CS375 / Psych 249:
Large-Scale Neural Network Models for Neuroscience

Lecture 4: Model-Brain Mapping Methods

2025.01.14-16

Atlas Kazemian
Departments of Psychology
Stanford Neuroscience and Artificial Intelligence Laboratory
Stanford University

Outline

- Comparison methods

  1. The early days
     - Examples: subjective comparisons, sparsity, response properties.

  2. Using stimulus-by-stimulus similarity matrices.
     - Examples: RSA, CKA

  3. Learning a mapping from models to neural data.
     - Examples: One to one matching, linear regression, procrustes, soft matching, nonlinear mapping

- Selecting the right method:

  - Bidirectionally vs symmetry.
  - Using IATC for choosing the correct metric

- Noise ceiling estimates

Why do we compare neural networks to the brain?

As **scientists** we care about understanding the brain:

- Does the model encode similar features as neural populations?
- Is the model solving the task using similar transformations?
- Which architectural or learning constraints are allow us to better explain neural responses.

As **engineers** we care about building a good model of the brain:

- Models allow rapid, large-scale testing of hypotheses that would be infeasible in humans or animals (ablation studies)
- Models can inform brain–computer interfaces and personalized treatments.

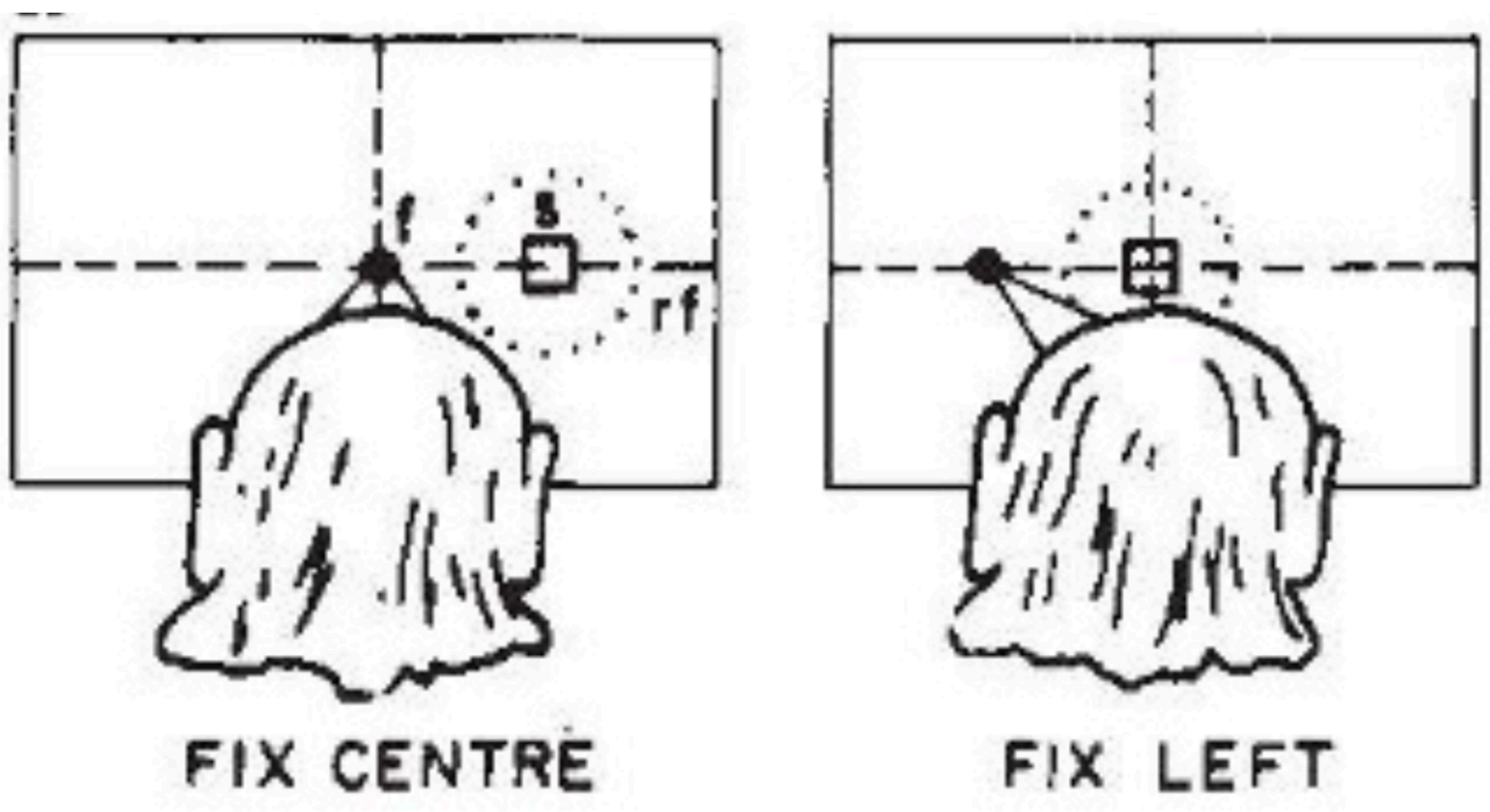
Early days: subjective comparisons

Idea: subjectively compare properties in models and neural data

Zipser & Andersen (1988): Study how the brain encodes retinal location and eye position together to represent object location in posterior parietal cortex.

Neural data

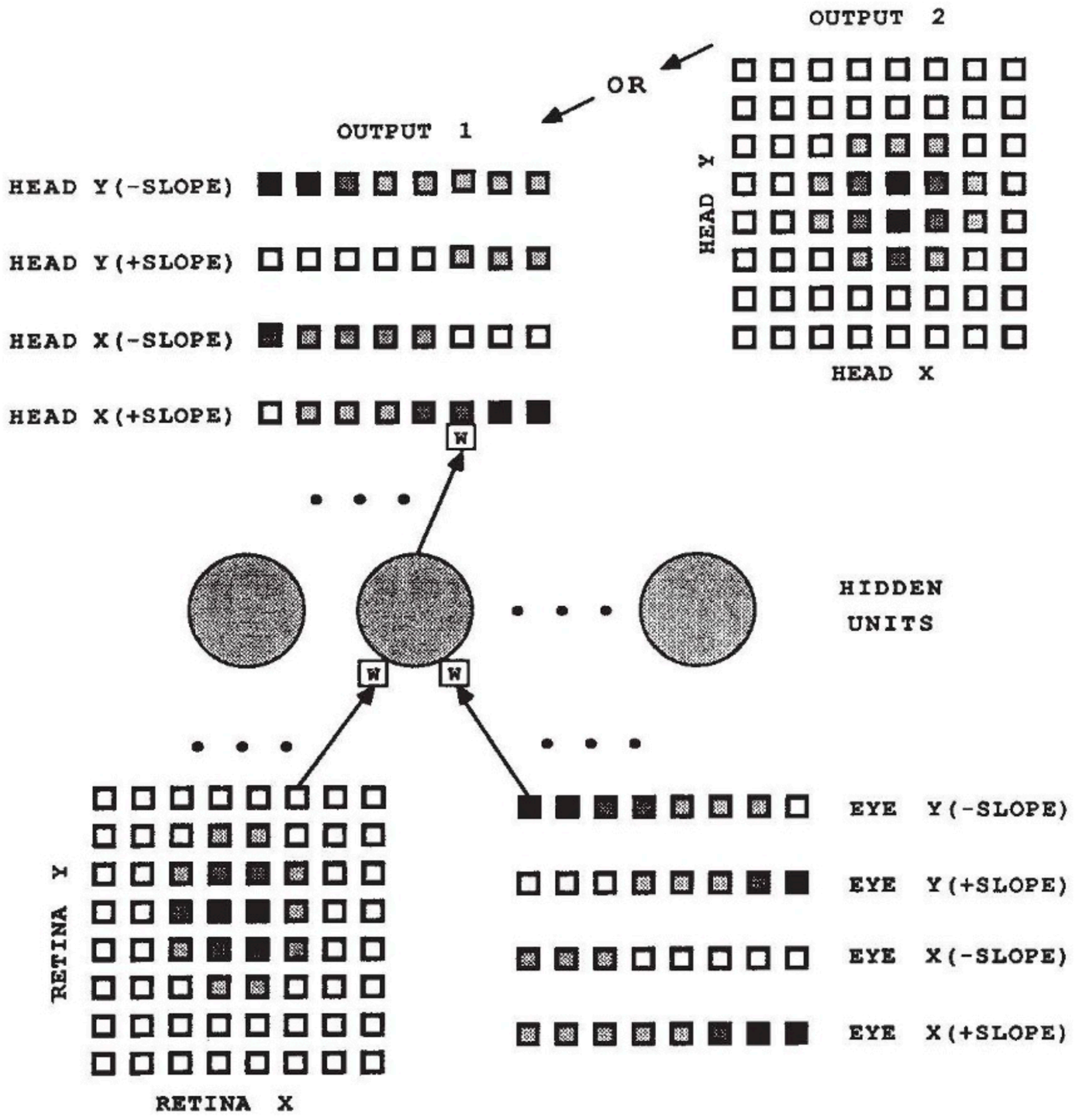
- Single-unit recordings from area 7a in awake monkeys
- Visual stimulus is flashed at many retinal (x, y) locations during fixation
- Firing rate measured for each location



Zipser & Andersen, 1988

Neural Network Model

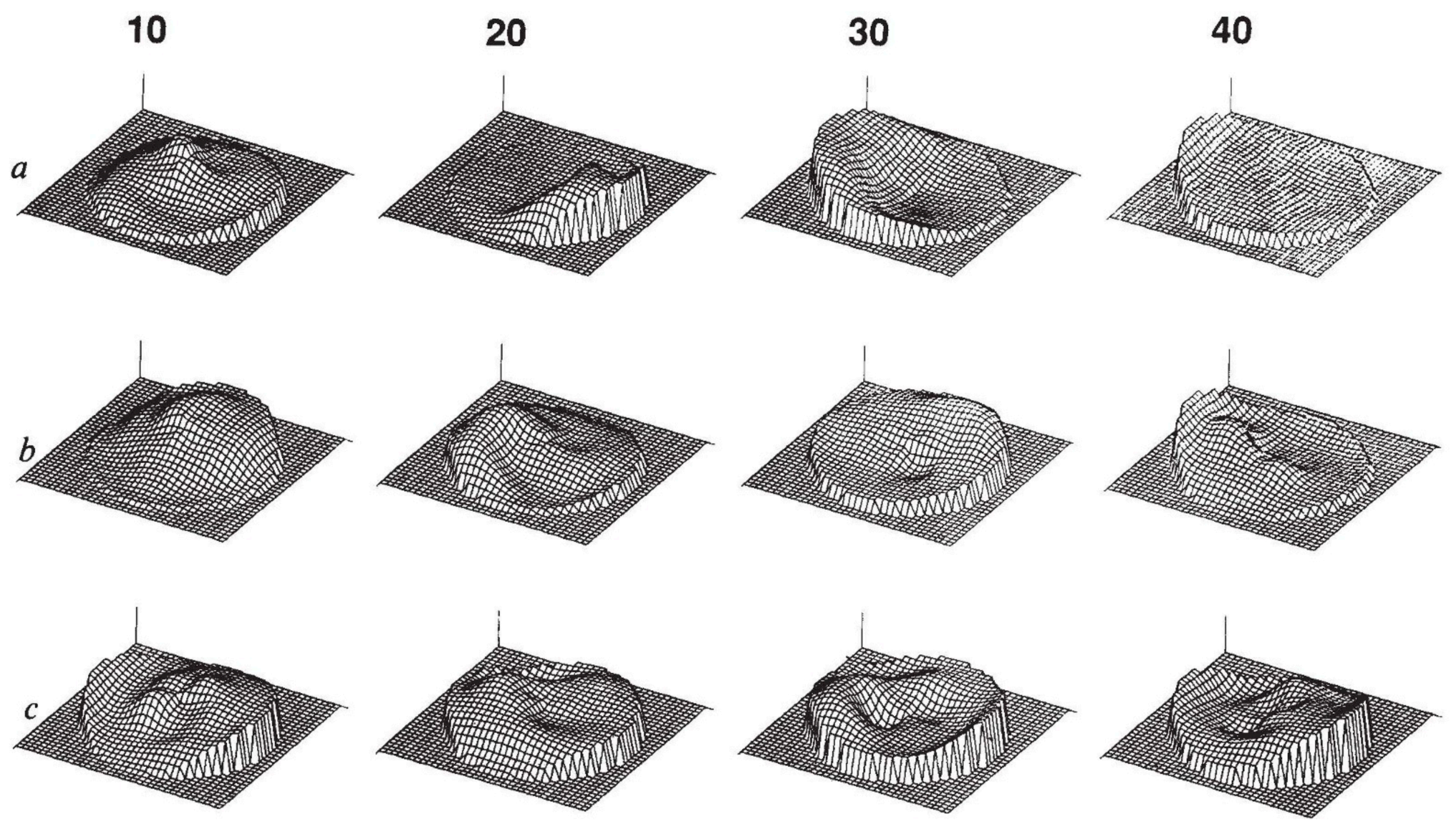
- 3-layer feedforward trained with backpropagation
- **Inputs:**
    1. Retinal position
    2. Eye position
- **Task:** Learn head-centered target locations



Zipser & Andersen, 1988

Comparing neural data with the model

Monkey receptive fields



Single peak

Single peak + complexities

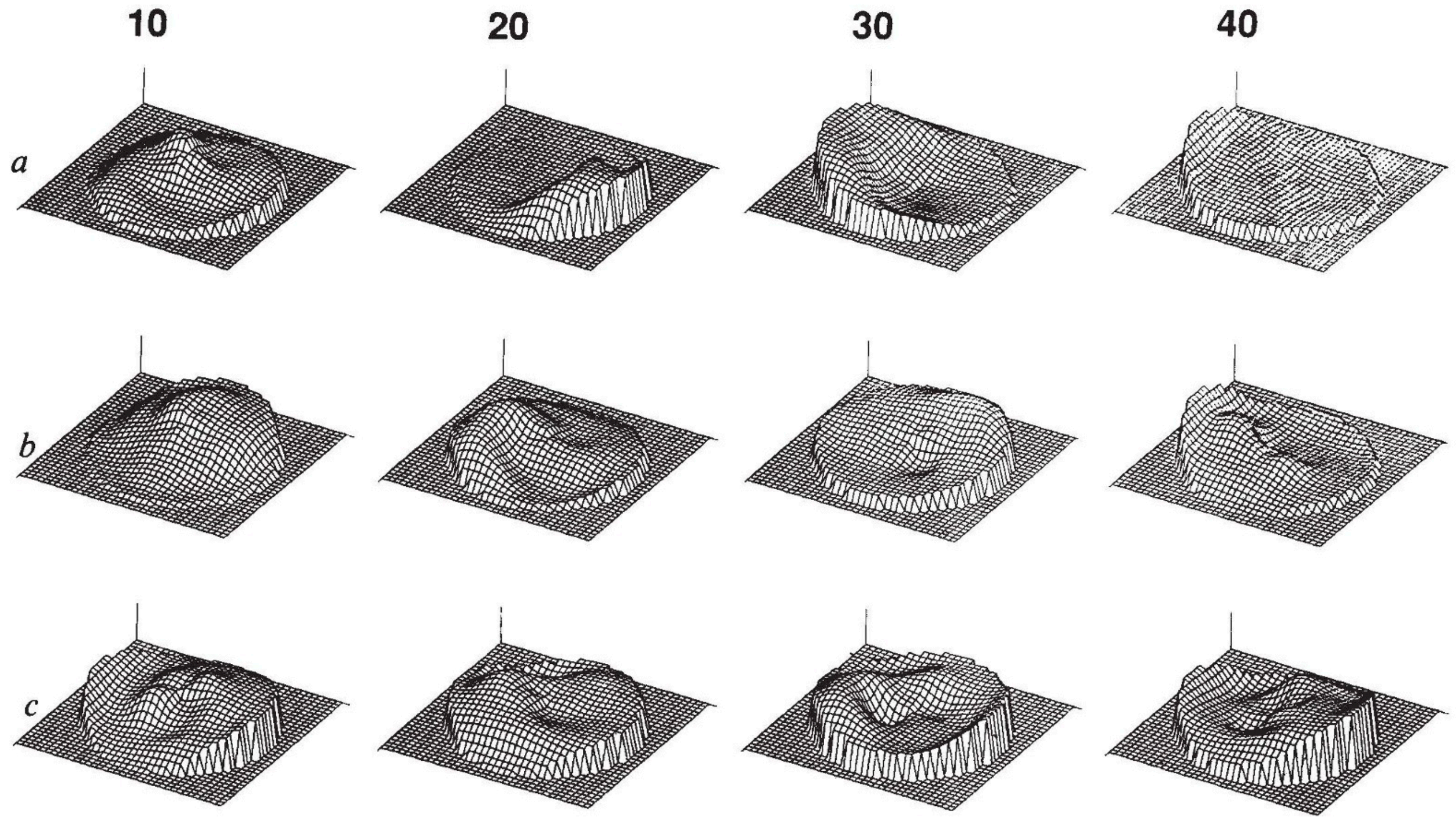
Multi peak

Eccentricity

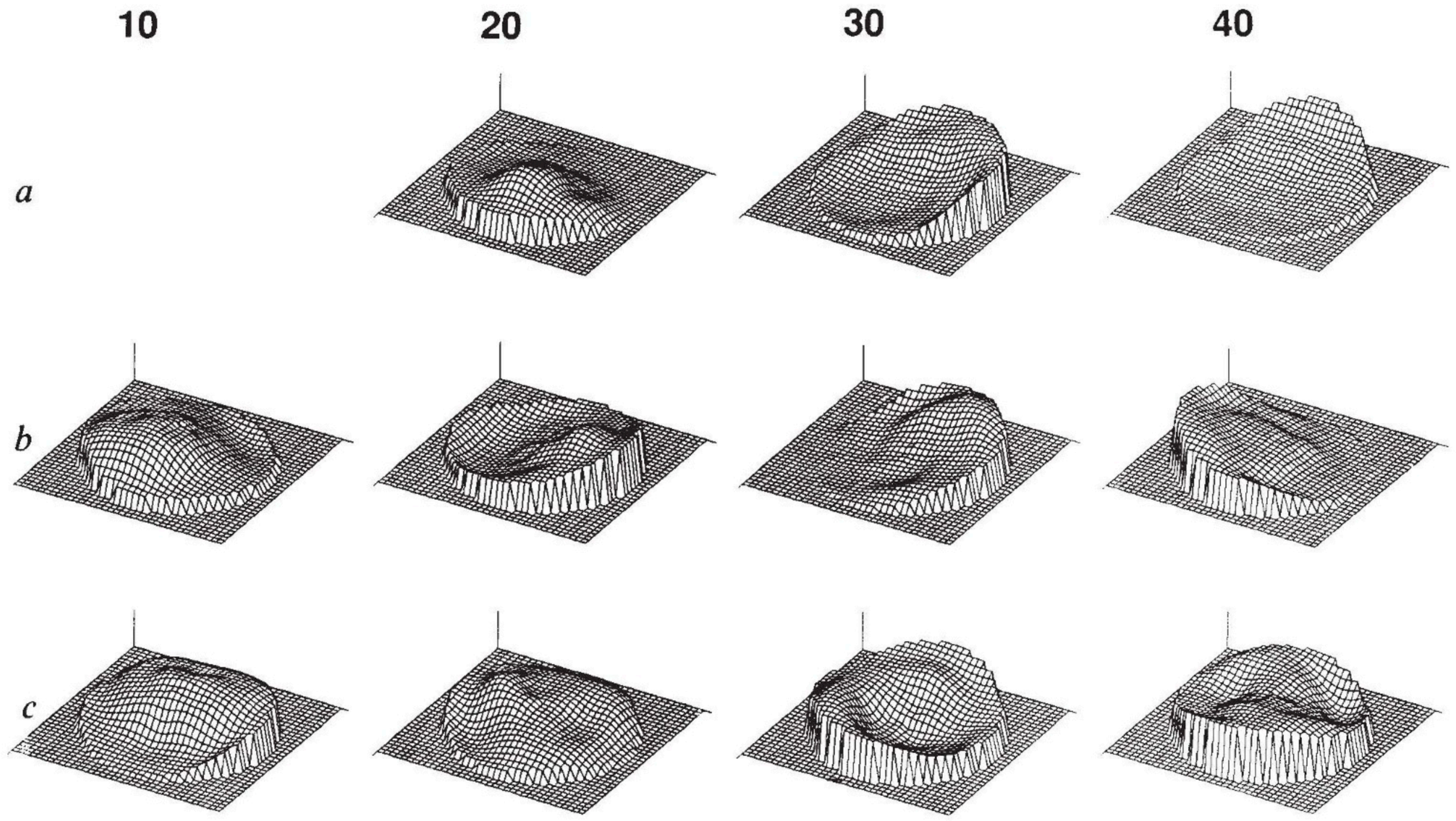
Zipser & Andersen, 1988

Comparing neural data with the model

Monkey receptive fields



Model receptive fields

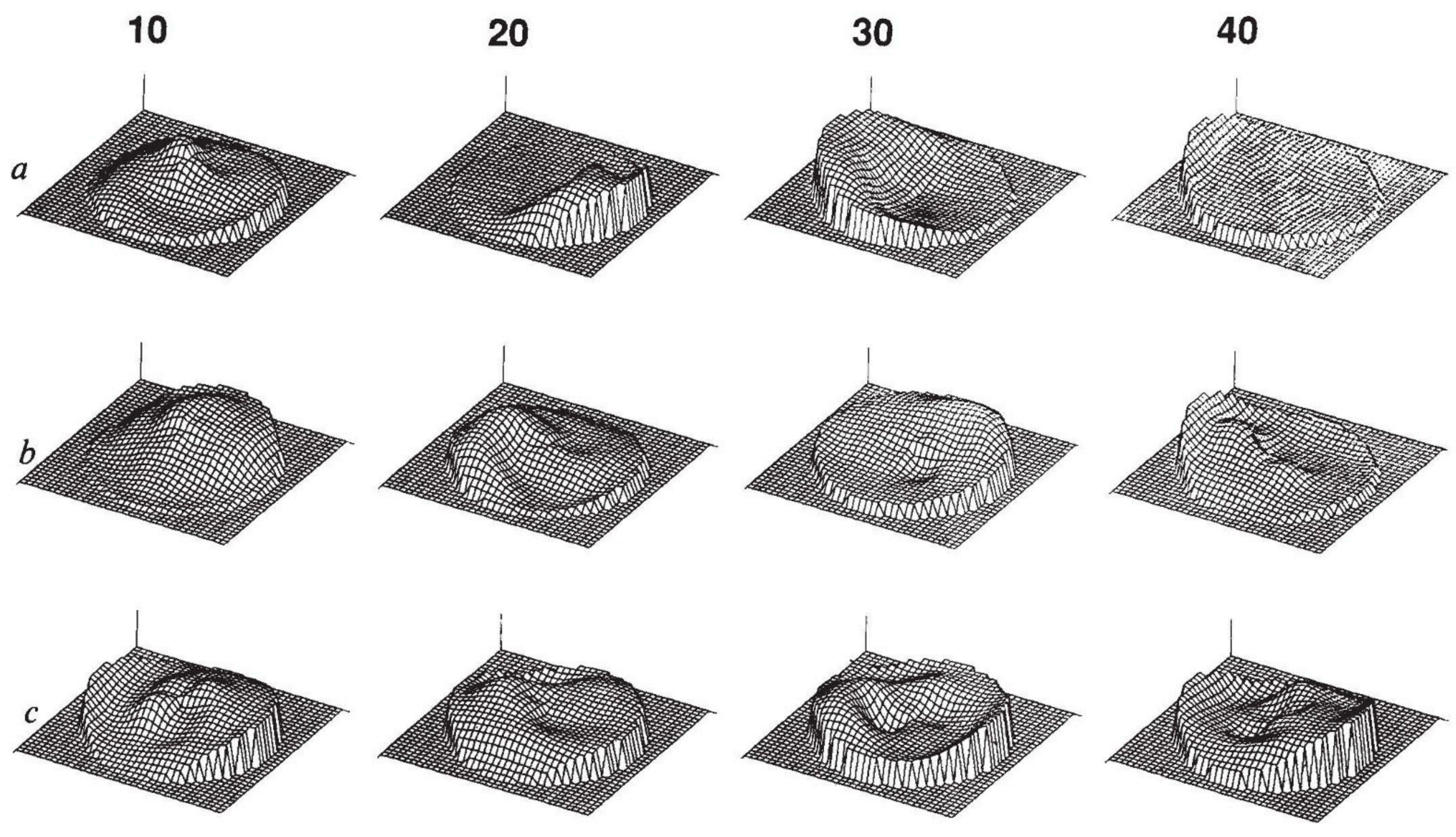


Zipser & Andersen, 1988

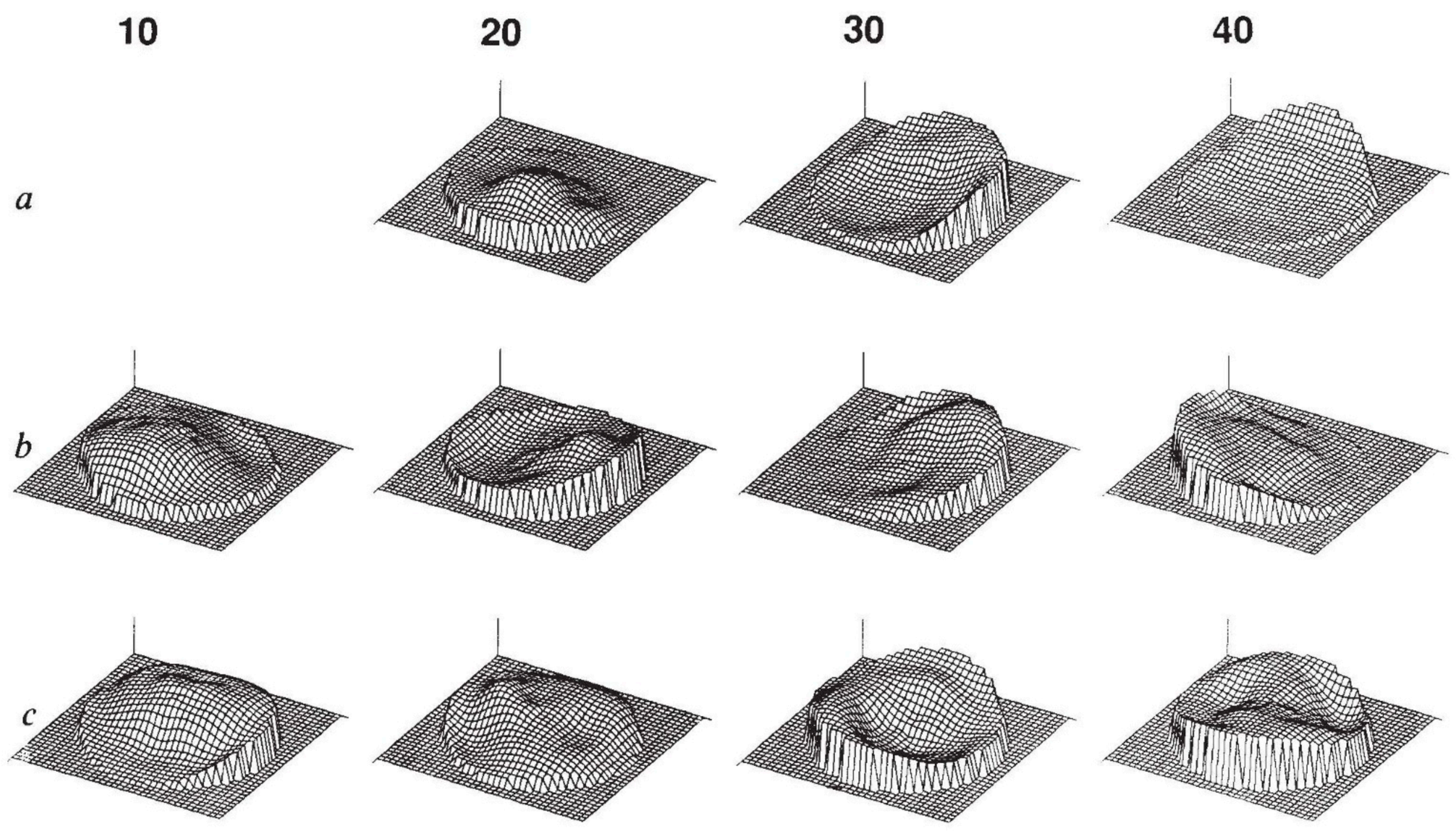
Comparing neural data with the model

…"The comparison process contains an element of subjectivity, but it demonstrates that the trained model generates retinal receptive fields remarkably similar to the experimentally observed fields."…

Monkey receptive fields



Model receptive fields



Zipser & Andersen, 1988

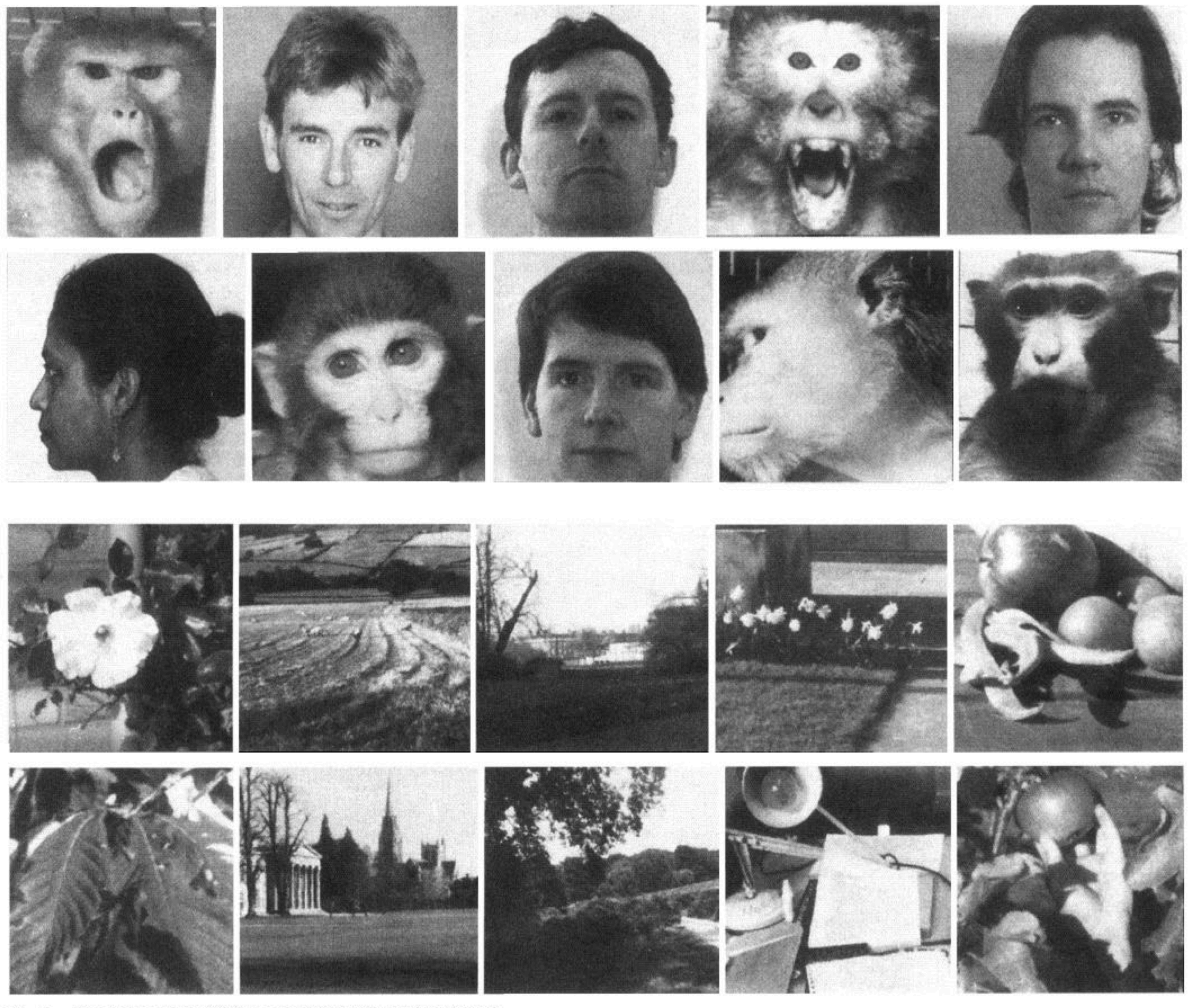
Early days: comparing sparseness

Idea: Move beyond subjective comparison by comparing sparseness and population statistics

Rolls & Tovee (1995): Are object representation in IT encoded using a dense, localist, or sparse distributed code?

Neural data

- Single-unit recordings from macaque IT
- Monkeys viewed large “diverse” sets of complex visual stimuli (objects, faces, scenes)

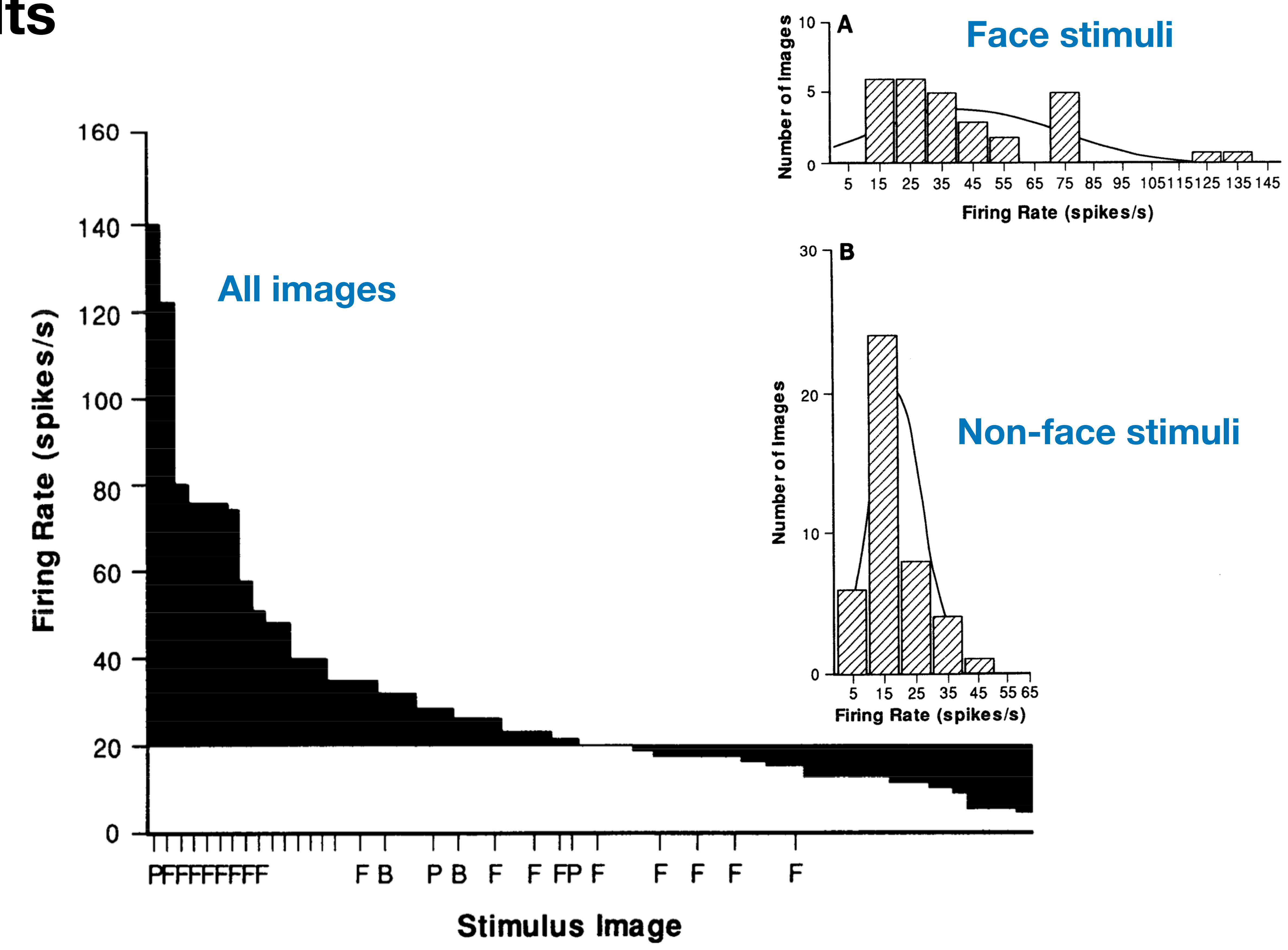


Rolls & Tovee, 1995

Representational theories

1. **Dense distributed coding:** Many neurons active for most stimuli
2. **Localist (grandmother-cell)** coding: One neuron per object
3. **Sparse distributed coding:** Few neurons active per stimulus

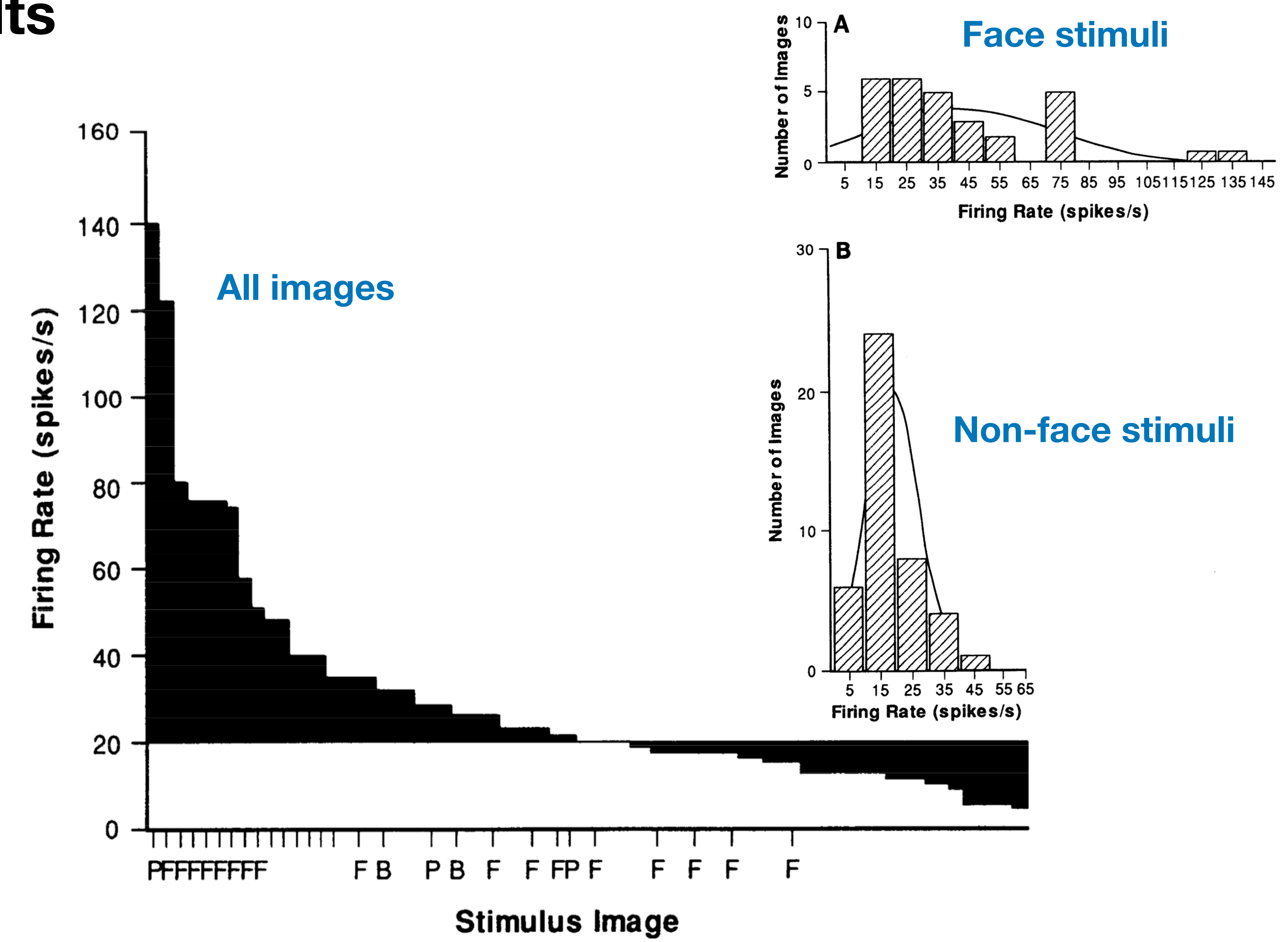
Results

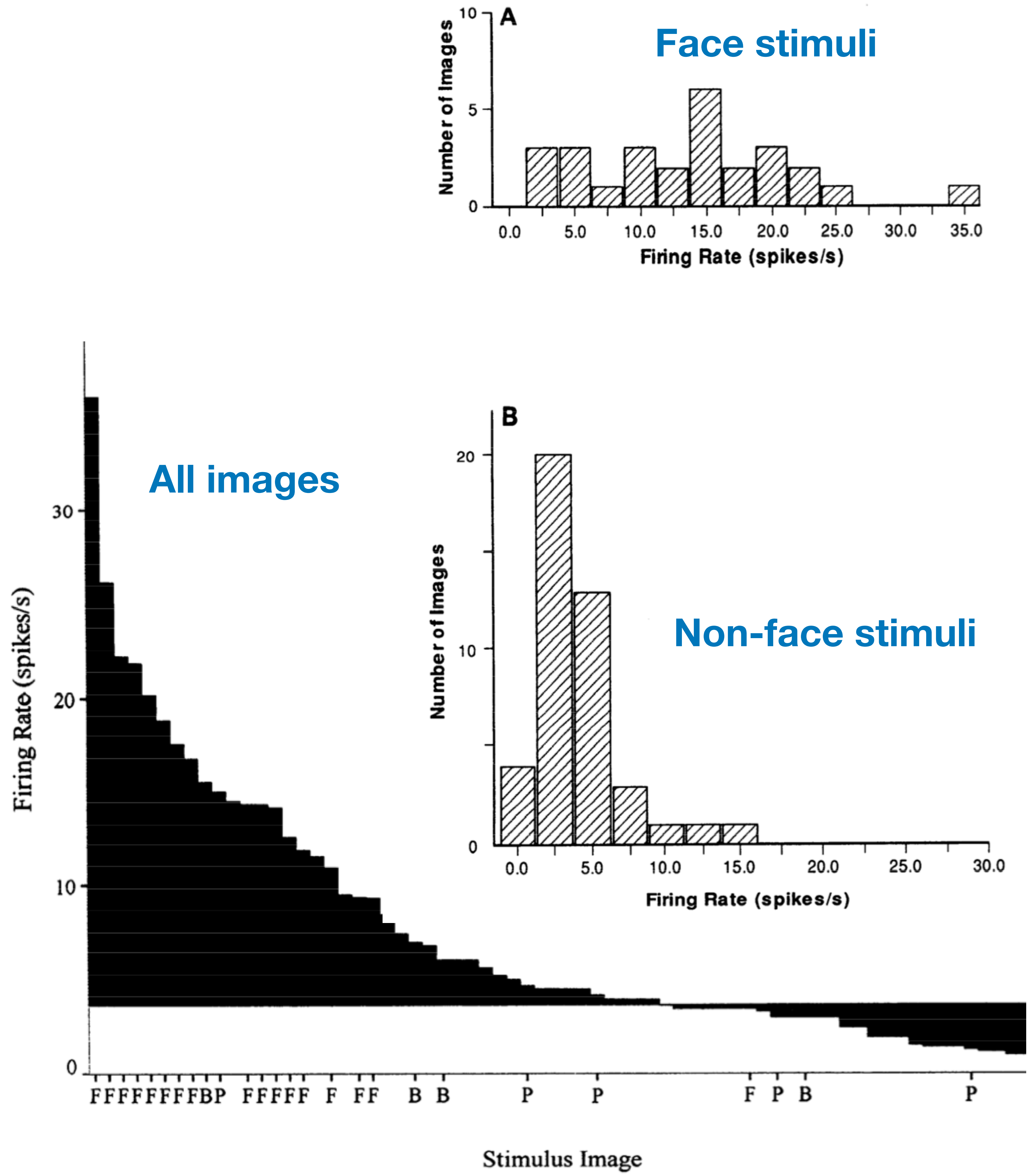


Representational theories

1. **Dense distributed coding:** Many neurons active for most stimuli
2. **Localist (grandmother-cell)** coding: One neuron per object
3. **Sparse distributed coding:** Few neurons active per stimulus

Results



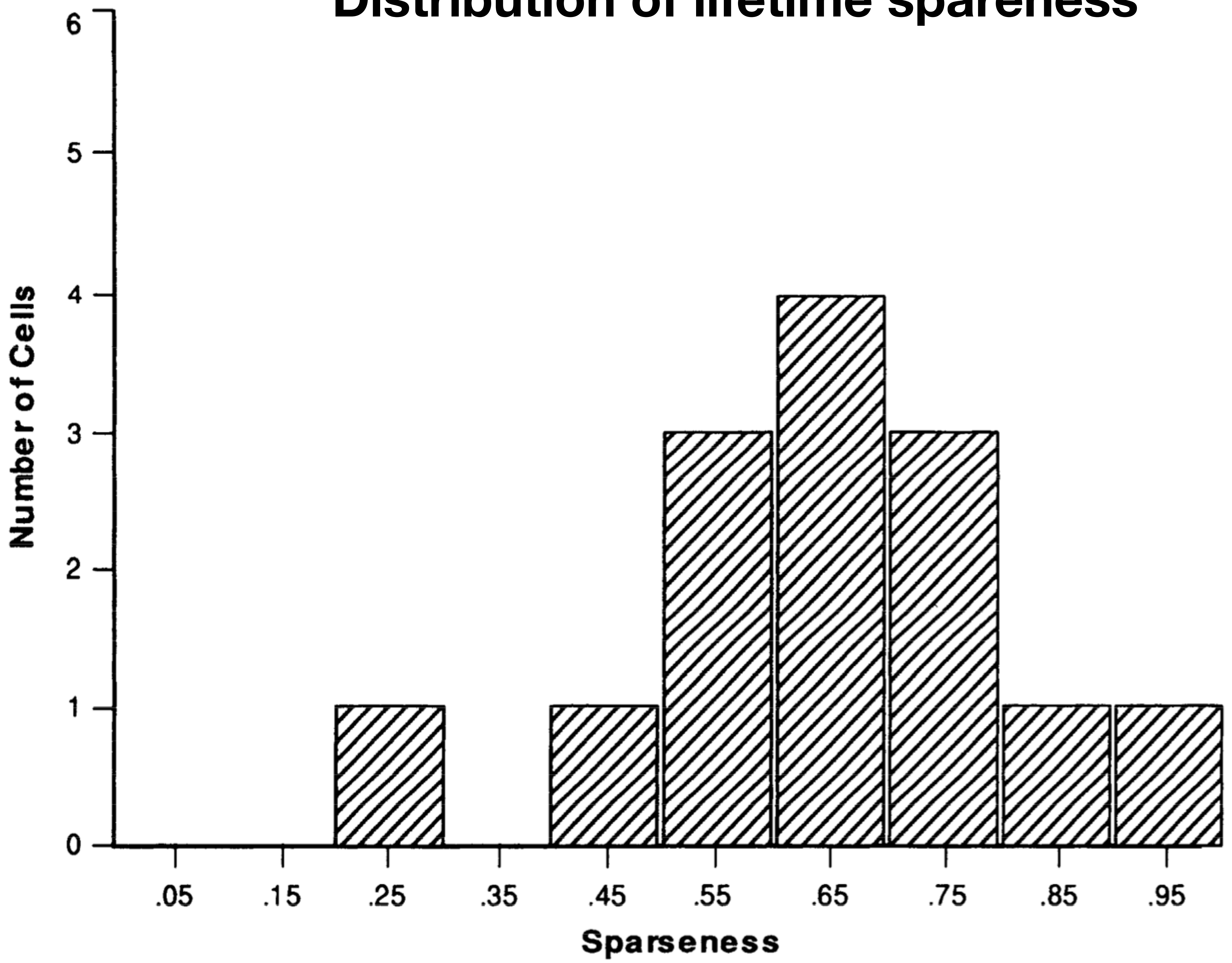


Comparing neural data with theory

- **Lifetime sparseness:** how selectively a *neuron* responds across a large set of different visual stimuli over its lifetime.

- “Which class of representations could plausibly generate these neural response statistics?”

Distribution of lifetime spareness



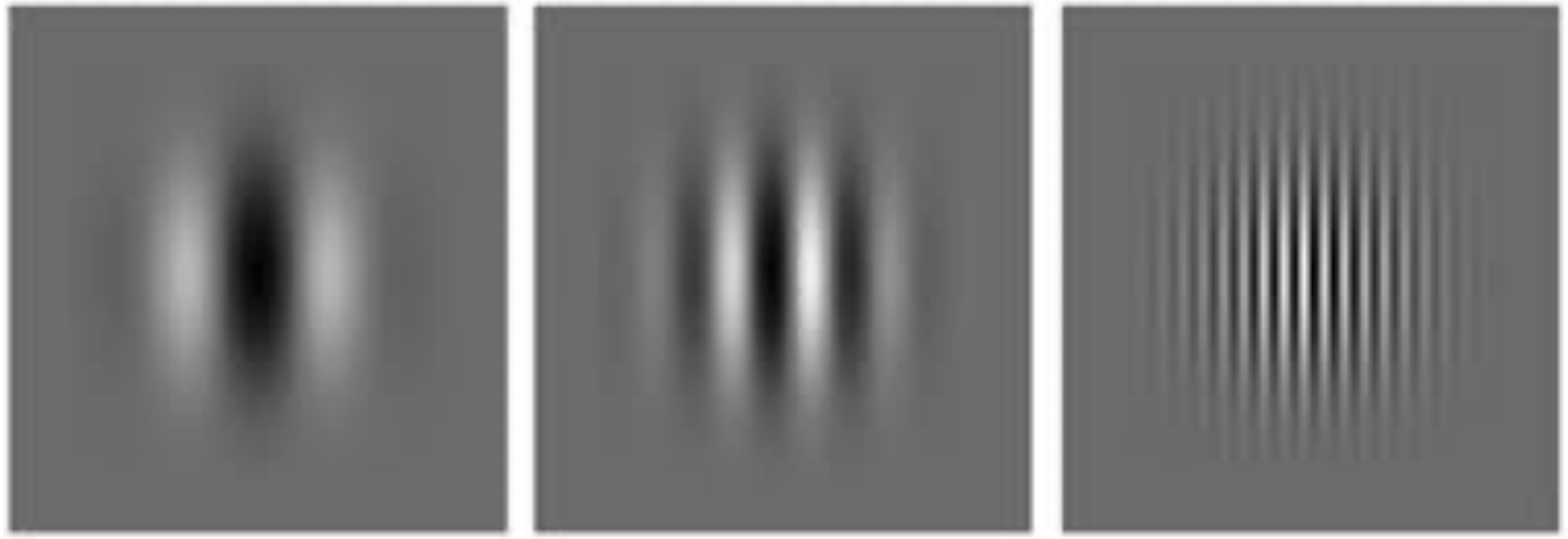
*"The mean response sparseness of **0.60** of this population of face-selective neurons indicates that, **within the class faces**, these neurons implement **distributed encoding**”*

Early day: Comparing response properties

Idea: compare tuning properties of cells with those of networks

De Valois et al. (1982):

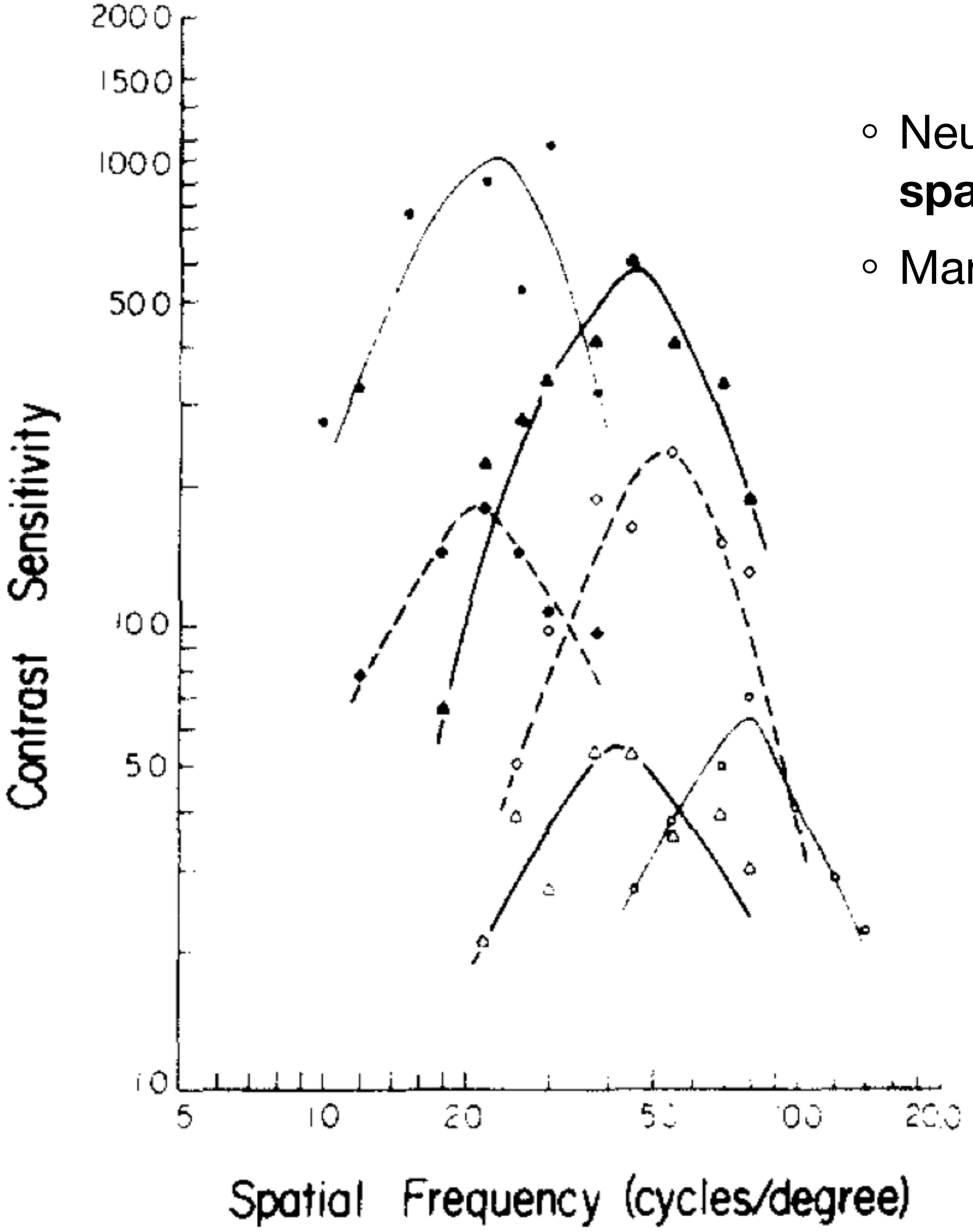
- How selectively do V1 neurons respond to different spatial frequencies in sine gratings? (Are V1 neurons bandpass filters?)



Neural data

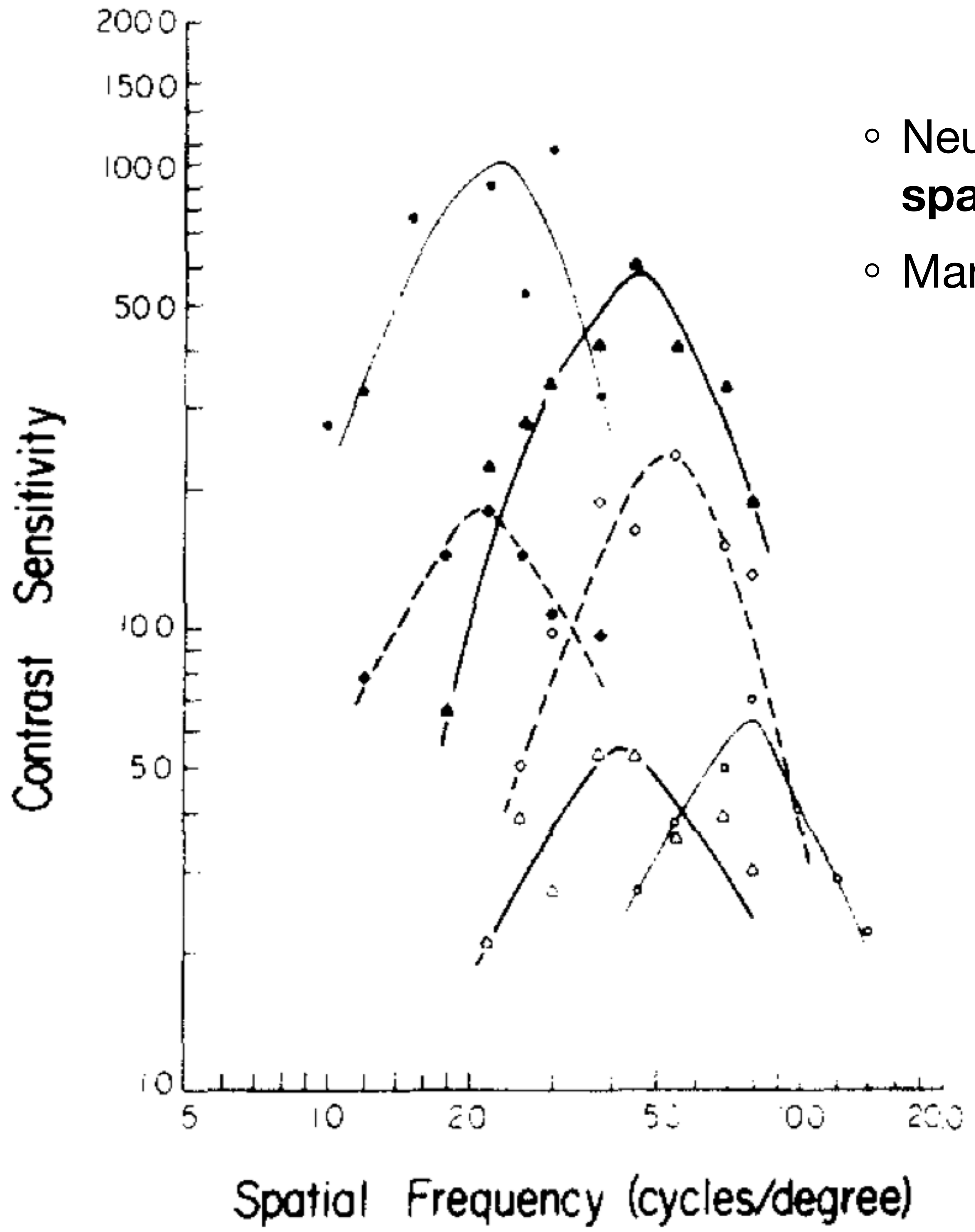
- single-unit recordings from macaque V1
- Present sinusoidal gratings at many orientations and spatial frequencies
- Spatial frequency = Number of cycles (dark-light)/ visual angle (degrees)

Comparing results to neural networks



- Neurons span a **wide range of preferred spatial frequencies**
- Many neurons are **narrowly tuned**

Comparing results to neural networks

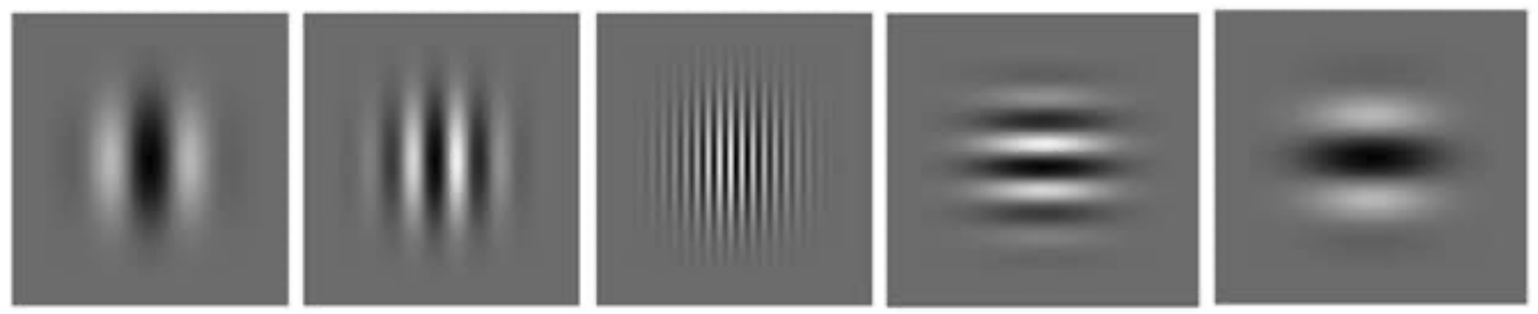


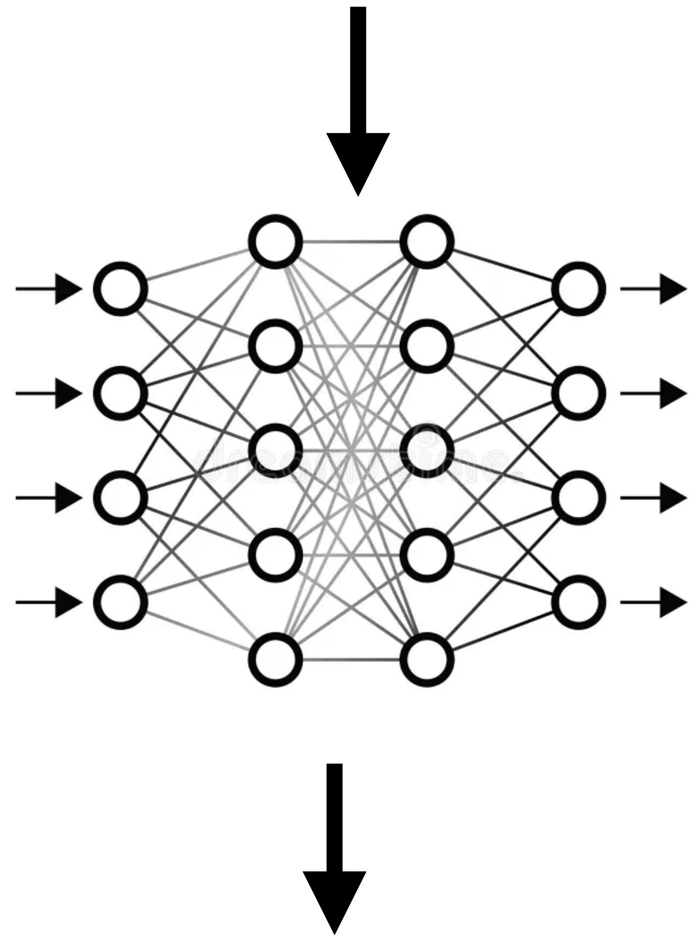
- Neurons span a **wide range of preferred spatial frequencies**
- Many neurons are **narrowly tuned**

SF1 SF2 SF3 . . .

u1
u2
u3
.
.
.

Input the same grating stimuli to a trained model

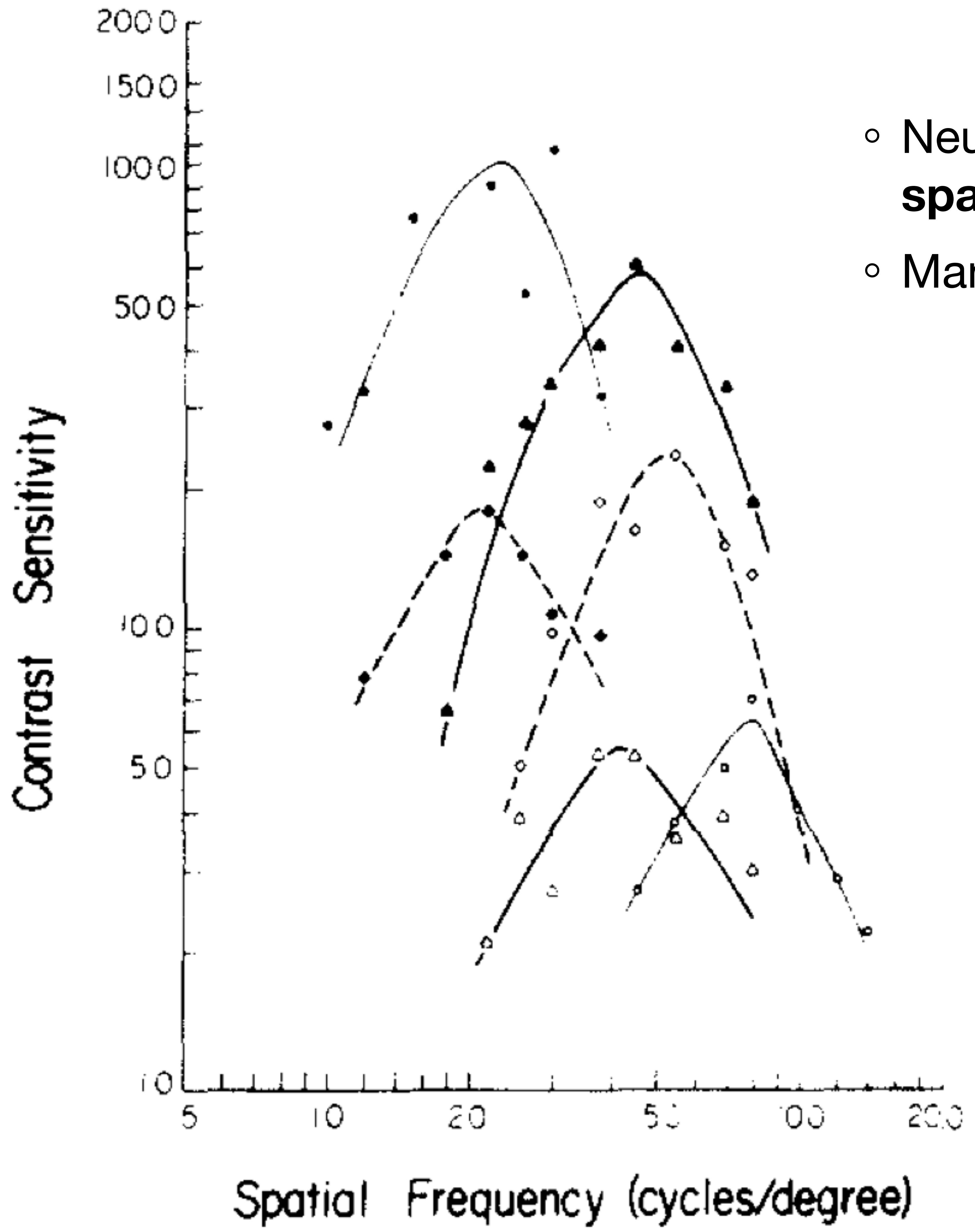




SF1 SF2 SF3 . . .

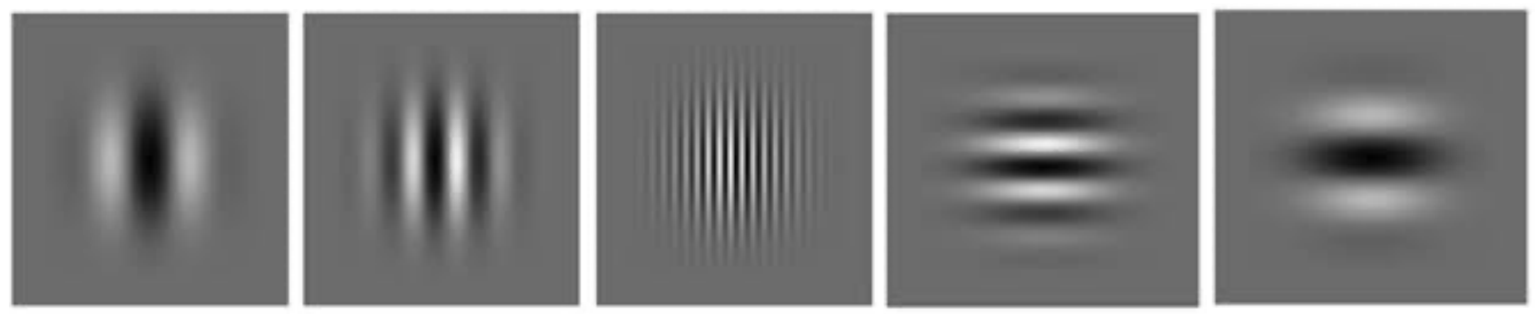
u1
u2
u3
.
.
.

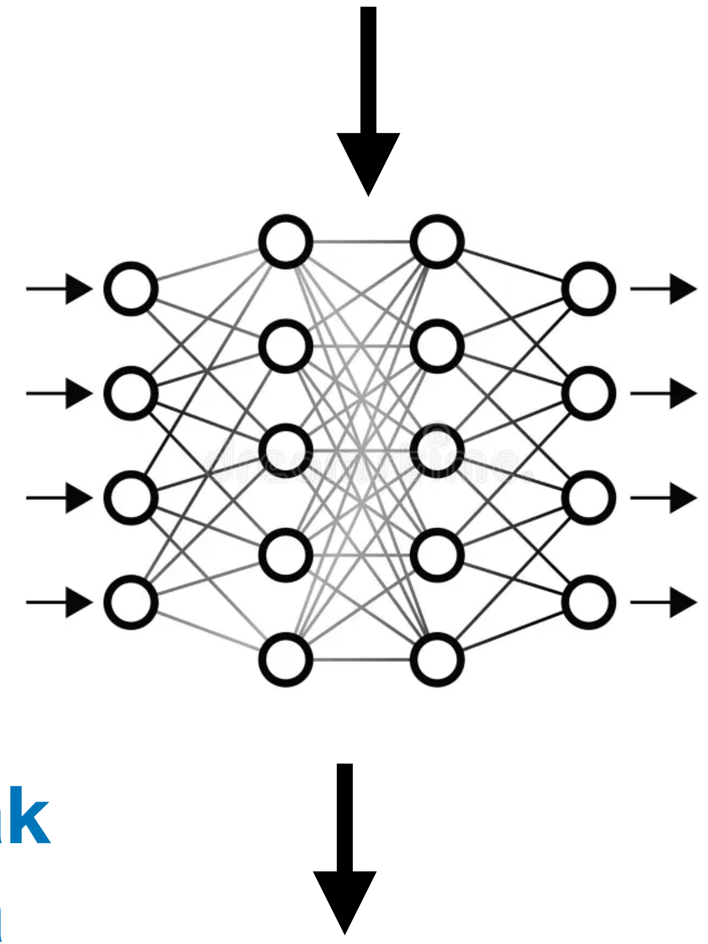
Comparing results to neural networks



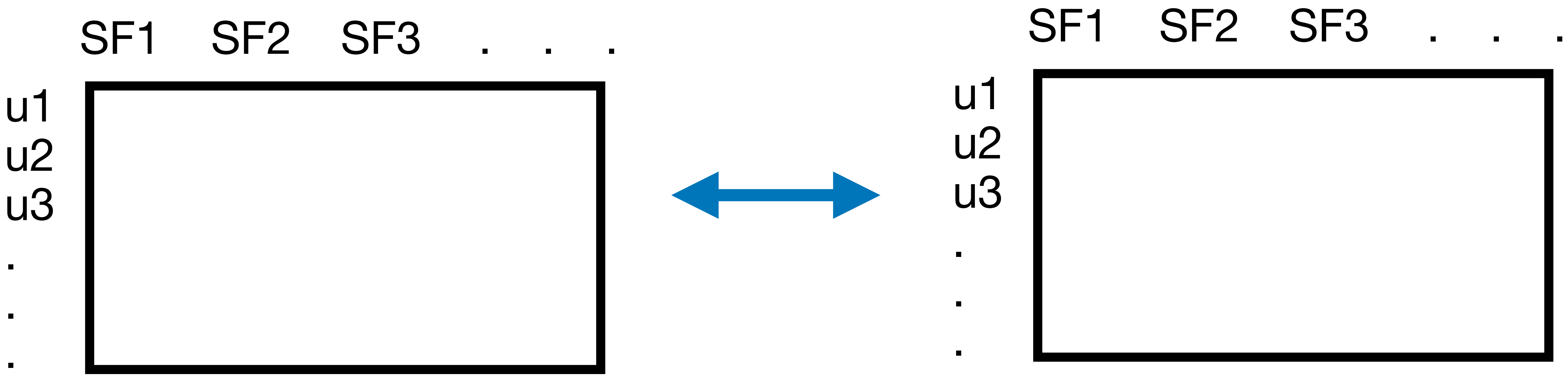
- Neurons span a **wide range of preferred spatial frequencies**
- Many neurons are **narrowly tuned**

Input the same grating stimuli to a trained model





Compare distributions of peak SF in model and neural data



Brain score platform

How to use

```
from brainscore_vision import load_benchmark
benchmark = load_benchmark("Marques2020_DeValois1982-peak_sf")
score = benchmark(my_model)
```

Benchmark API

Code examples

Data: Marques2020_DeValois1982

Find on GitHub

Metric: peak_sf

Find on GitHub

Model scores

Score Legend

Min Alignment — Max Alignment

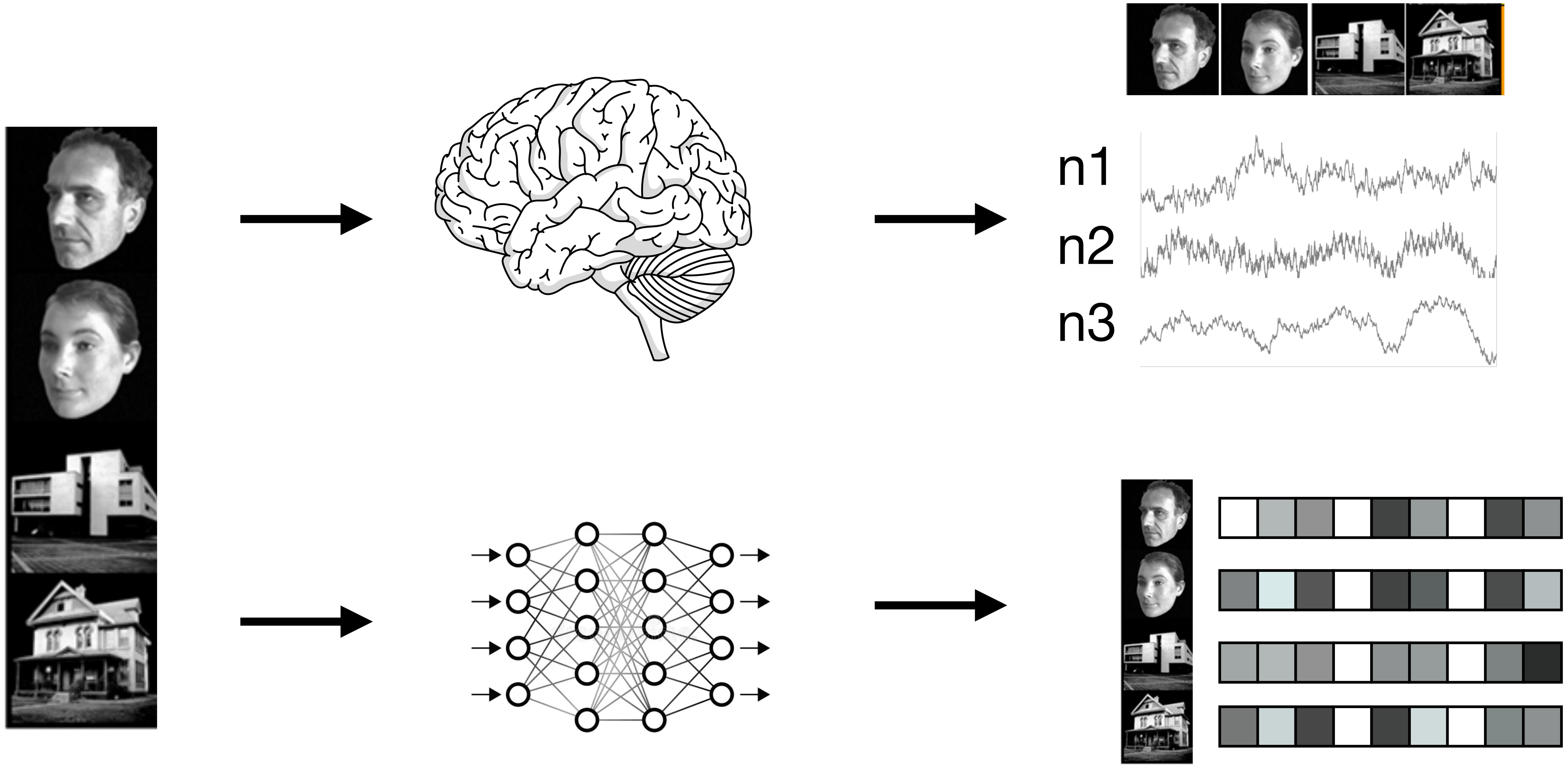
Rank	Model	Score
1	resnet-18-LC_w_sh_100_iter_m	.964
2	resnet50_imagenet_10_seed-0	.950
3	alexnet_training_seed_01	.943
4	resnet-18-LC_m	.941
5	resnet50_linf_4_robust	.935
6	alexnet_training_seed_08	.933

www.brain-score.org

2. Using stimulus-by-stimulus similarity matrices

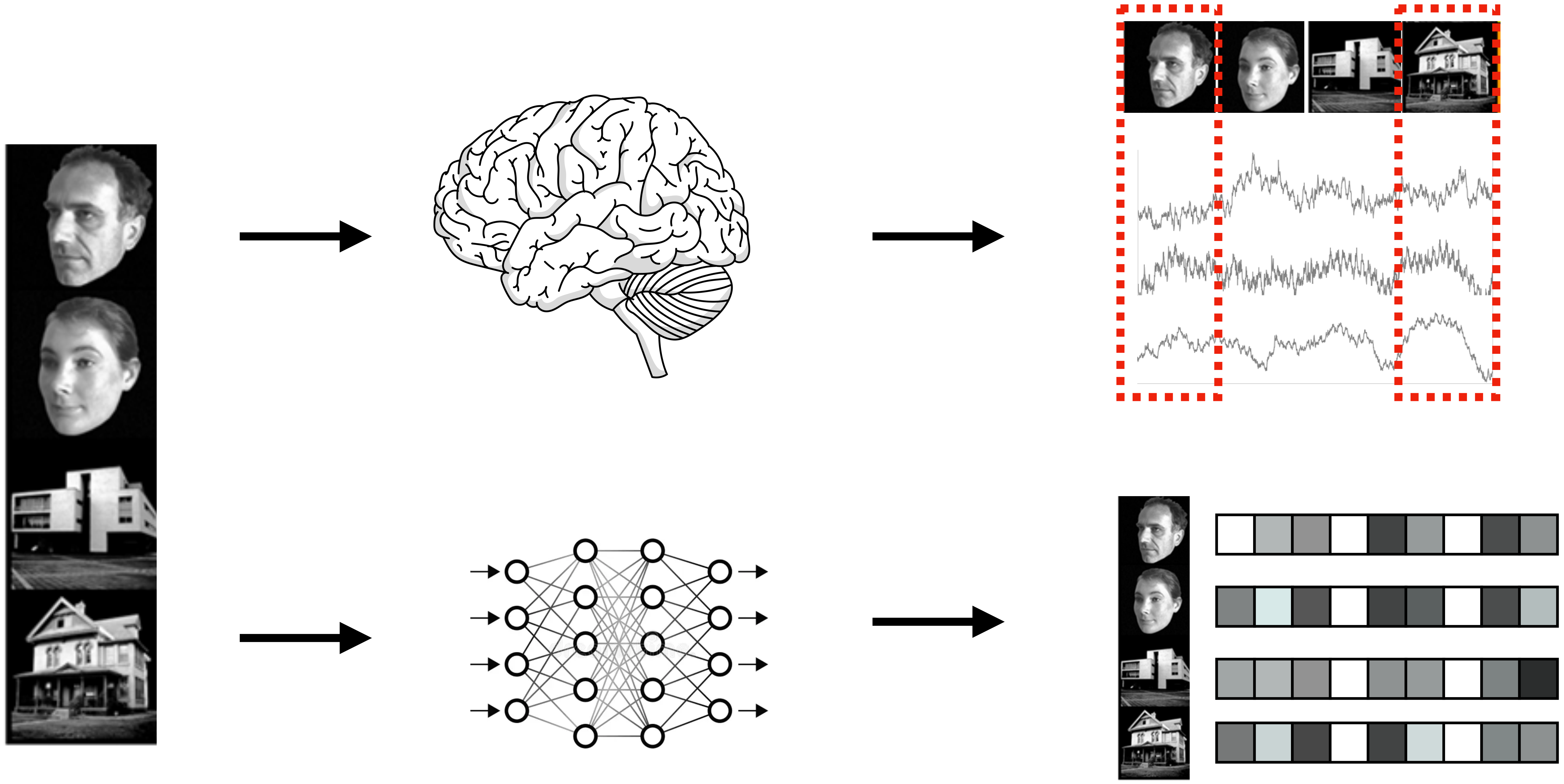
❖ Compare representations via stimulus–stimulus relationships

❖ Ignore neuron-to-neuron correspondence entirely



2. Using stimulus-by-stimulus similarity matrices

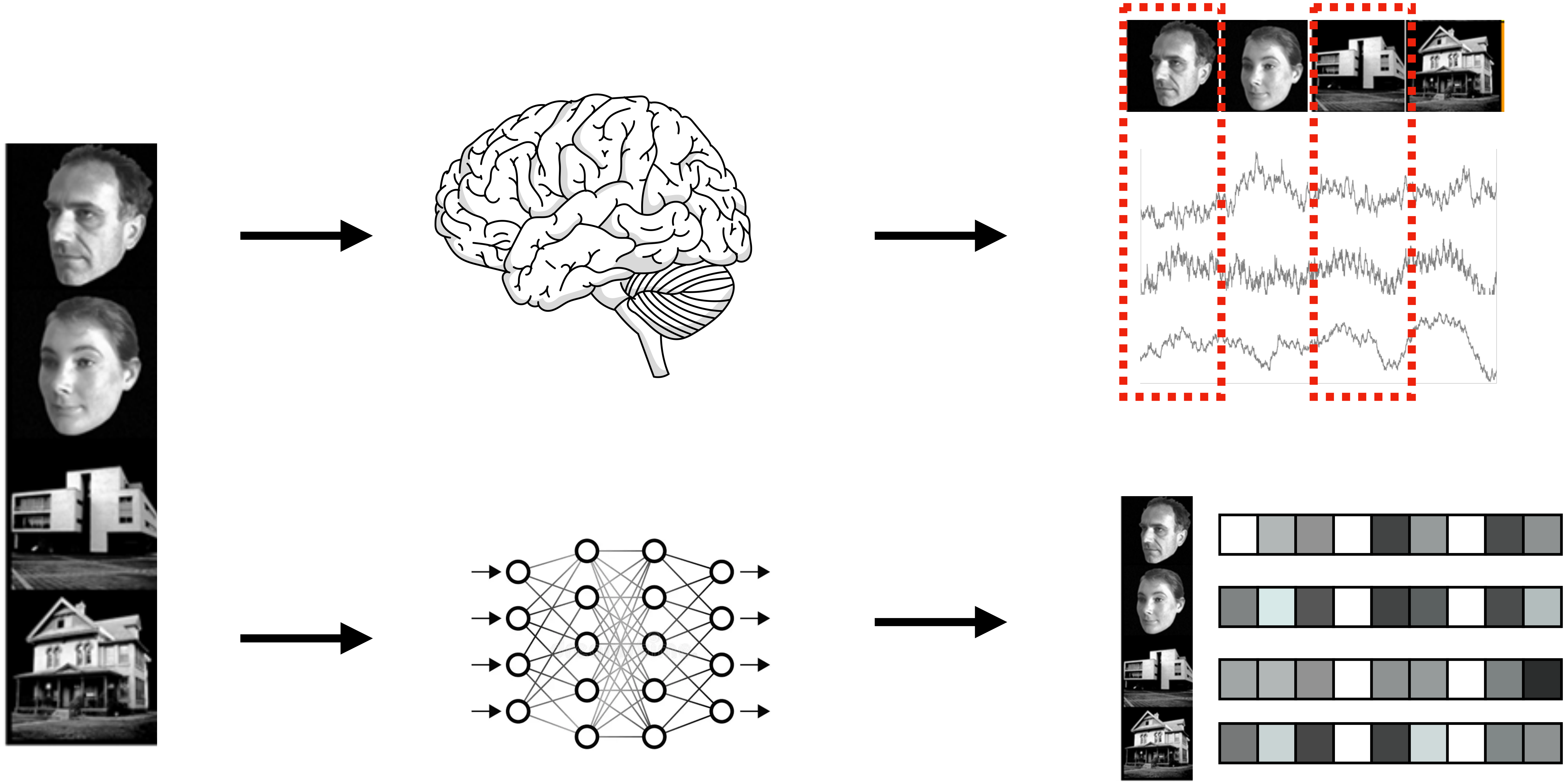
- ❖ Compare representations via stimulus–stimulus relationships
- ❖ Ignore neuron-to-neuron correspondence entirely



2. Using stimulus-by-stimulus similarity matrices

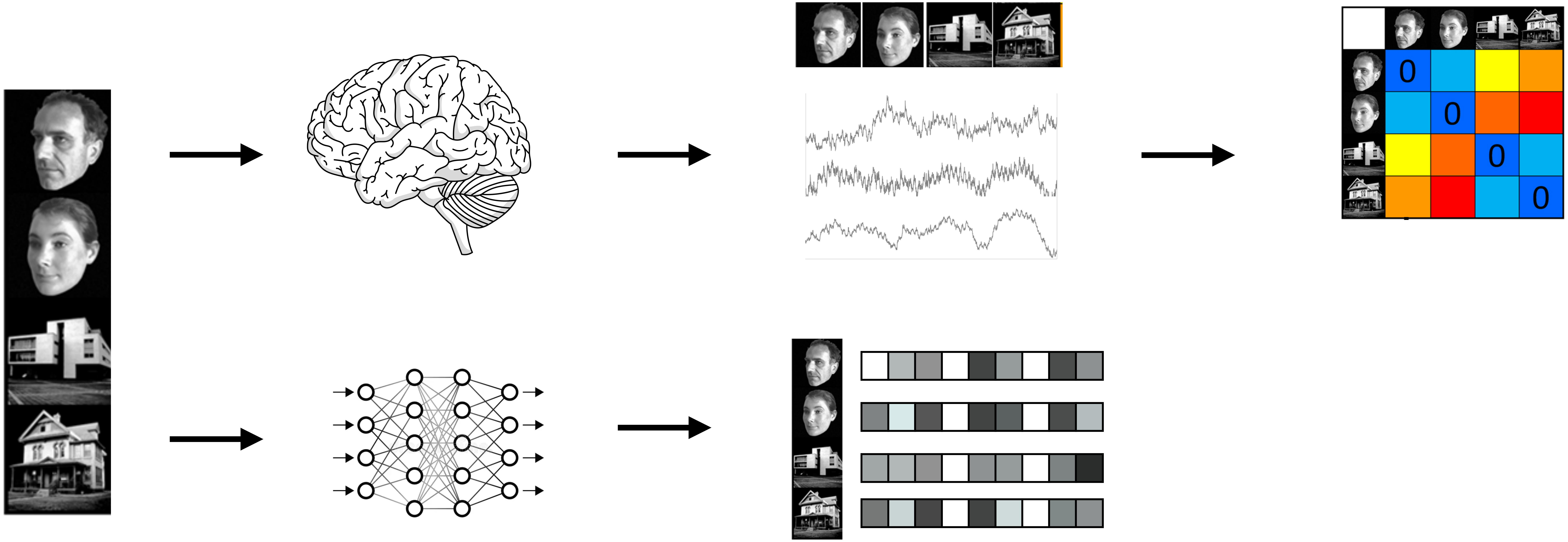
❖ Compare representations via stimulus–stimulus relationships

❖ Ignore neuron-to-neuron correspondence entirely



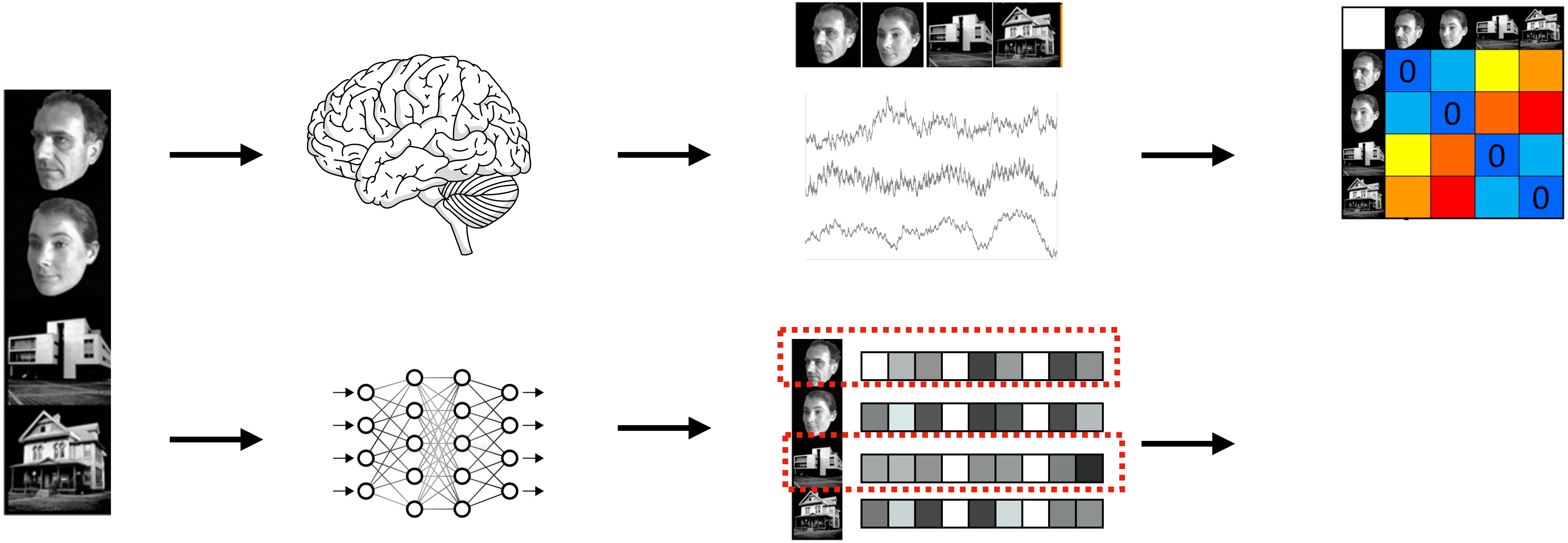
2. Using stimulus-by-stimulus similarity matrices

- Compare representations via stimulus–stimulus relationships
- Ignore neuron-to-neuron correspondence entirely



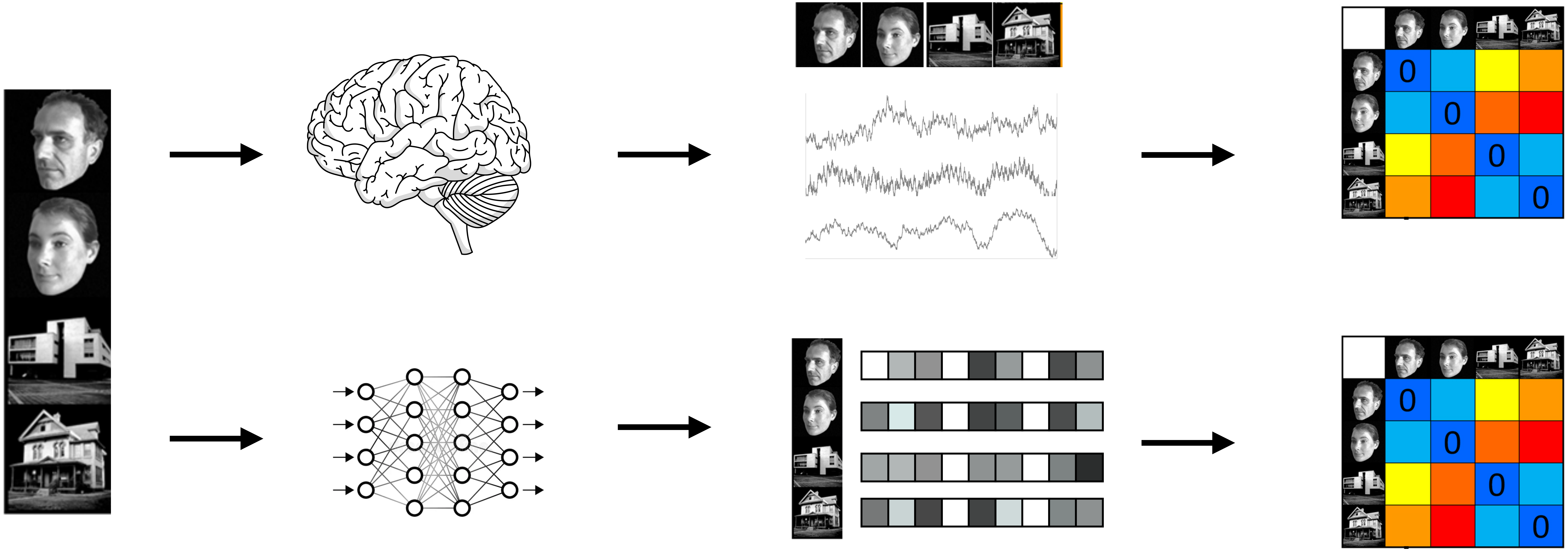
2. Using stimulus-by-stimulus similarity matrices

- ❖ Compare representations via stimulus–stimulus relationships
- ❖ Ignore neuron-to-neuron correspondence entirely



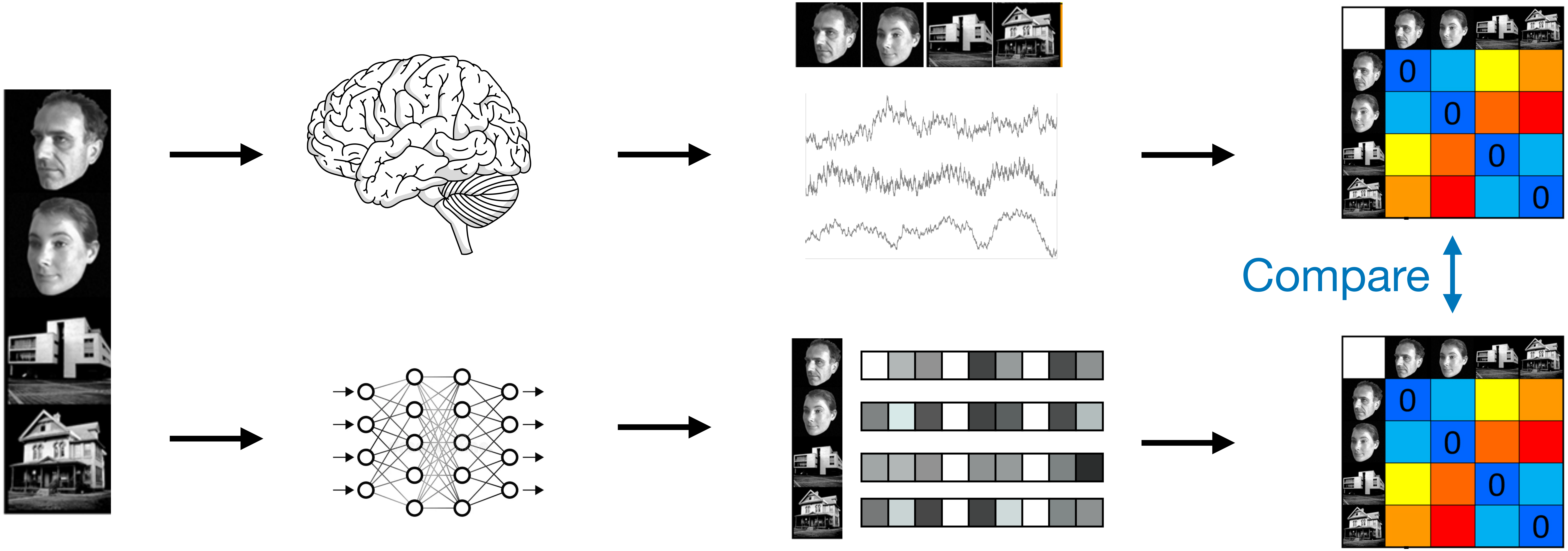
2. Using stimulus-by-stimulus similarity matrices

- Compare representations via stimulus–stimulus relationships
- Ignore neuron-to-neuron correspondence entirely

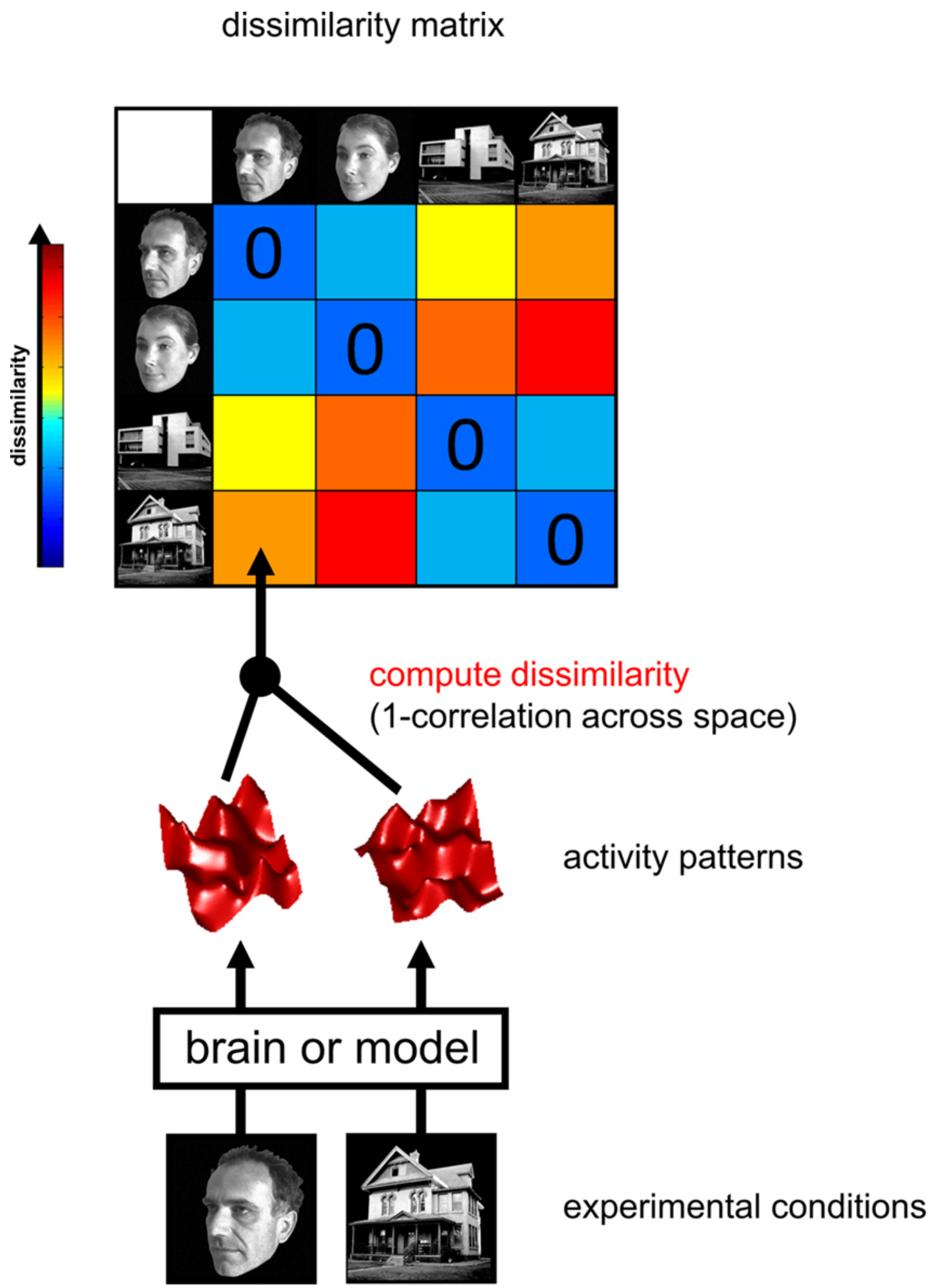


2. Using stimulus-by-stimulus similarity matrices

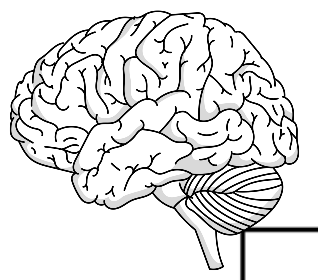
- Compare representations via stimulus–stimulus relationships
- Ignore neuron-to-neuron correspondence entirely



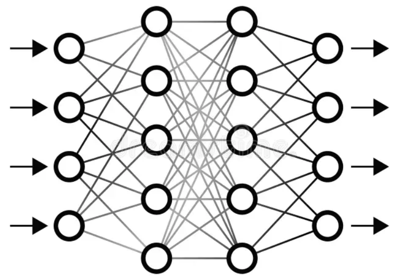
Example: Representational Similarity Analysis (RSA)

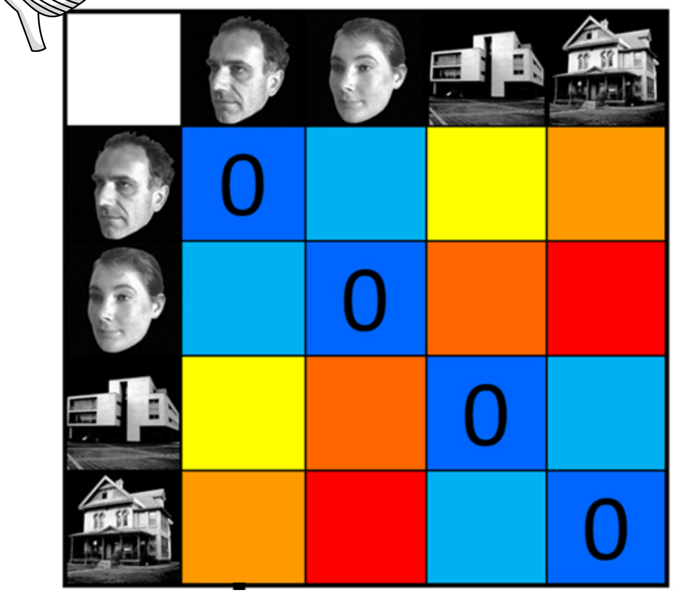


- **“Do two systems organize stimuli in the same geometric way?**
- **Pros:** For the first time allowed comparison between any systems as long as the stimuli was the same. Also doesn’t need any training params.
- **Cons:** Very similar systems (up to a linear transform) can look very different under RSA



$$\mathrm{RSA}(X, Y) = \mathrm{corr}(\mathbf{r}_X, \mathbf{r}_Y)$$



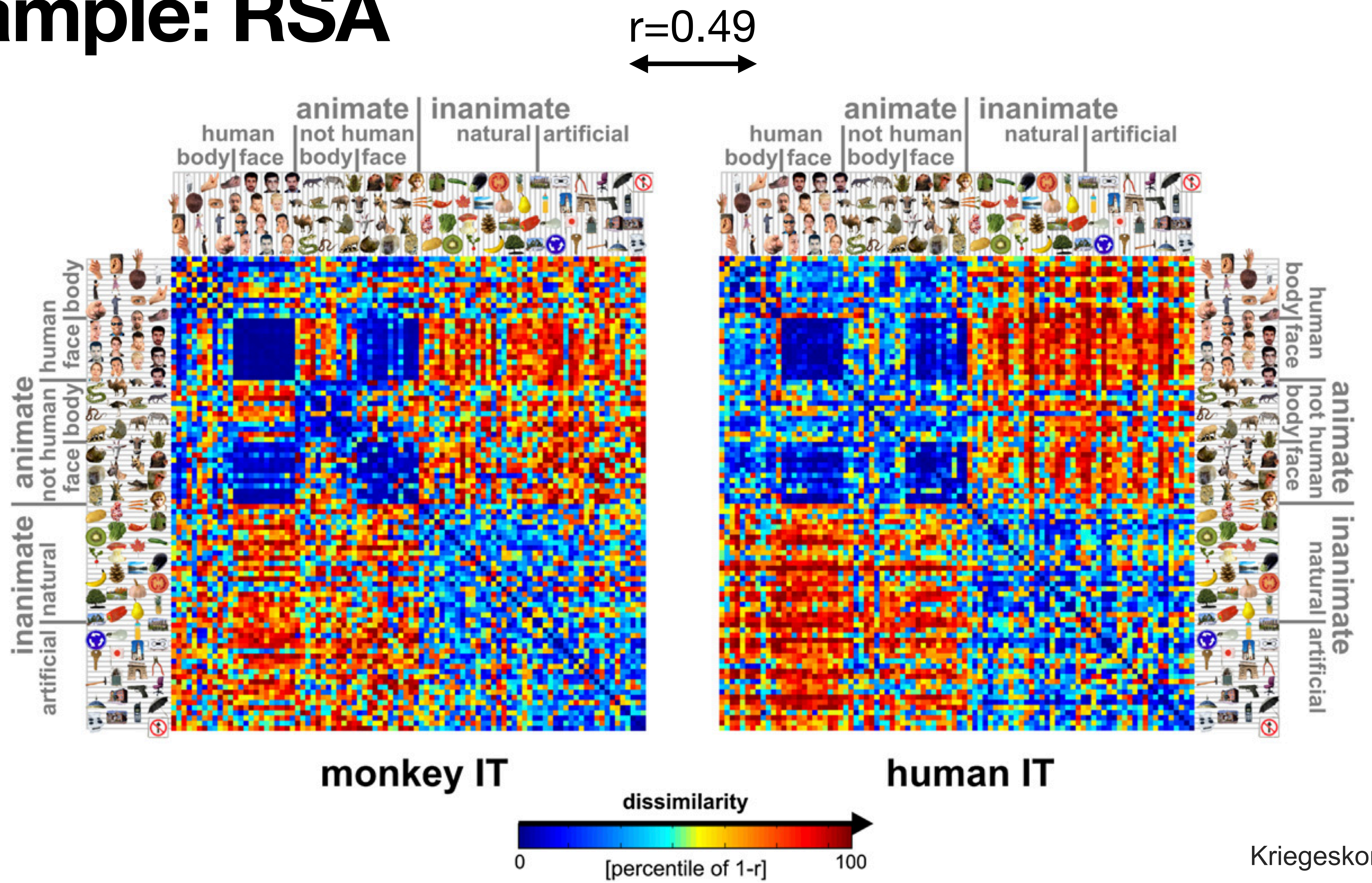


RDM_X

RDM_Y

Kriegeskorte et al, 2008

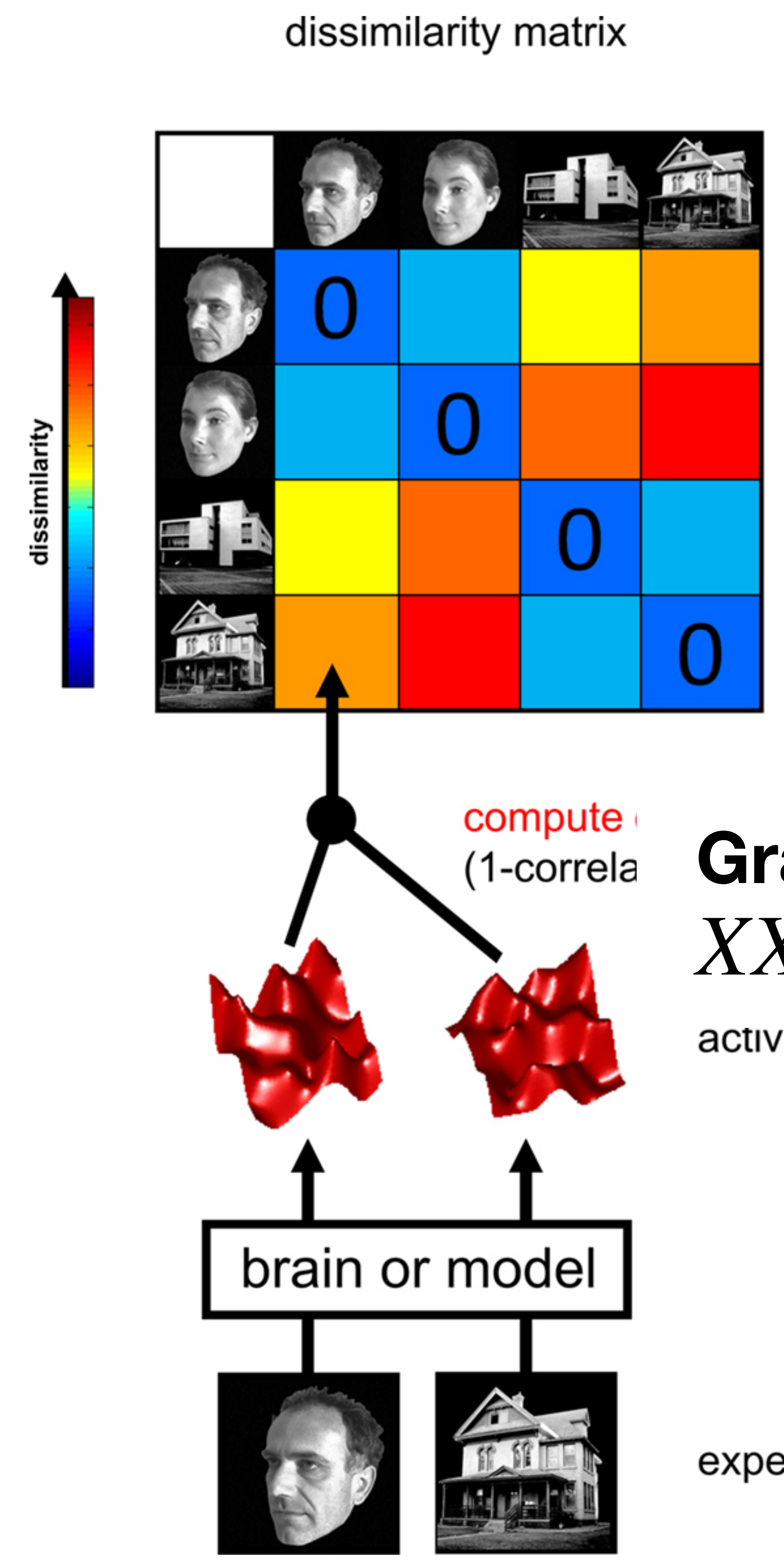
Example: RSA



Kriegeskorte et al, 2008

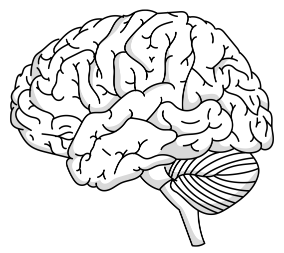
Example: Centered Kernel Alignment (CKA)

- Similar method to RSA, but operates on **similarities** rather than **distance**
- Computes cosine similarity between centered gram matrices
- **Pros:** more flexible than RSA, invariant to rotation, scaling
- **Cons:** Similar systems (up to a linear transform) can look different under CKA.



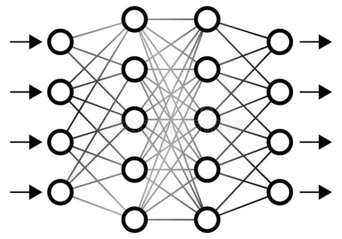
Gram matrix for X is: $XX^\top \in \mathbb{R}^{n \times n}$

$$\mathrm{CKA}(X, Y) = \cos\big(\angle(K_X, K_Y)\big)$$



$K_X = CXX^\top C$

$K_Y = CYY^\top C$

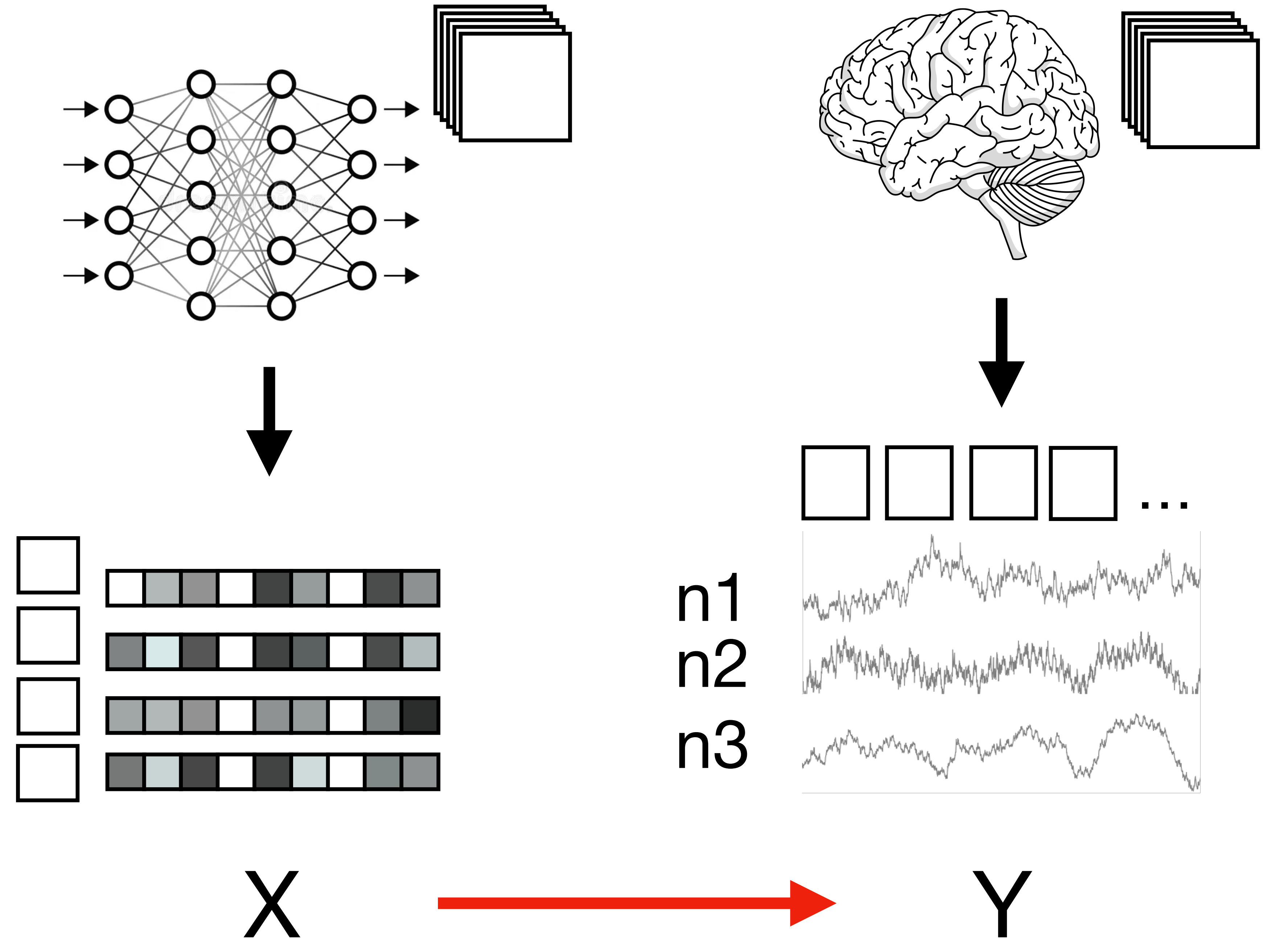


3. Learning mappings from the model to the brain

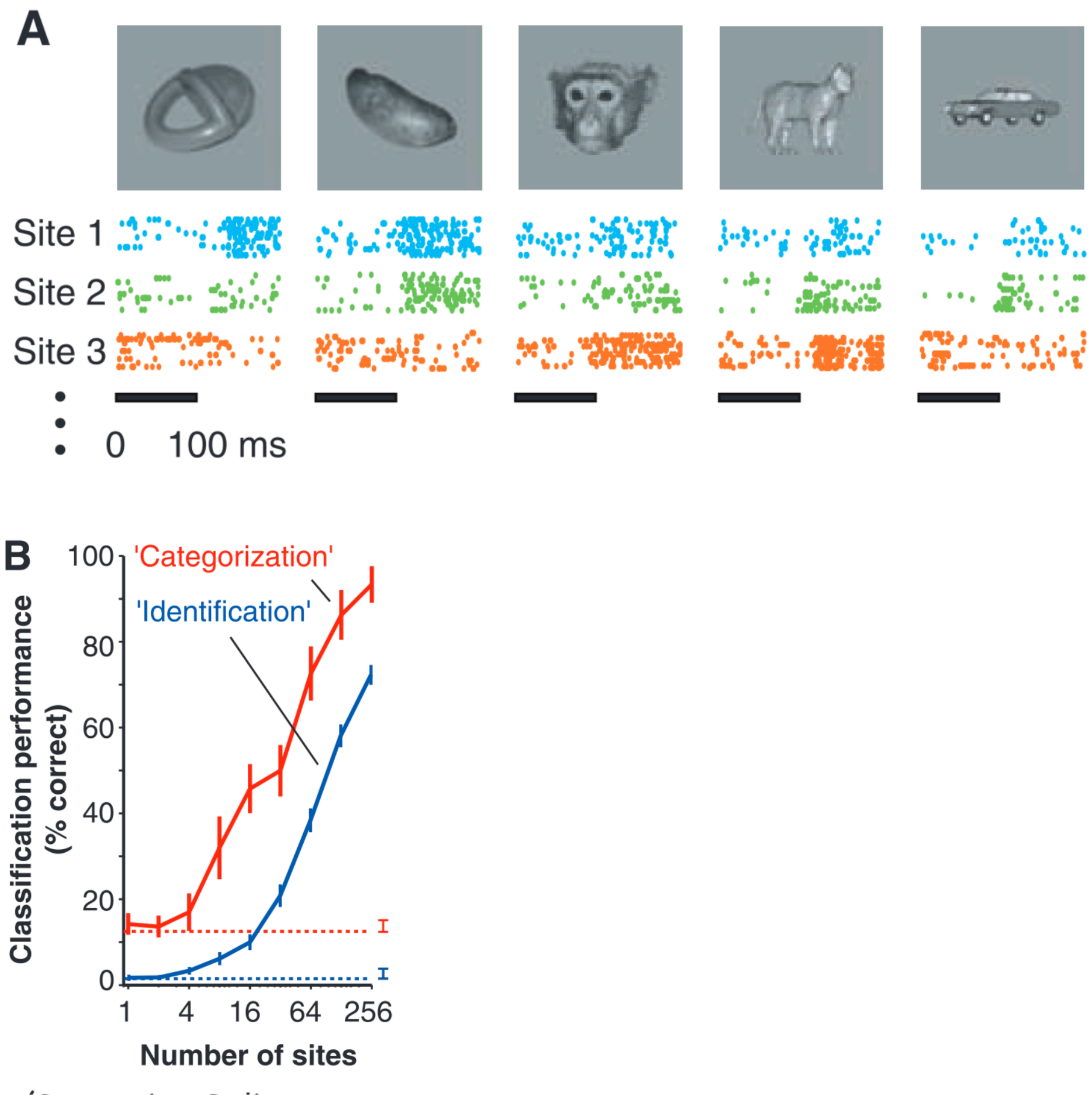
$$X \in \mathbb{R}^{n \times N_x} \quad , Y \in \mathbb{R}^{n \times N_y}$$

Rows = stimuli, Columns = neurons / features

- Most methods focus on learning a linear mapping
- Let X be the model representations, and Y the neural responses to the same set of stimuli.
- **Goal:** Find the best mapping from X to Y.

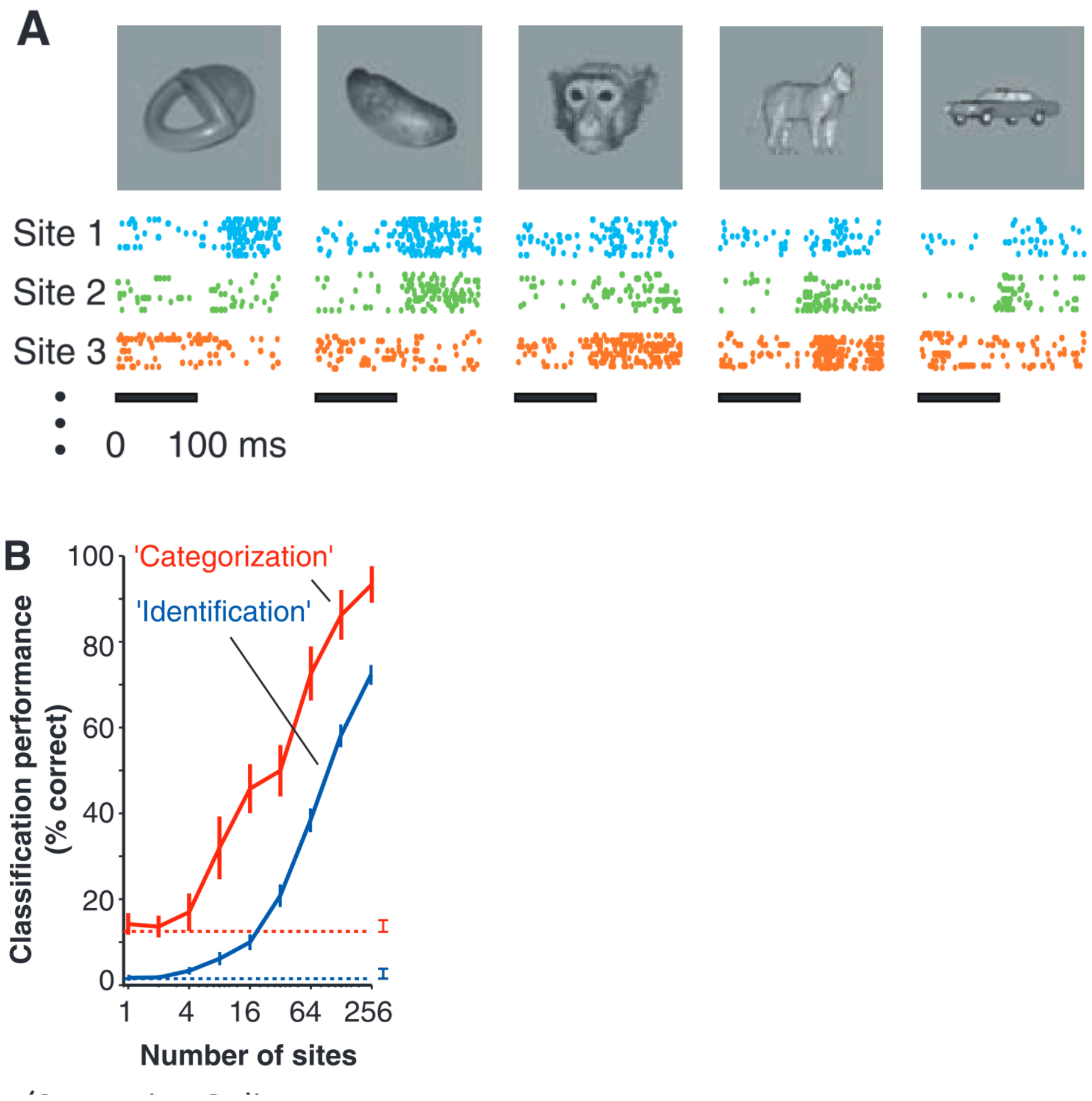


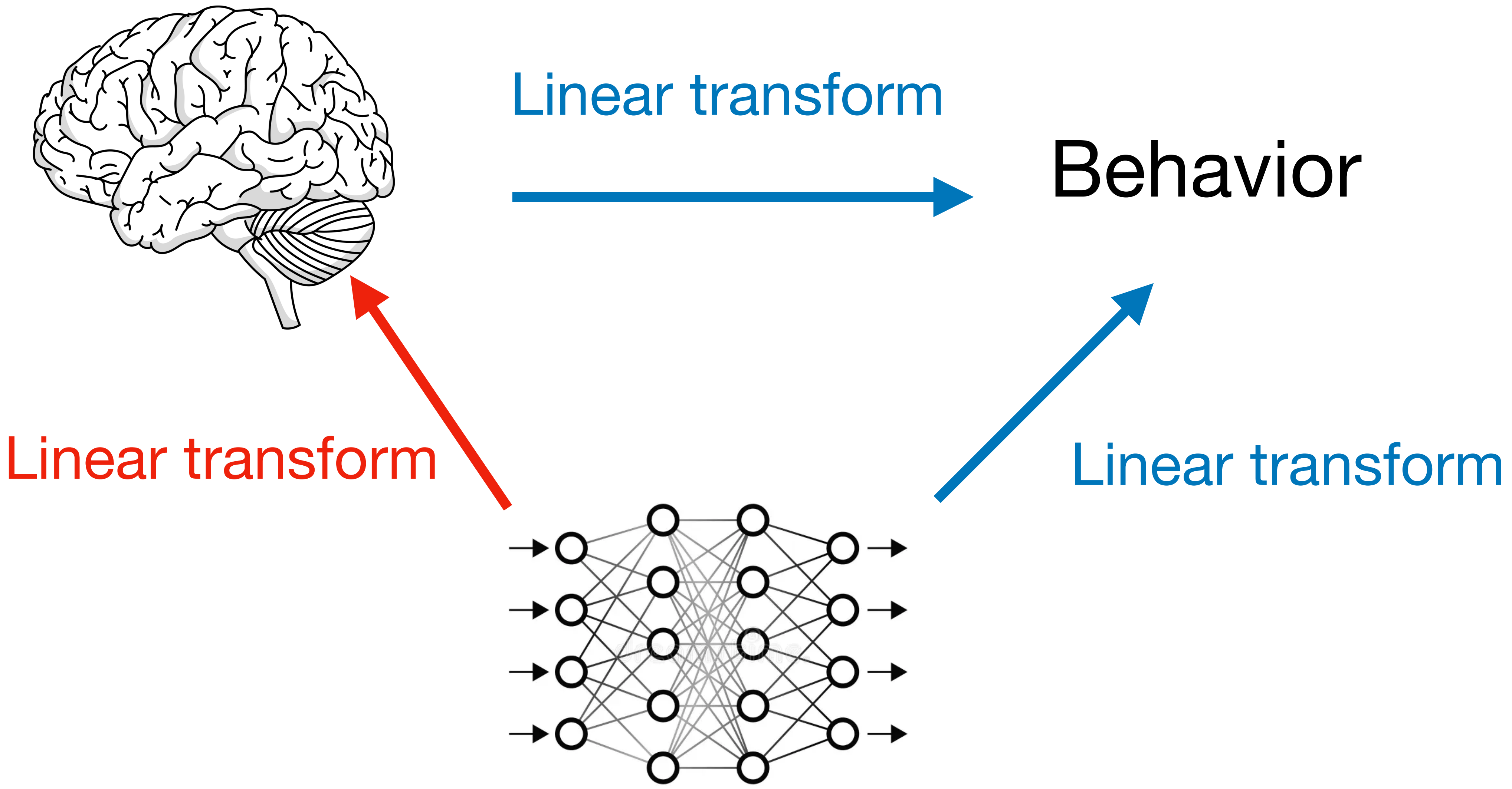
Why Linear Mapping?



Hung et al, 2005

Why Linear Mapping?

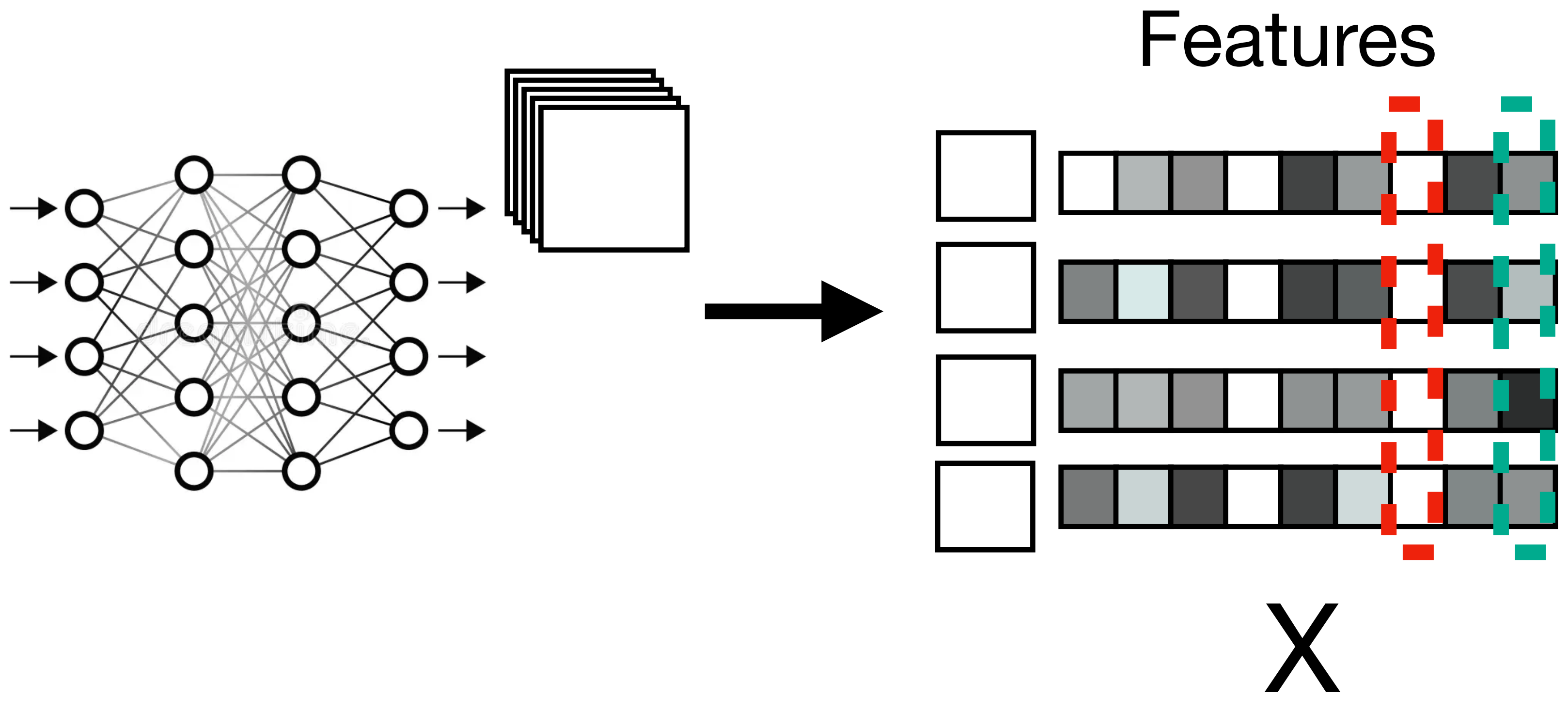


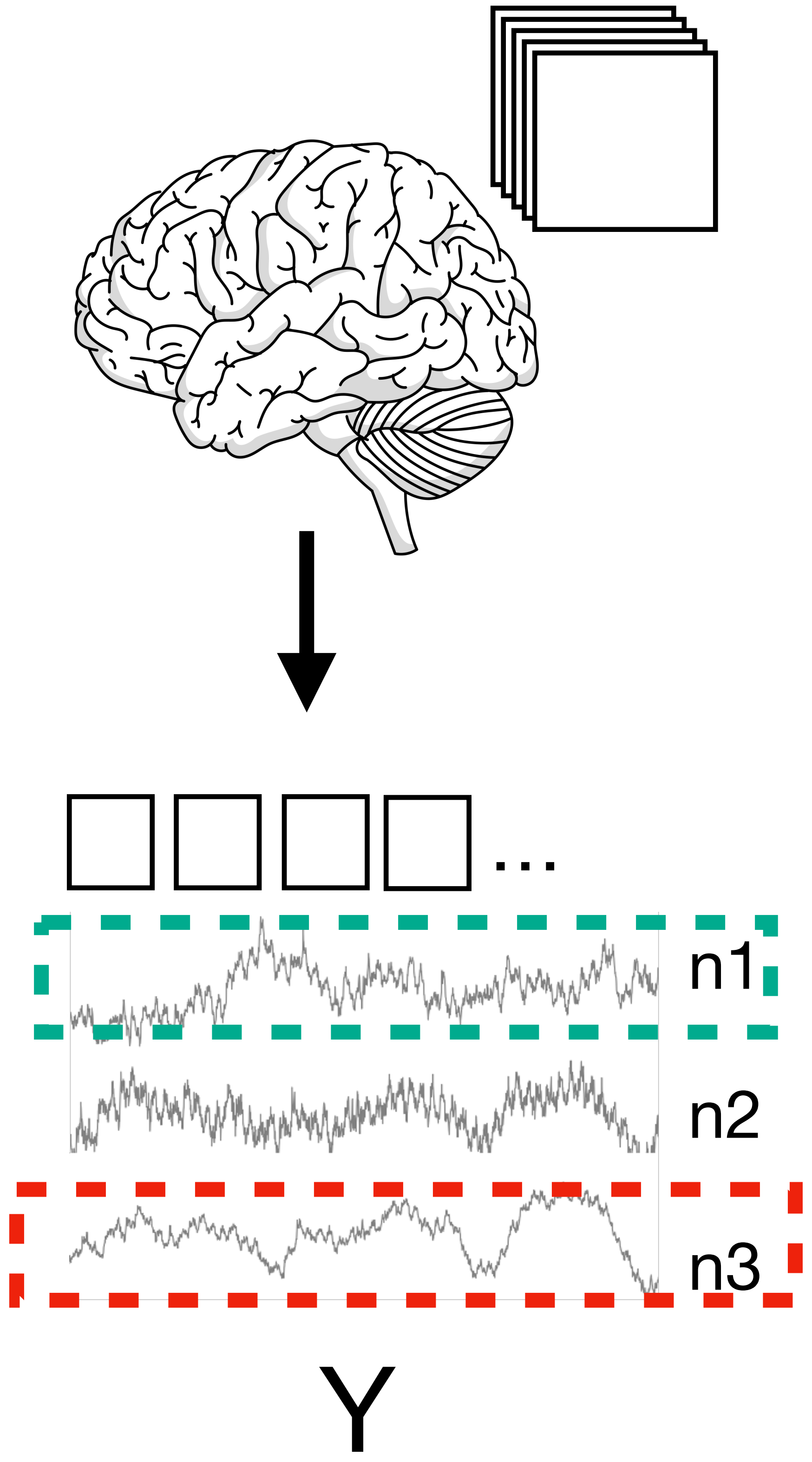


Hung et al, 2005

Example: One to one mapping

Idea: for 2 systems to be similar their parts should be similar



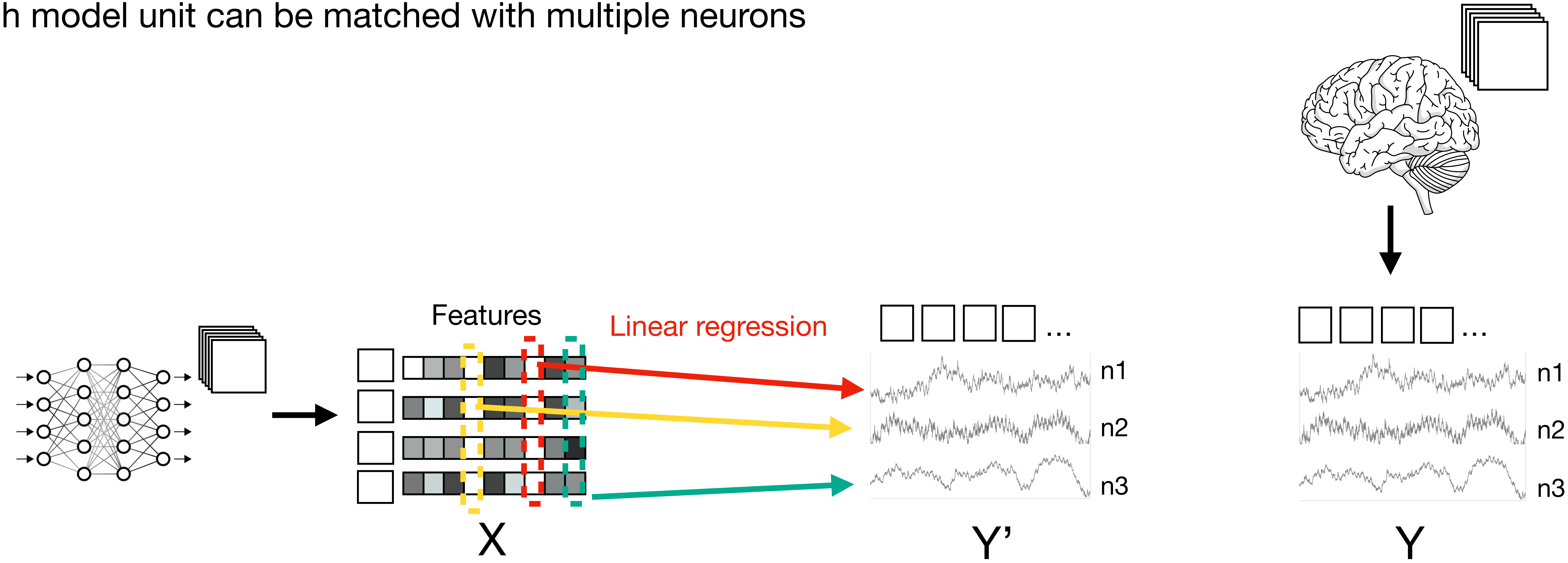


Example: One to one mapping

Idea: for 2 systems to be similar their parts should be similar

- For each neuron, find the model unit with the highest correlation, then compute the optimal linear mapping from neuron to model unit

- Each model unit can be matched with multiple neurons



Example: One to one mapping

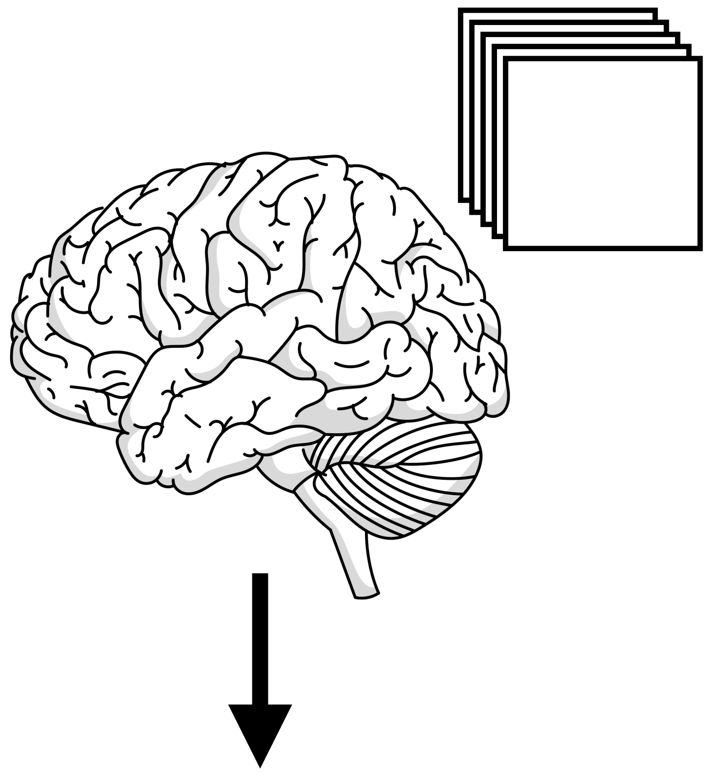
Idea: for 2 systems to be similar their parts should be similar

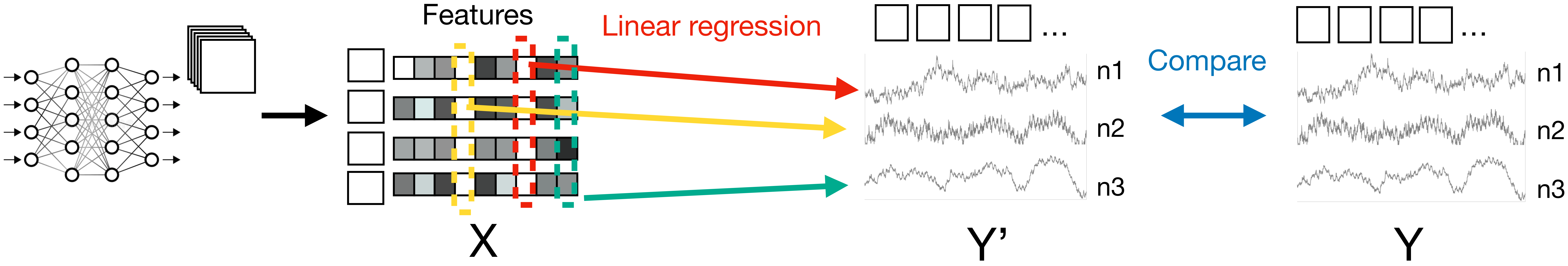
- ❖ For each neuron, find the model unit with the highest correlation, then compute the optimal linear mapping from neuron to model unit

- ❖ Each model unit can be matched with multiple neurons

- ❖ **Pros:** Simple and strict, effective for comparing very similar regions where parts of the system are consistent across individuals (ex: retina)

- ❖ **Cons:** Most brain areas don't have the exact same units in different subjects (ex: IT)

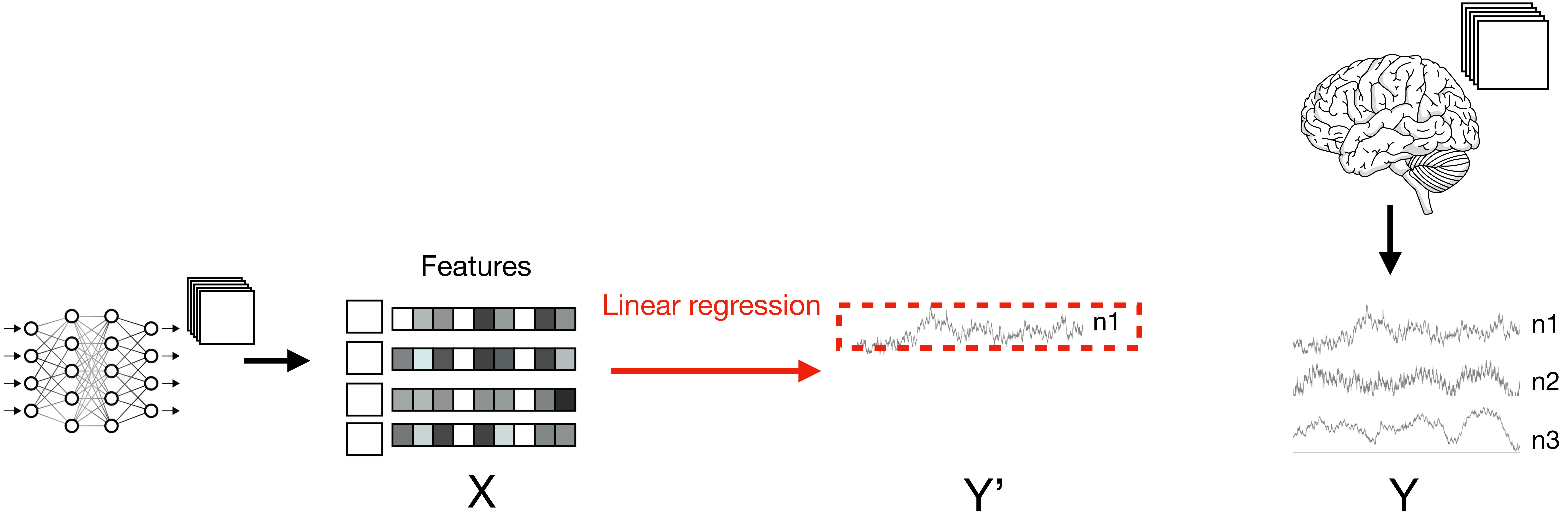




Example: Linear Regression

Idea: Find linear combinations of model units that together produce a 'synthetic neuron'

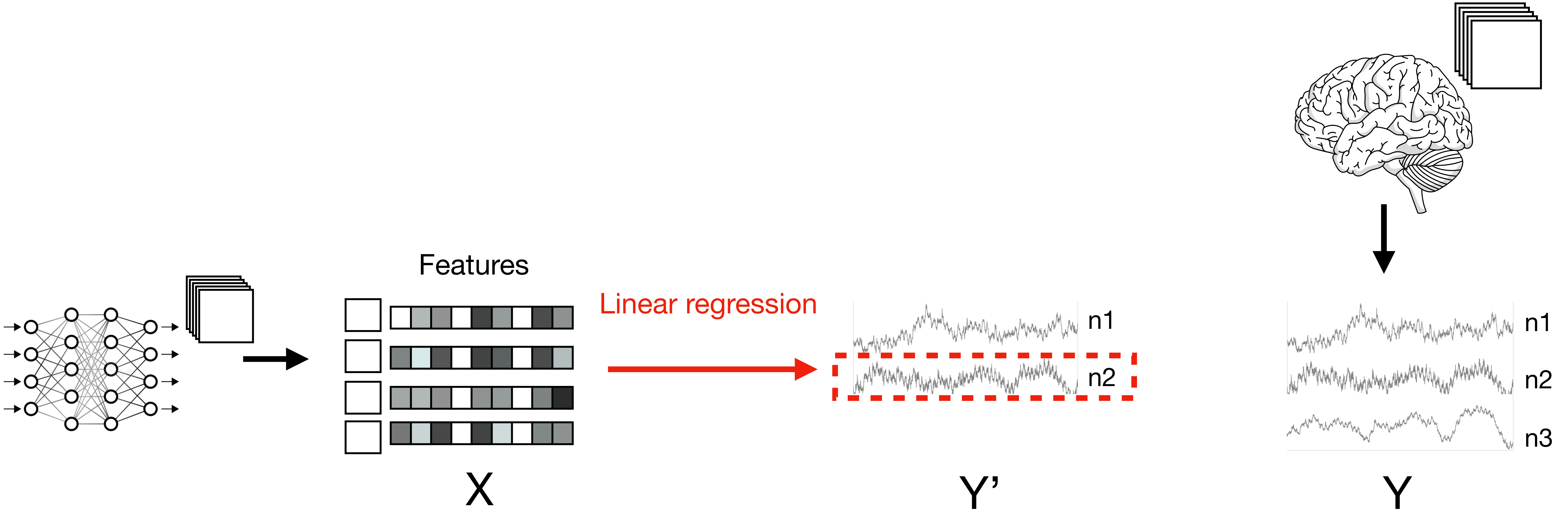
❖ Learn a mapping from all model units to each target neuron.



Example: Linear Regression

Idea: Find linear combinations of model units that together produce a 'synthetic neuron'

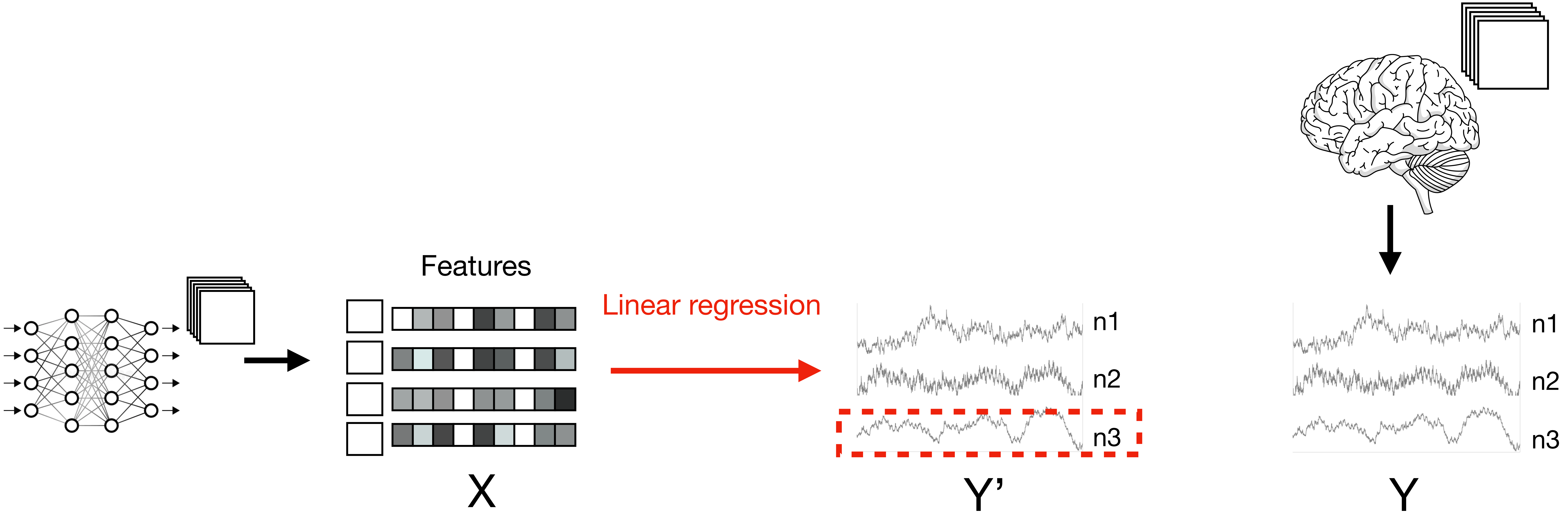
❖ Learn a mapping from all model units to each target neuron.



Example: Linear Regression

Idea: Find linear combinations of model units that together produce a 'synthetic neuron'

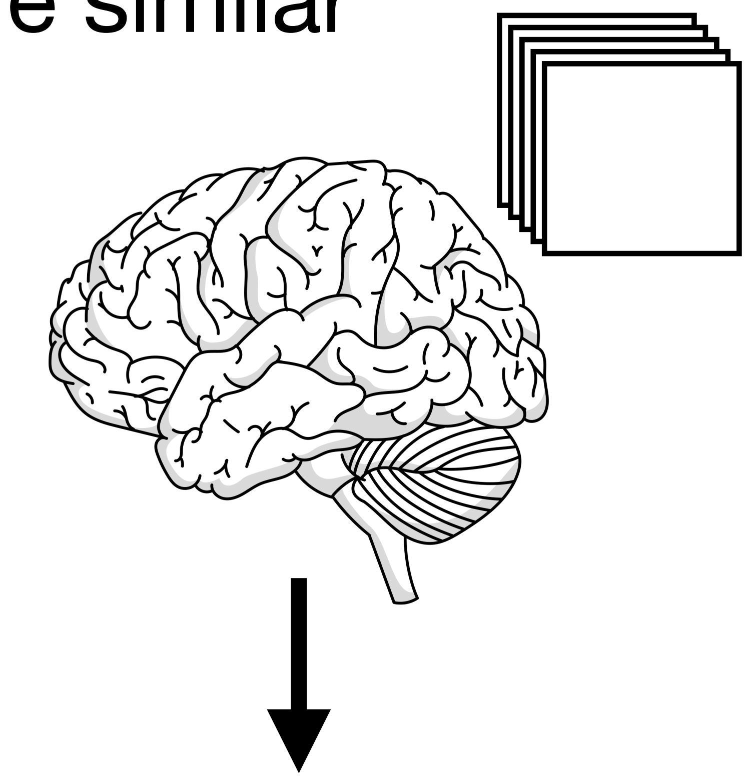
❖ Learn a mapping from all model units to each target neuron.

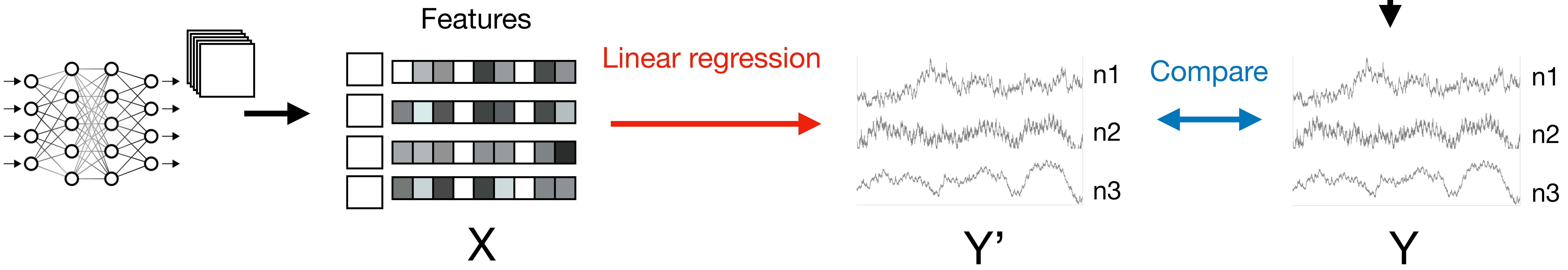


Example: Linear Regression

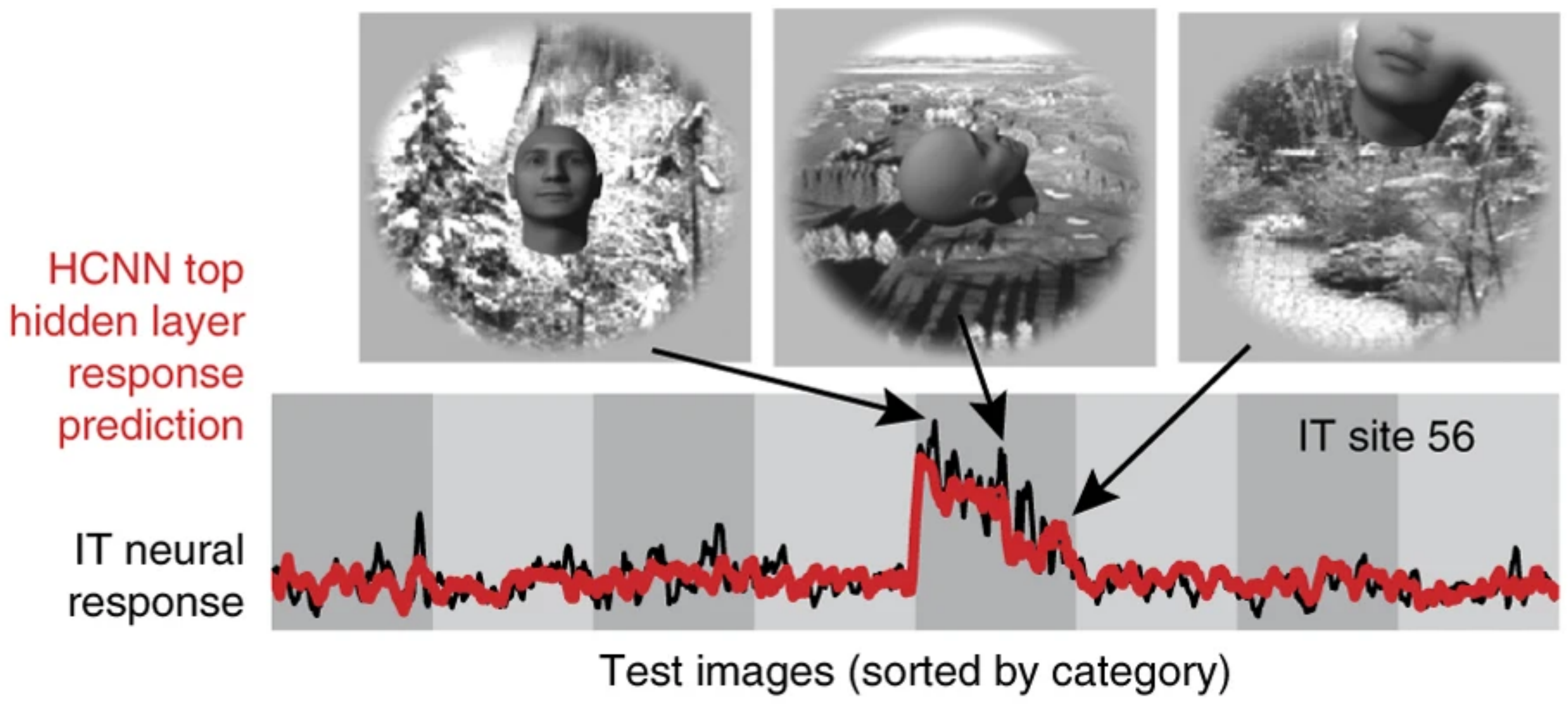
Idea: Find linear combinations of model units that together produce a 'synthetic neuron'

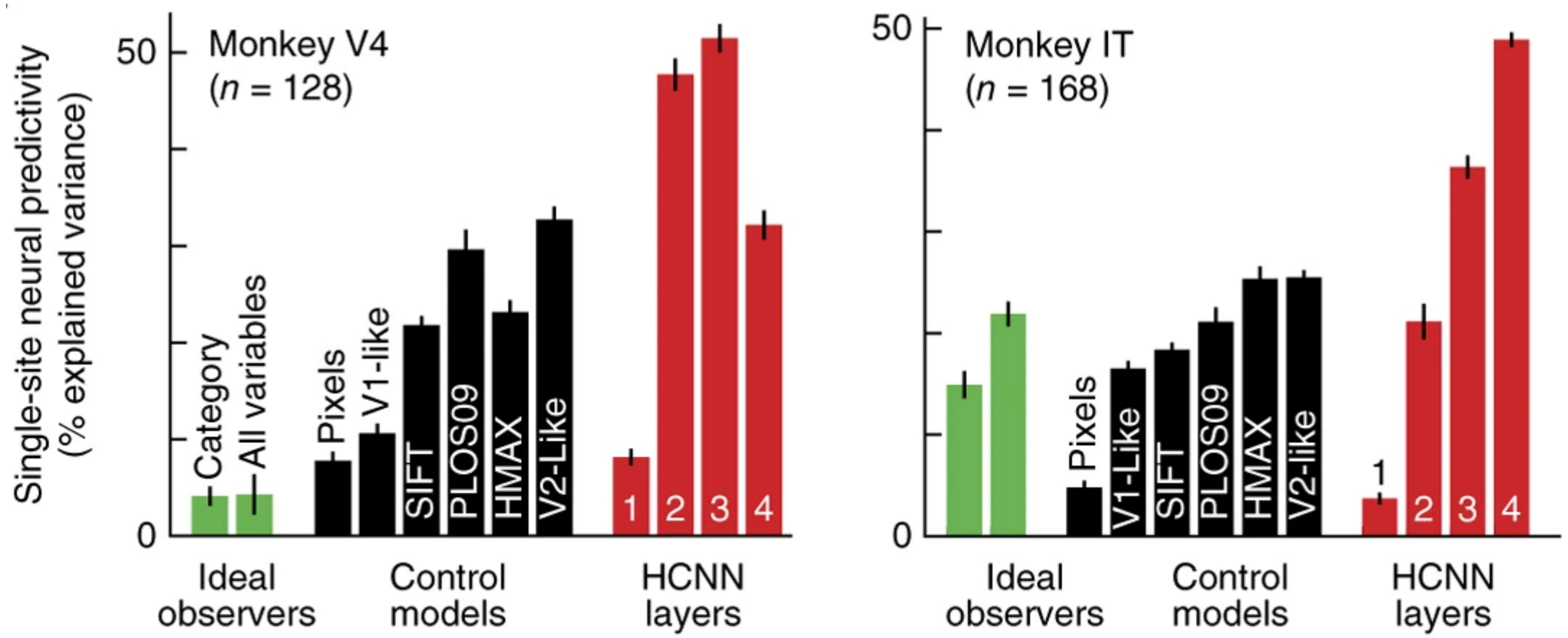
- ❖ Learn a mapping from all model units to each target neuron.
- ❖ **Pros:** More flexible than RSA & one to one matching, not prone to errors when systems are similar
- ❖ **Cons:** Need to train parameters





Example: Linear Regression





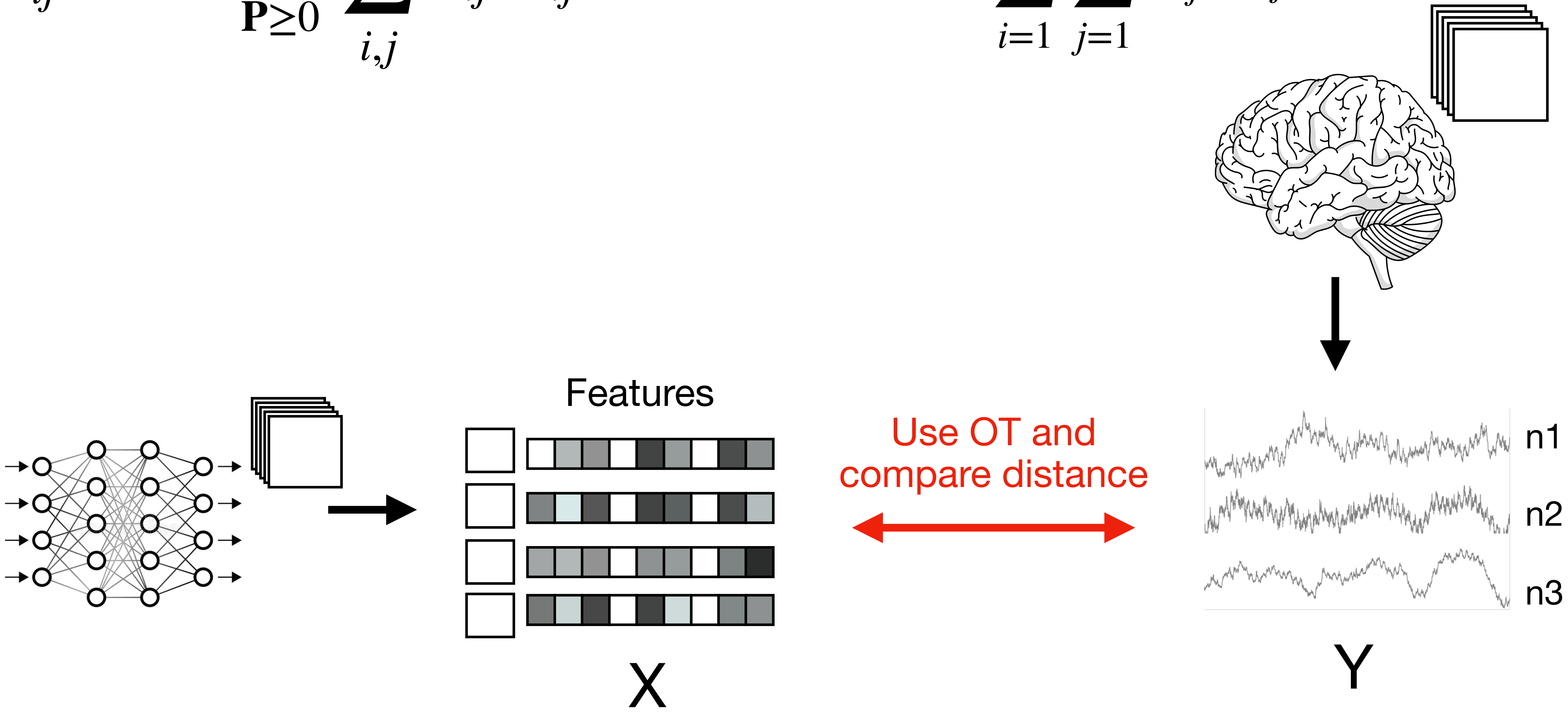
Yamins & DiCarlo, 2016

Example: Soft Matching

Idea: Match individual model units to individual neurons without requiring an exact one to one match

❖ Solve an Optimal Transport (OT) problem: “how much is a source unit matched to a target unit subject to mass conservation constraints”

$$M_{ij} = 1 - \mathrm{corr}\left(x_i, y_j\right), \qquad P^\star_{ij} = \underset{\mathbf{P}\geq 0}{\arg\min} \sum_{i,j} P_{ij} M_{ij} \qquad \mathrm{SMD}(X, Y) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} P^\star_{ij} M_{ij}$$



Khosla & Williams, 2023

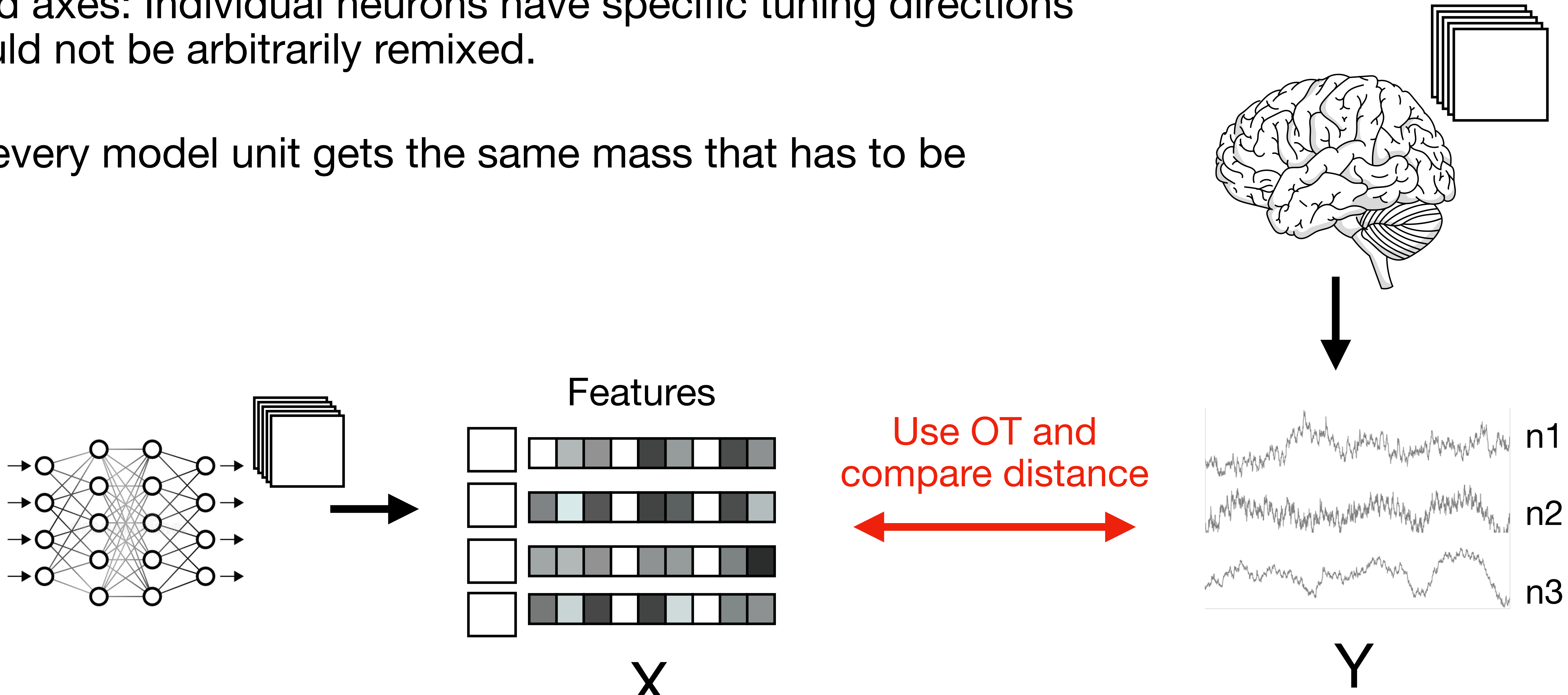
Example: Soft Matching

Idea: Match individual model units to individual neurons without requiring an exact one to one match

❖ Solve an Optimal Transport (OT) problem: “how much is a source unit matched to a target unit subject to mass conservation constraints”

❖ **Pros:** Supports the idea of privileged axes: Individual neurons have specific tuning directions that matter mechanistically and should not be arbitrarily remixed.

❖ **Cons:** Mass constraint means that every model unit gets the same mass that has to be distributed somewhere



Khosla & Williams, 2023

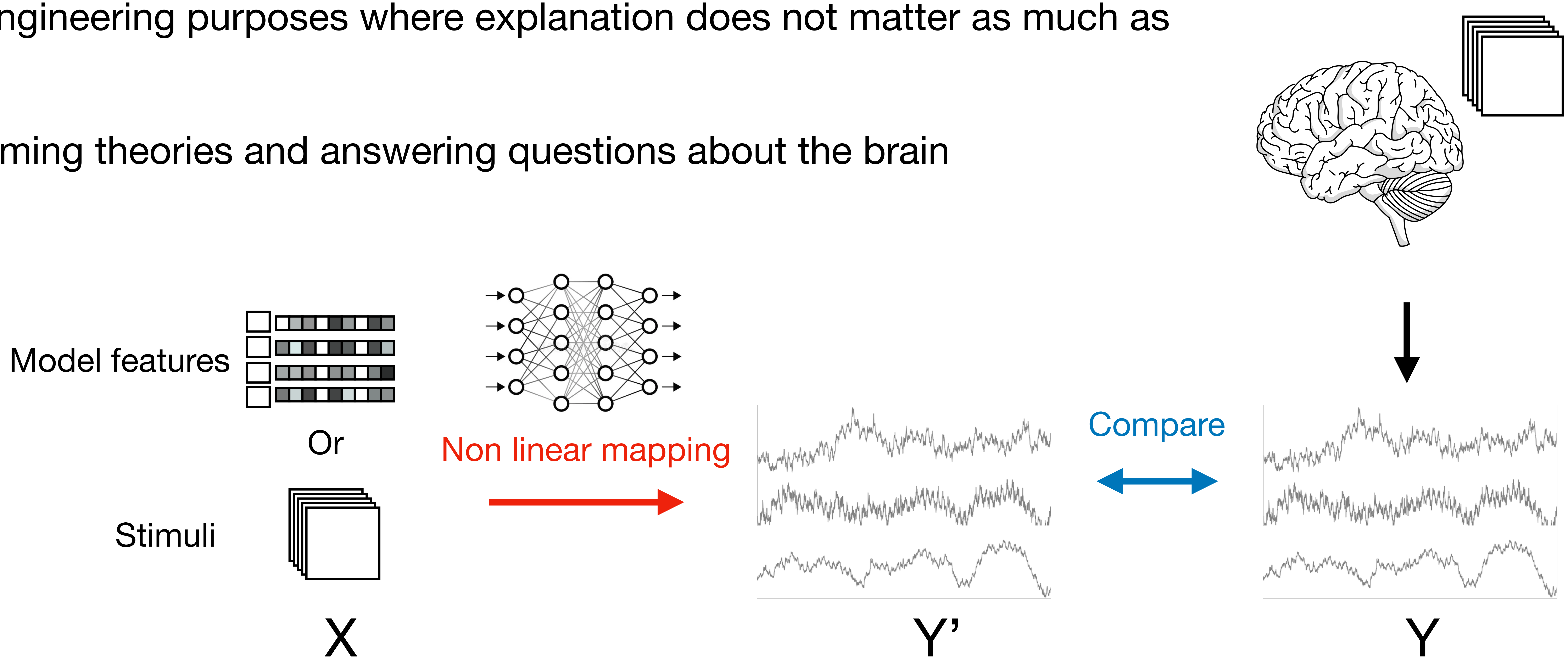
Nonlinear mapping

Idea: Use a neural network (transformer, convnet, etc) to learn the brain data from the stimuli or model features

❖ Predict neural data using back propagation

❖ **Pros:** Very useful for engineering purposes where explanation does not matter as much as prediction accuracy

❖ **Cons:** not great for forming theories and answering questions about the brain



Selecting the right method

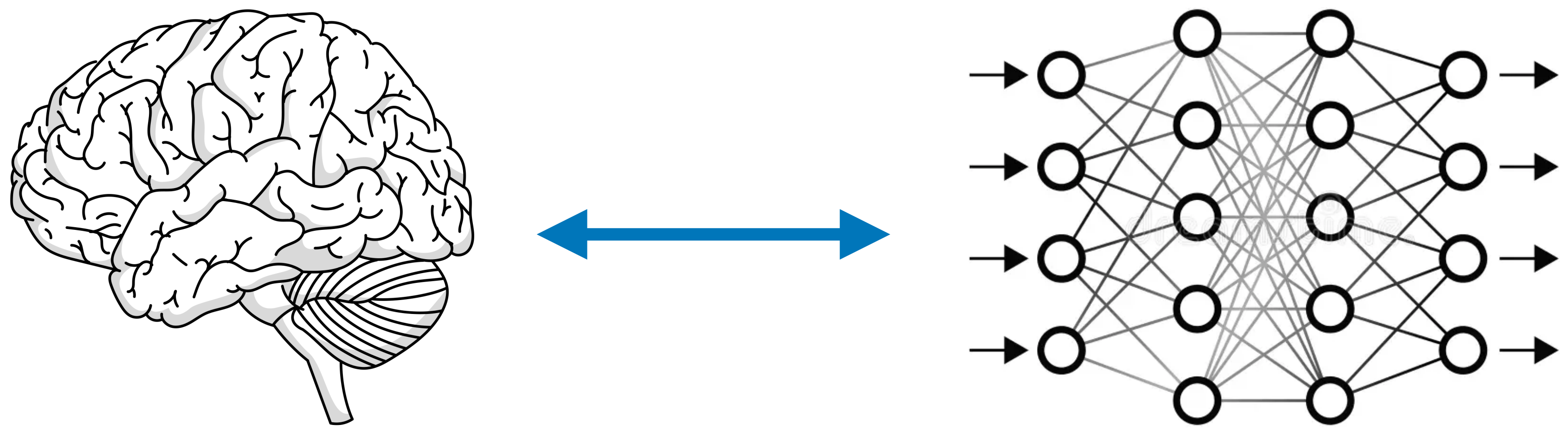
What is your goal?

Studying the brain -> linear mapping methods, RSA, CKA, etc

Building a model of the brain -> nonlinear mapping (brain foundation models)

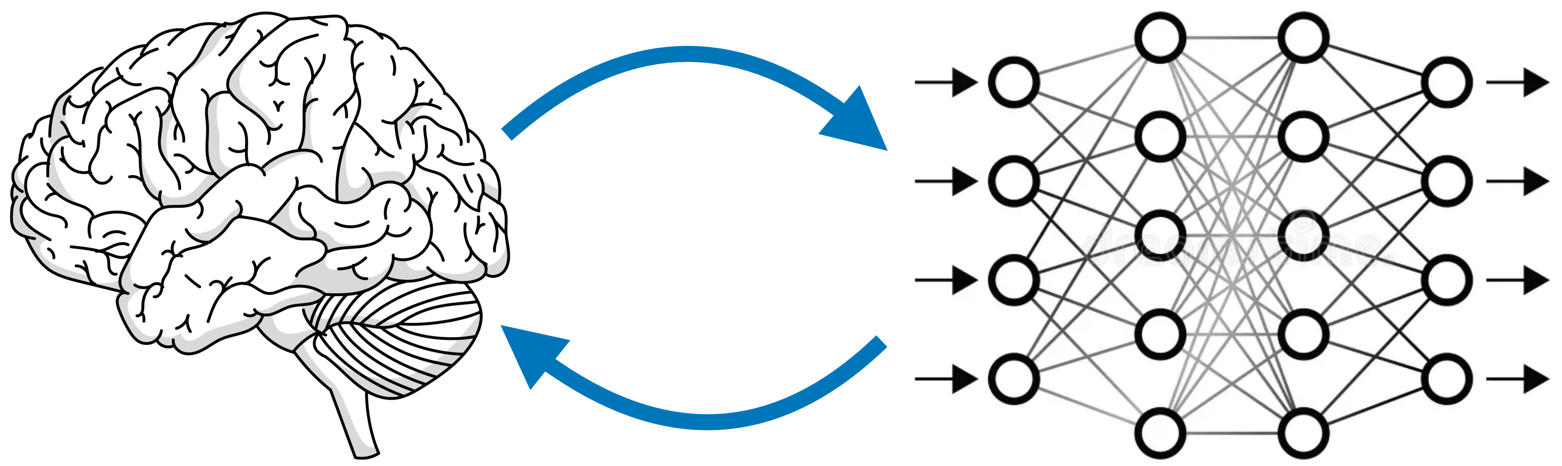
Selecting the right method (for studying the brain)

Symmetry vs bidirectionally



Symmetry

- Many metrics are symmetric by definition (RSA, CKA, soft matching)
- **Problem:** we have access to all model units but often only a small amount of brain units



Bidirectionally

- Brain-brain transform is not symmetric, why should model-brain be?

The inter animal transform class (IATC) framework

- Identify the narrowest class of transforms that maps responses between subjects for a given brain area and species.

- The right class of transform should be:

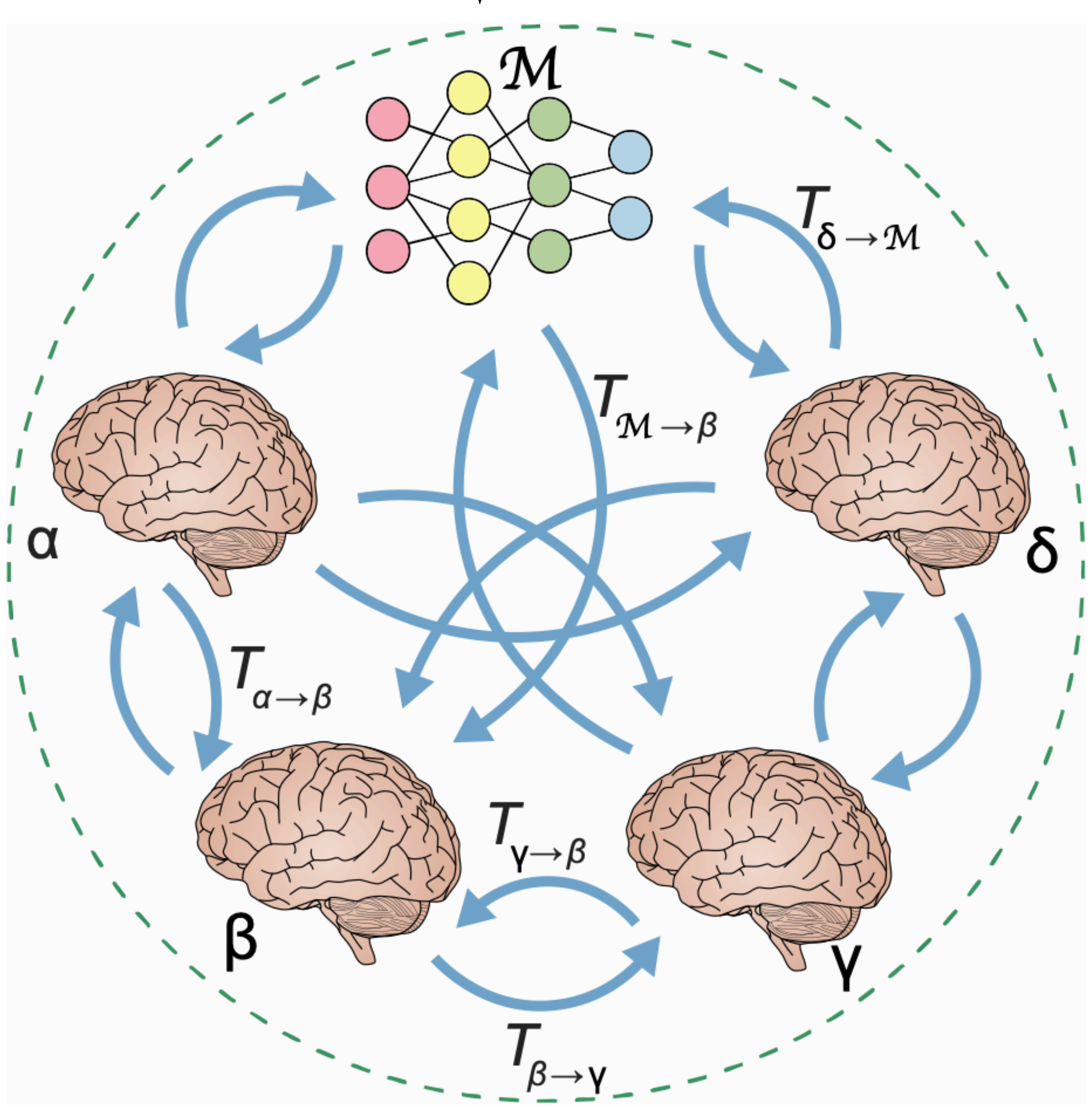
Predictive

- Maximally predict neural responses

Strict

- Distinguish brain areas while recognizing the same areas across subjects

"how well can the model masquerade as a member of the population?"



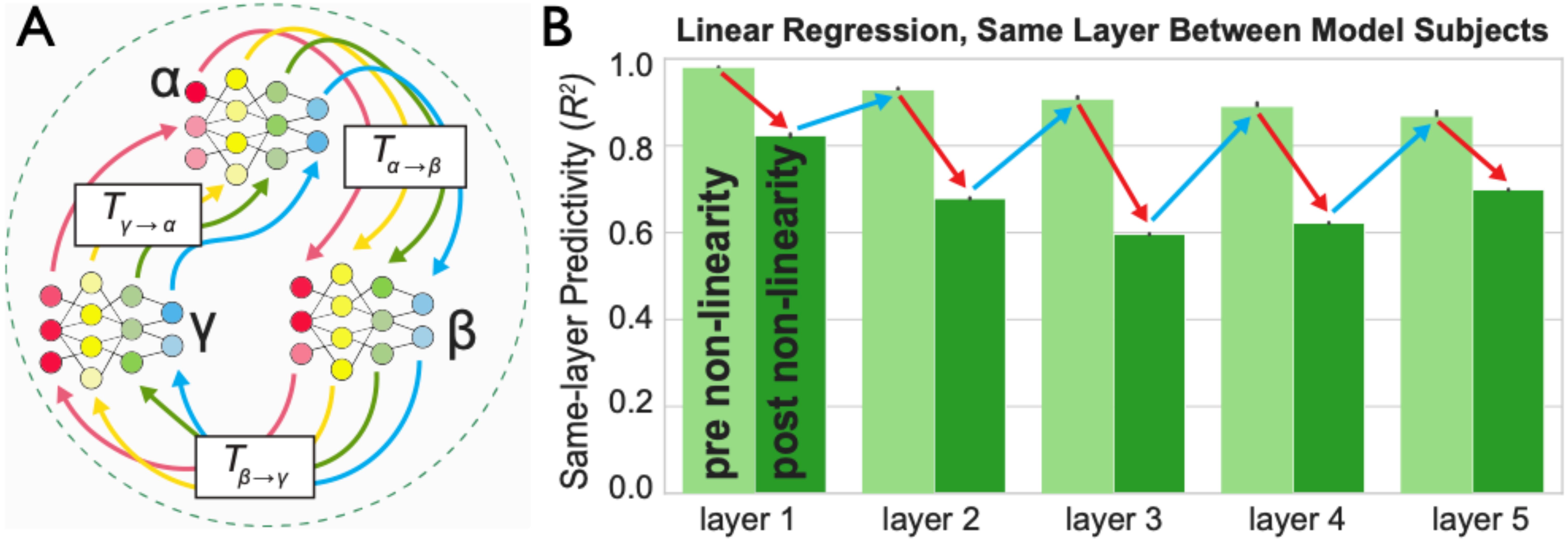
Thobani et al, 2025

Assessing same-area similarity in a model population

Model: Modified AlexNet

- Trained with contrastive learning
- Softplus activation function + Poisson-like noise

Population simulation: vary the random seed controlling initialization and training data order.



Zippering effect

Thobani et al, 2025

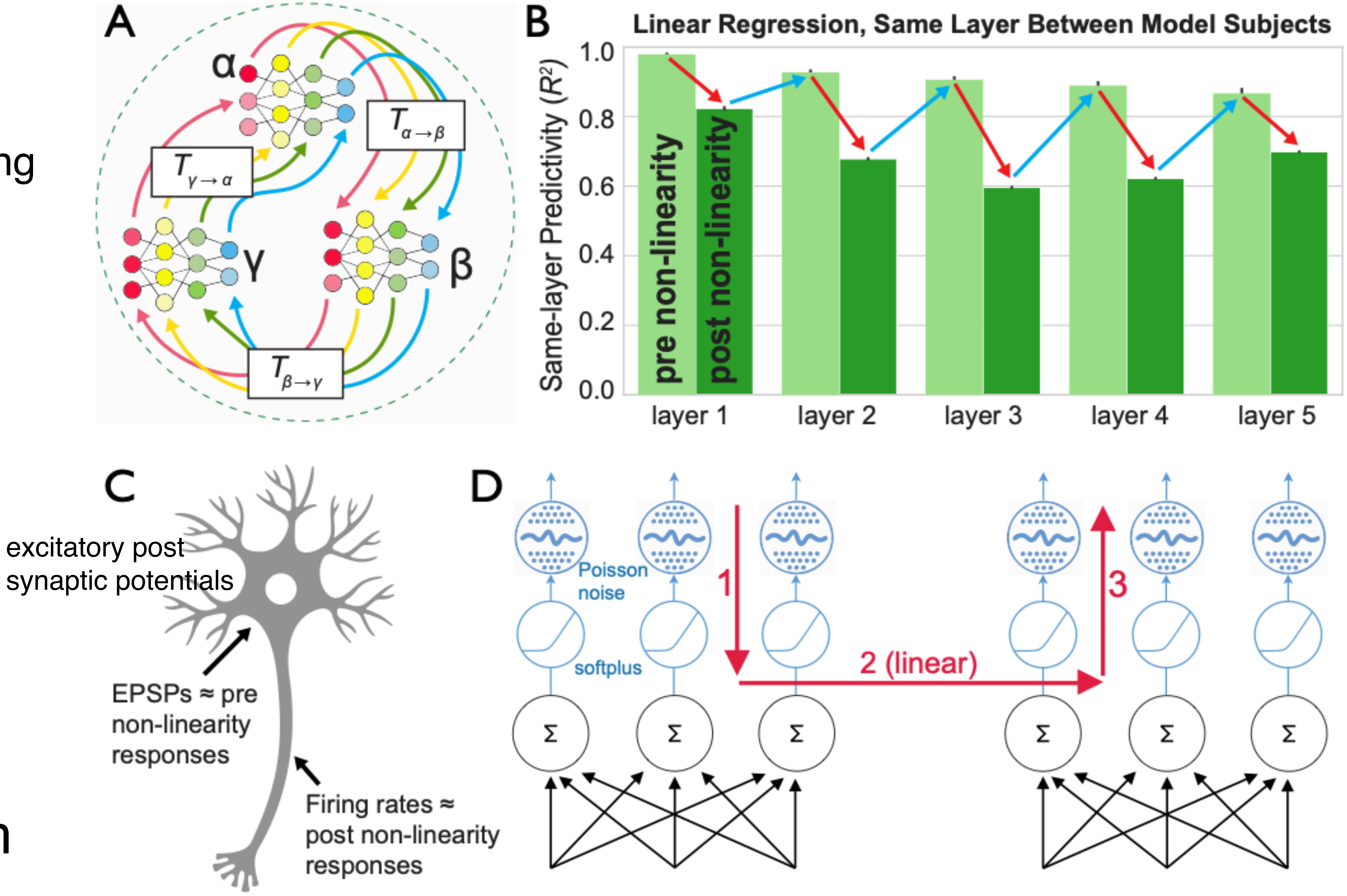
Assessing same-area similarity in a model population

Model: Modified AlexNet

- Trained with contrastive learning
- Softplus activation function + Poisson-like noise

Population simulation: vary the random seed controlling initialization and training data order.

New transform class: The zippering transform



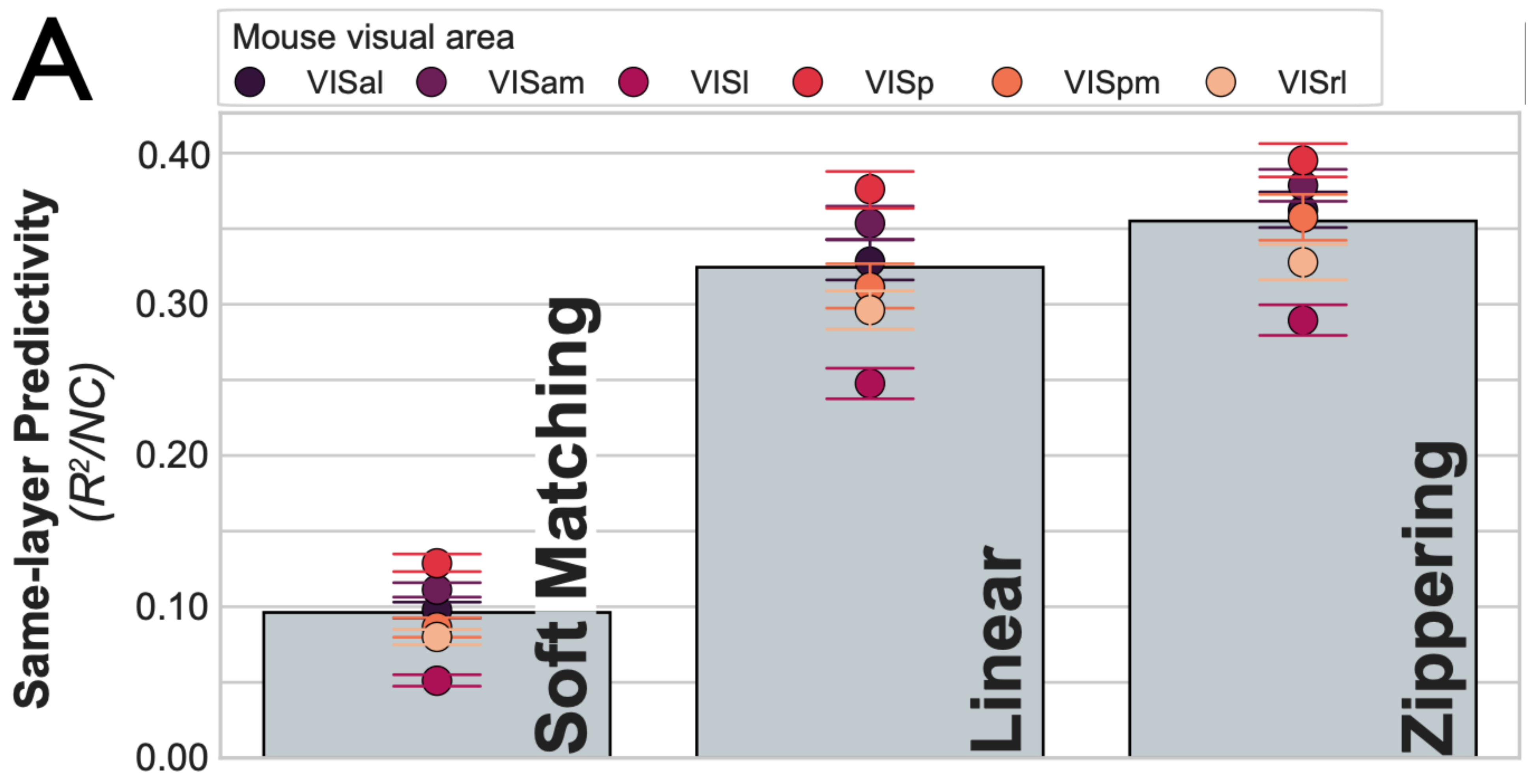
Thobani et al, 2025

Applying IATC to the mouse neural data

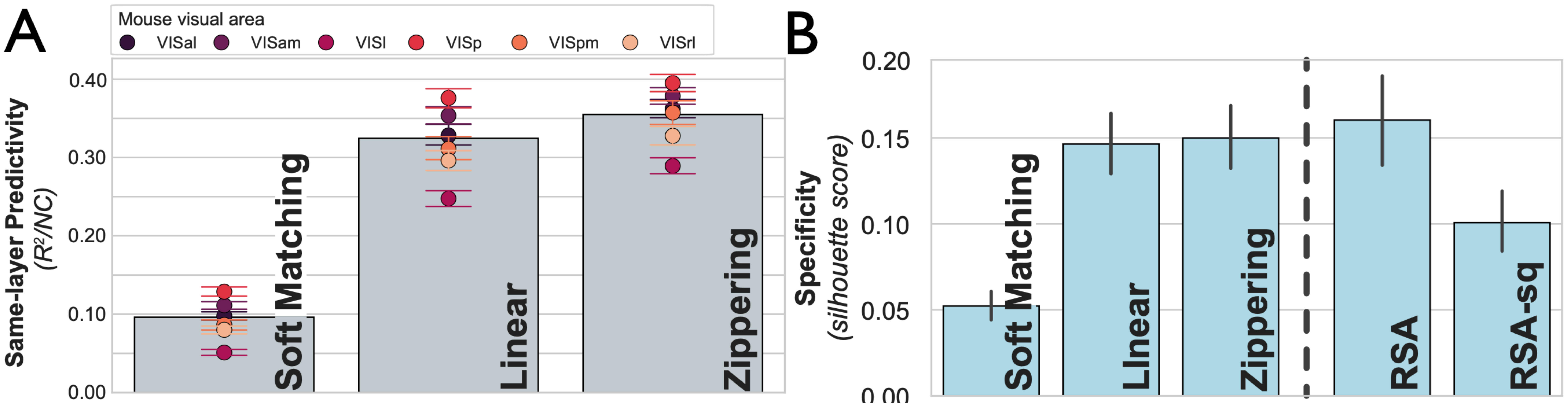
Dataset

- Neuropixel recordings for 31 subjects
- 6 brain areas
- The mice passively viewed 118 different visual stimuli.

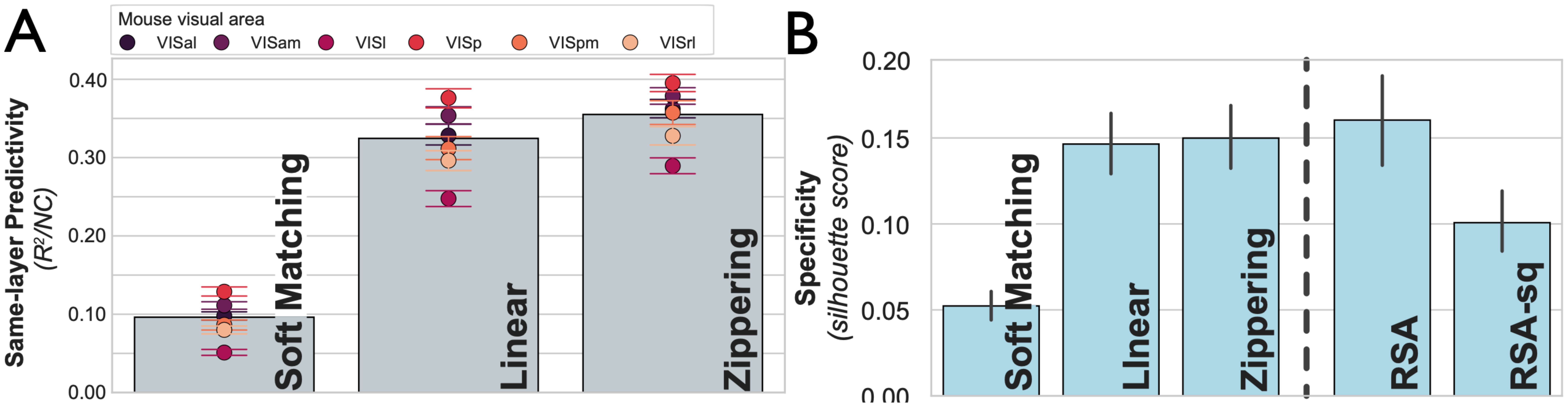
Applying IATC to the mouse neural data



Applying IATC to the mouse neural data



Applying IATC to the mouse neural data

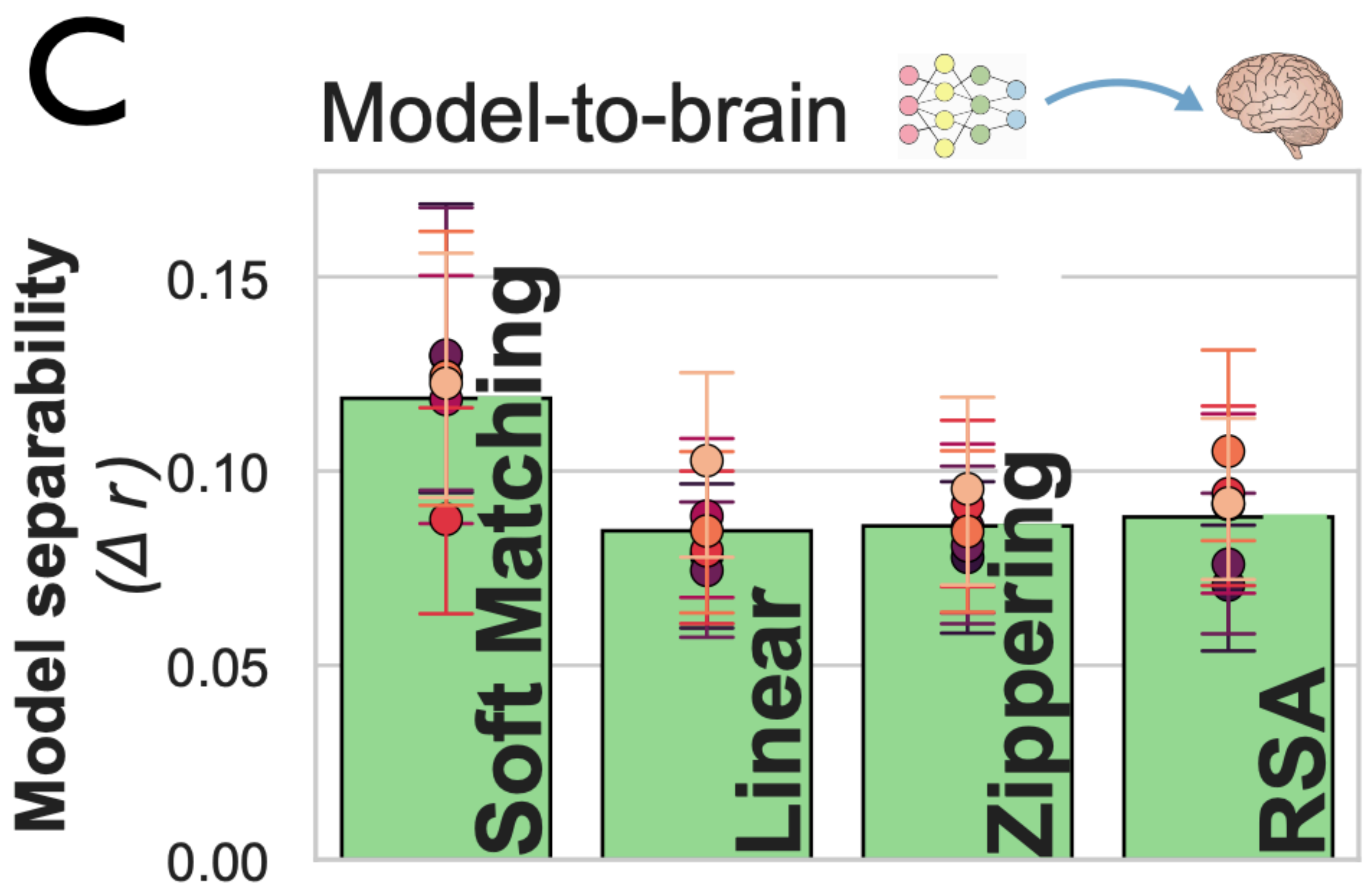


$$s(i) = \frac{b(i) - a(i)}{\max(b(i), a(i))}$$

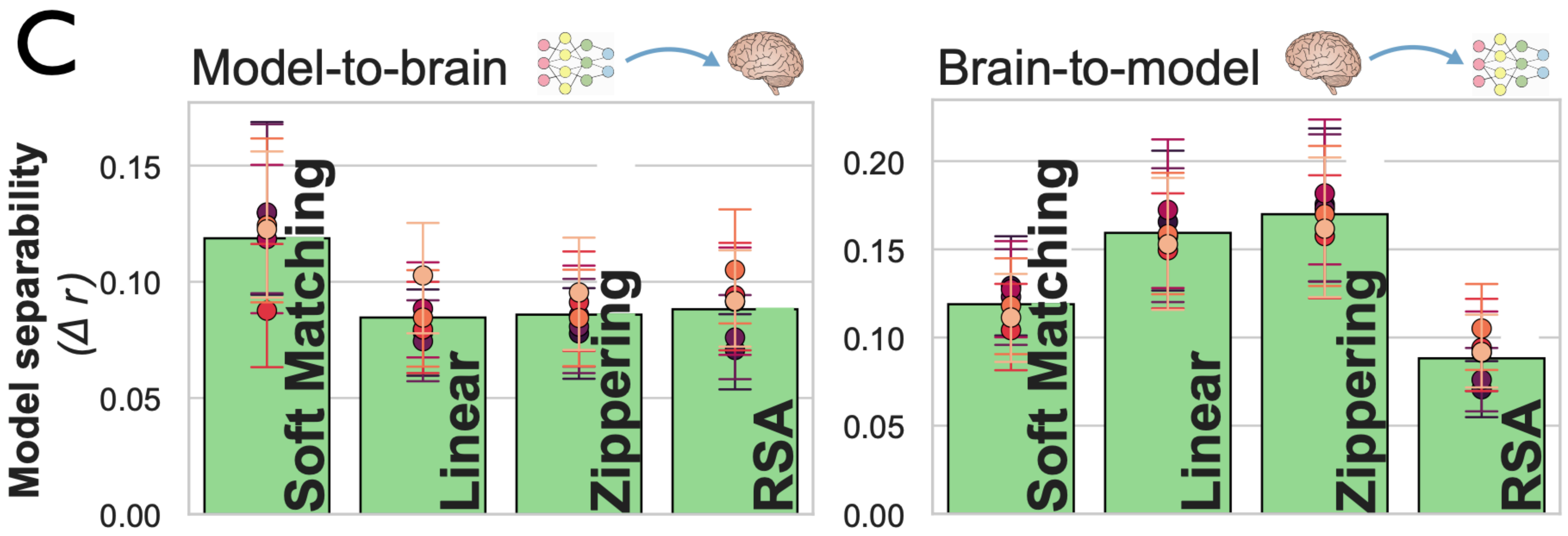
a(i): within-area dissimilarity
b(i): between-area dissimilarity

Good specificity: **low a(i)** and **high b(i)**

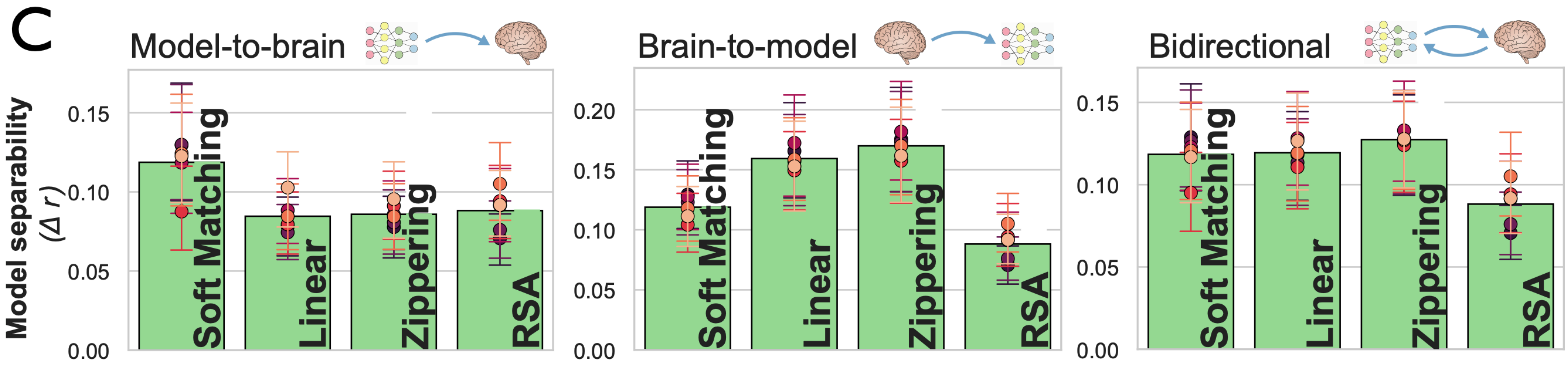
IATC Guided Model Separability



IATC Guided Model Separability



IATC Guided Model Separability



Noise in neural data

- A fundamental challenge in evaluating the performance of NN models lies in the noise inherent in empirical data

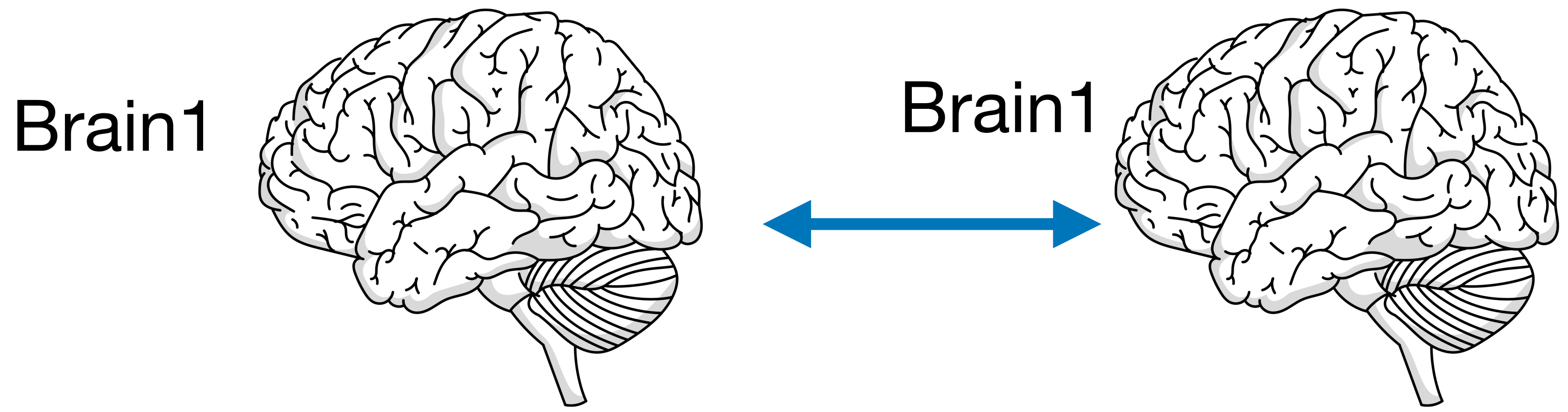
- Examples of noise:
  - Motion artifacts (head motion)
  - Attention fluctuations
  - Arousal
  - Eye movements

- Why does this matter?

  - If a model only captures 20% variance in the data, this could indicate poor model performance.
  - If there is a high degree of noise, 20% may be as good as it gets.

Noise ceiling estimates: Inter animal vs cross animal

Inter animal spit-half reliability

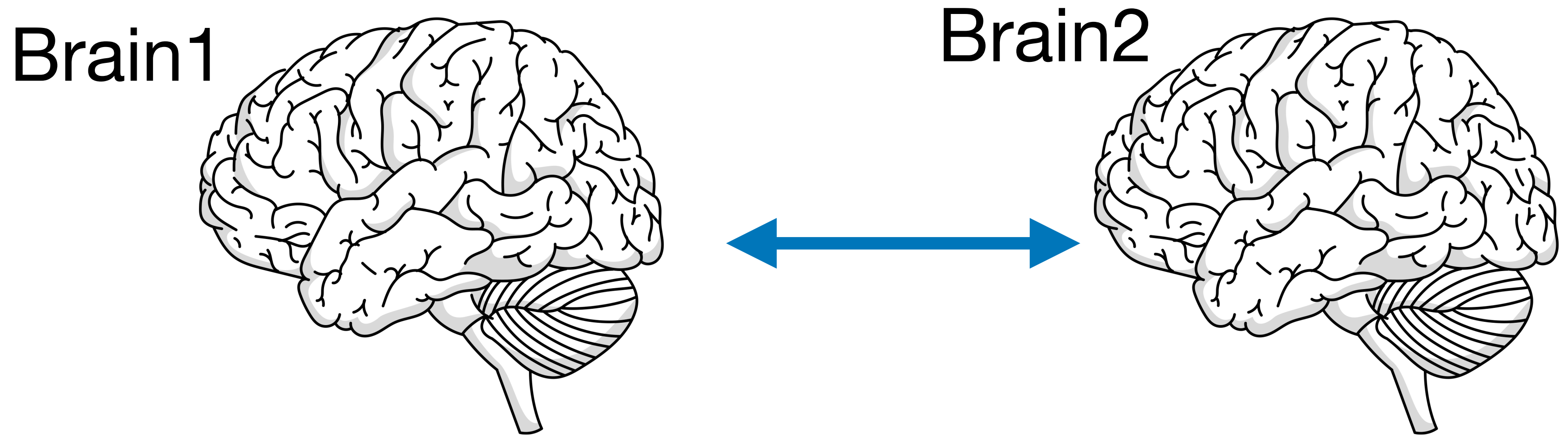


- Common for evaluating individual participant data when you have repeats

$$r_{\mathrm{sh}} = \mathrm{corr}\left(Y^{(1)}, Y^{(2)}\right)$$

- Repeated measurements are divided into two halves and the responses are correlated.

Cross animal



- **Fundamental question:** How do we decide how to measure similarity across different animals?
  - Use the IATC framework to find the right class of transforms