Research problem: visual perception

- Primary visual cortex (V1)

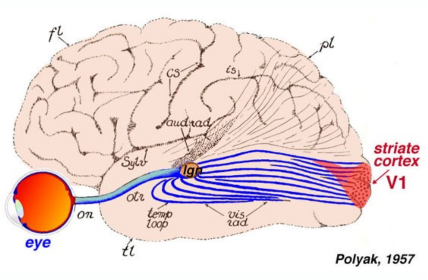


Figure 1: Visual input goes from the eye to LGN and then to primary visual cortex (V1). (Adapted from Polyak (1957))

- Question: what is the functional geometry and circuit in V1?
- Data available: neural spiking data

Unsupervised Discovery of Functional Neural Circuits in Visual System through Tensor Factorization and Diffusion Map

Liu Zhang
Mentors: Prof. Steven W. Zucker and Dr. Luciano Dyballa

July 26, 2021

Visualizing a data sample

- Given some visual stimuli (both artificial gratings and natural images), we can record the responses from the neurons in V1.
- Neural recordings are stored as peristimulus time histogram (PSTH) diagram, visualized as an 8-by-33 image.

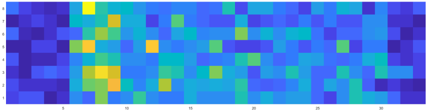


Figure 2: Visualizing PSTH diagram.

Challenges in neural data analysis in V1

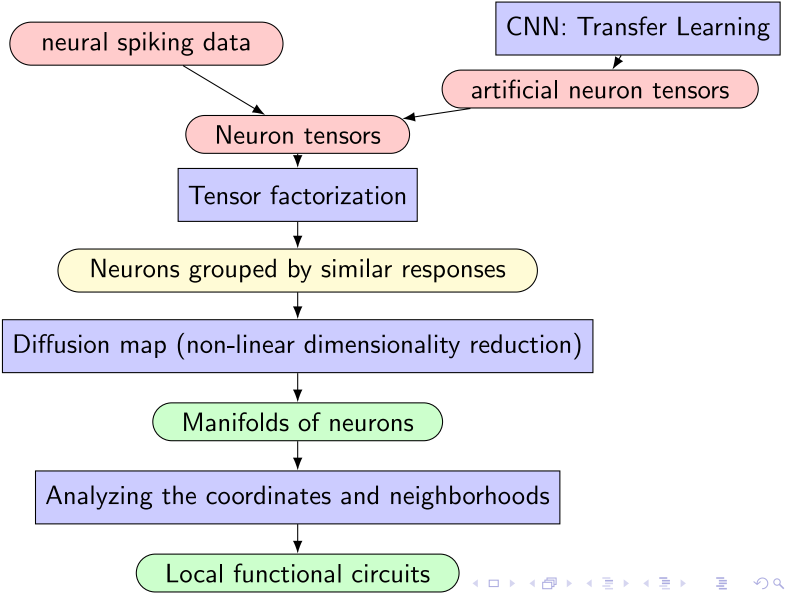
- Limited experimental results...

  Solution: use Artificial Neural Networks (ANNs) to generate artificial neuron tensors.
- High-dimensional...

  Solution: use dimensionality reduction methods like tensor factorization and diffusion map.
- Complex geometric structure...

  Solution: use analytic tools from differential geometry.

Summary of the method

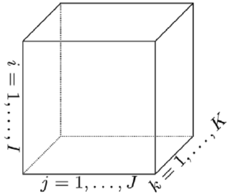


From matrices to tensors

Definition (Tensors)

An N-way tensor is an element of the tensor product of N vector spaces.

- 1-*way tensor* = *vector*, $v = [v_1 \quad v_2 \quad \dots \quad v_n]^T$.
- 2-*way tensor* = *matrix*, $A = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mn} \end{pmatrix}$.
- 3-*way tensor of dimension* I-*by*-J-*by*-K:



From matrix factorization to tensor factorization

$$m\left\{\overbrace{\begin{bmatrix} \quad & A & \quad \end{bmatrix}}^{n}\right. \approx m\left\{\overbrace{\begin{bmatrix} X \end{bmatrix}}^{k} \left.\overbrace{\begin{bmatrix} \quad & Y & \quad \end{bmatrix}}^{n}\right\}k\right.$$

Figure 3: Intuition for matrix factorization.

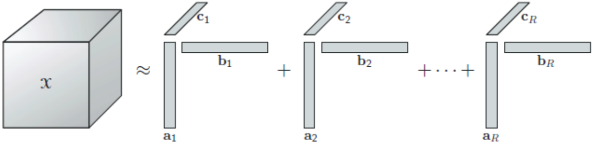


Figure 4: Intuition for tensor factorization. (Adapted from [1].)

Demo: apply tensor factorization to face image data

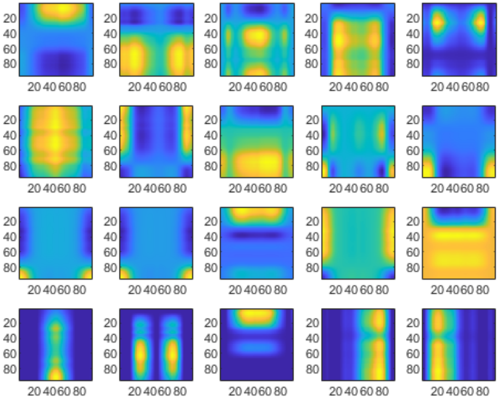


Figure 5: First 20 tensor factors for face image data.

Demo: apply tensor factorization to face image data

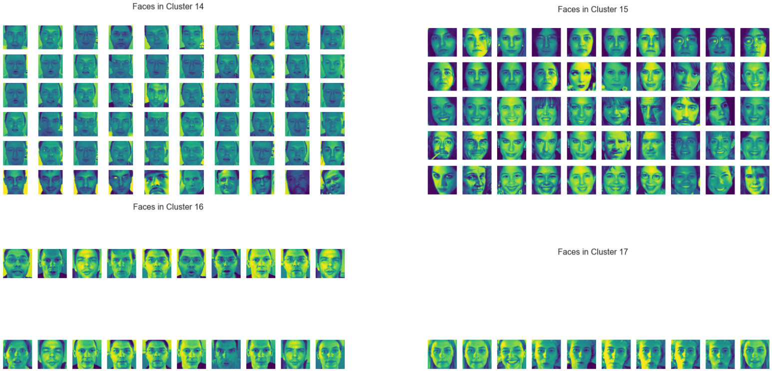


Figure 6: Some arbitrary clusters showing the results of TCA for face data.

Preliminary results: apply tensor factorization to retina neural data

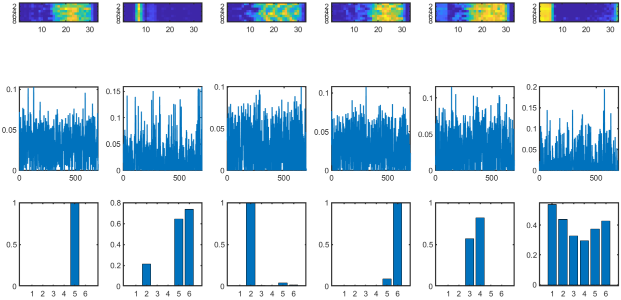
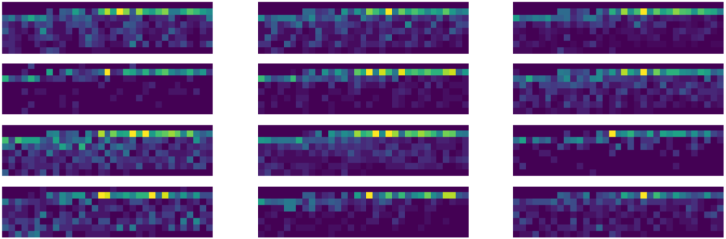
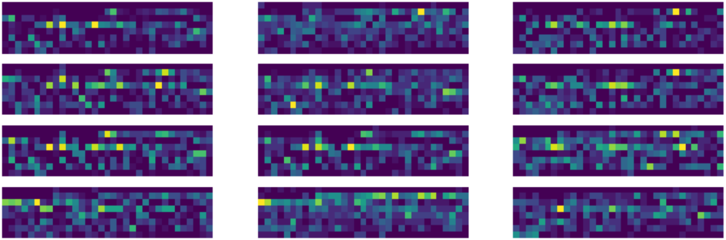


Figure 7: Visualizing the first six tensor factors for retina data.

Preliminary results: apply tensor factorization to retina neural data



(a) PSTH diagrams showing responses of neurons within some arbitrary cluster to stimuli of type 1.



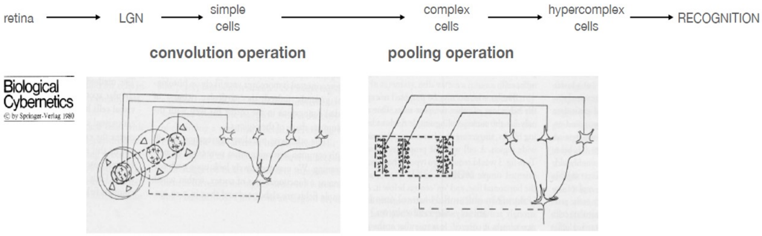
(b) PSTH diagrams showing responses of neurons within a different cluster to stimuli of type 1.

Modeling simple and complex cells: the hierarchy model by Hubel and Wiesel

- convolution operator: at simple cells. The convolution of $K(t)$ (e.g. a kernel) and $g(t)$ (e.g. an image), $K * g$ is defined by

$$[K * g](t) = \int_0^t K(u)g(t-u)du. \tag{1}$$

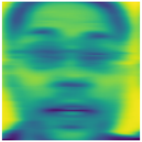
- pooling operator (taking the average or maximum): at complex cells



More on convolution and kernels

We can define the kernel and convolve it with an image so as to modify the image in a desired way. For example:

- The Gaussian kernel blurs the image:



- The Laplacian of Gaussian kernel sharpens the image:

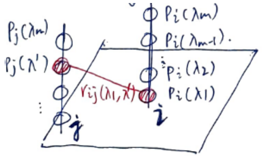
Alternative to the hierarchy model: recurrent models

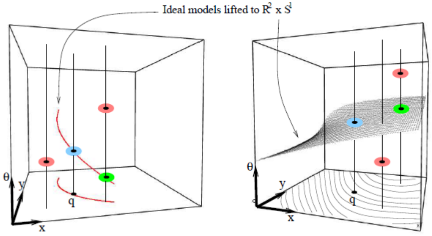
Recurrent models: study networks of cortical connections instead of individual cells.

Recurrent computation:

$$S_i(\lambda) = \sum_{j \in \text{neigh}(i)} \sum_{\lambda' \in \Lambda(j)} r_{i,j}(\lambda, \lambda') p_j(\lambda') \tag{2}$$

$$P_i^{t+1}(\lambda) = \prod_k [P_i^t(\lambda) + \delta s_i(\lambda)] \tag{3}$$

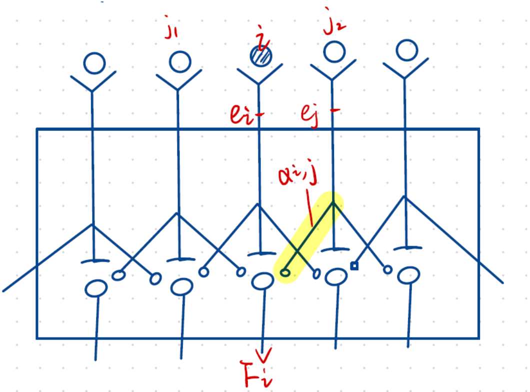




Bonus slide: modeling lateral inhibition

Algebraic model for lateral inhibition:

$$F_i = e_i - \sum_{j \in \text{neighbors}(i)} \alpha_{i,j} \quad e_j, \quad \alpha_{i,j} \in \mathbb{R}^+. \tag{4}$$



Acknowledgements

I would like to thank my mentors Prof. Steven W. Zucker and Dr. Luciano Dyballa for their generous guidance and advice. They have made possible the many serendipitous moments in this project. I am also grateful for my advisor Prof. Francesca Spagnuolo for her unwavering support throughout this project and my mathematical career so far.

References

Tamara G. Kolda and Brett W. Bader.
Tensor decompositions and applications.
SIAM Rev., 51(3):455–500, August 2009.