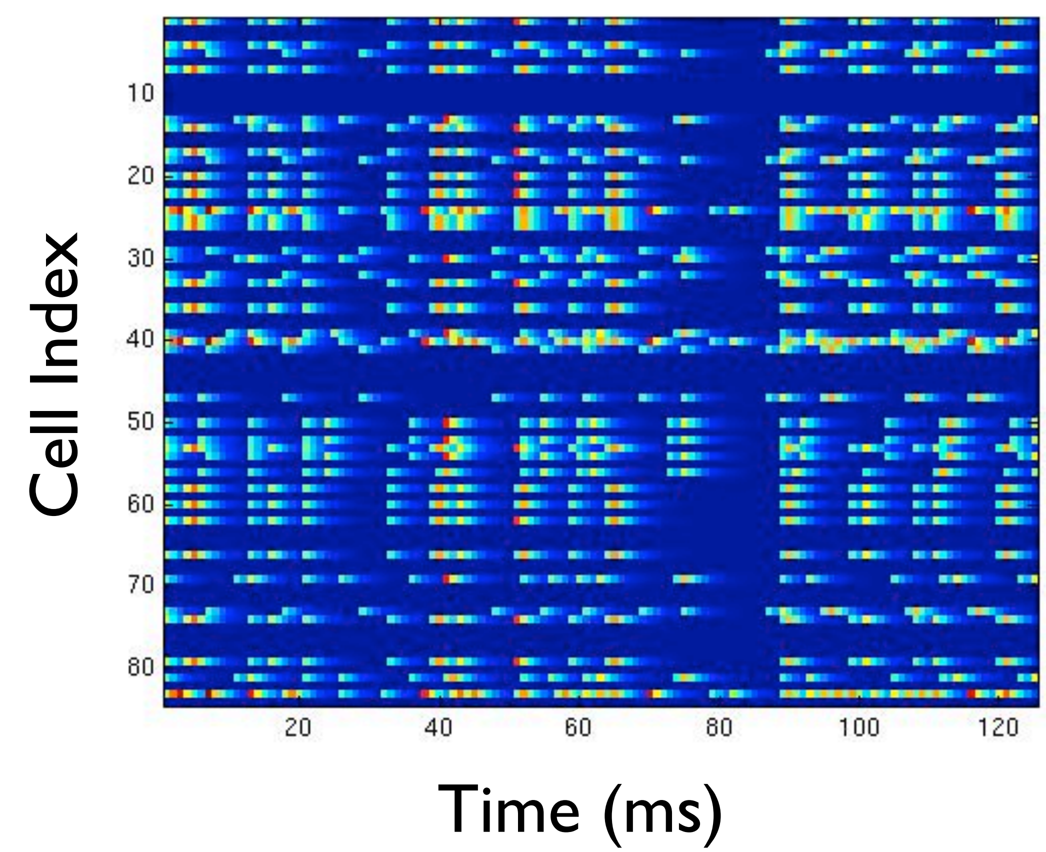




This network could likely be tuned by **inhibition**
(also **desynchronisation?**) to transfer more information







Summary

Application of metric techniques to spike train space allows information theoretic analysis

Clustering allows identification of putative transformations

This can be used to repeat the previous granule cell study but with a more realistic model

Further work

Not clear how to choose the space/clustering method - ideally build a 'neurometric'

Application of topology - removing the need to define a metric