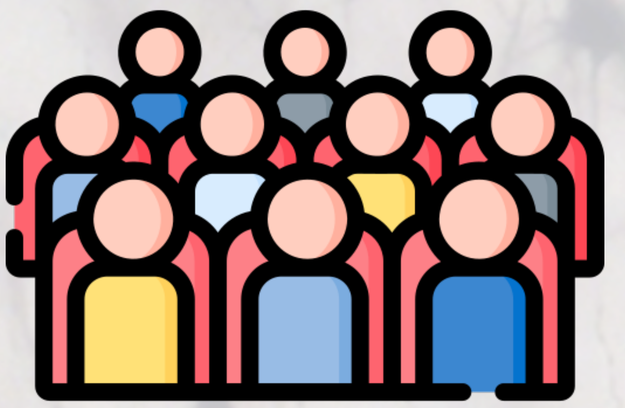
BNP4BNP

Bayesian Nonparametric Models for Biomarkers and Neuronal Patterns

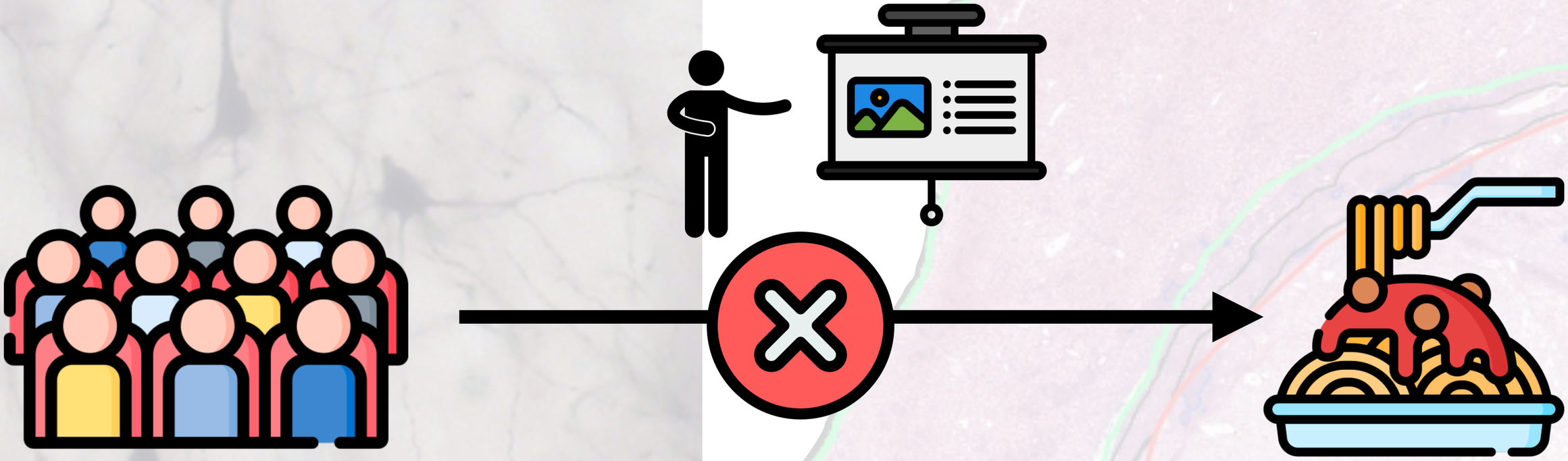
Francesco Denti
francesco.denti@unipd.it

Università degli Studi di Padova
Dipartimento di Scienze Statistiche

I am well aware of this



I am well aware of this



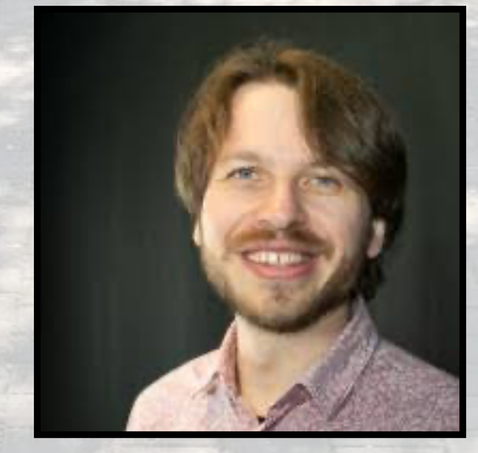
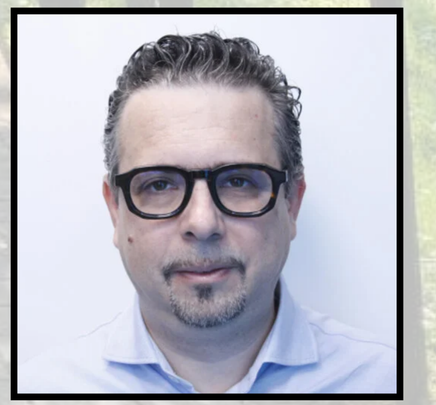
A good place for Bayes!









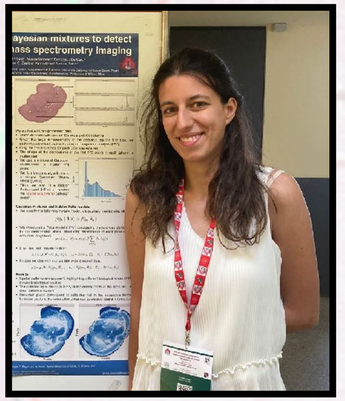
My collaborators

Laura D’Angelo
@ University of Milan-Bicocca

Michele Guindani @ UCLA

Antonio Canale @ UniPD

Cecilia Balocchi
@ University of Edinburgh



Giulia Capitoli
@ University of Milan-Bicocca

MALDI imaging

- **Matrix-Assisted Laser Desorption/Ionisation** (**MALDI**) - MSI tool extracts the distribution of **molecules** (such as **lipids**) in different locations of a biological sample
- It allows the detection of critical biological traits that would be overlooked with a simple visual morphological assessment

- A slide of a biological sample is divided into a grid of pixels of **50 microns each**
- For each pixel, the instrument acquires a **mass spectrum,** a **collection of molecular abundances** measured at different values of the mass-to-charge index **m/z** (also known as **signal**)

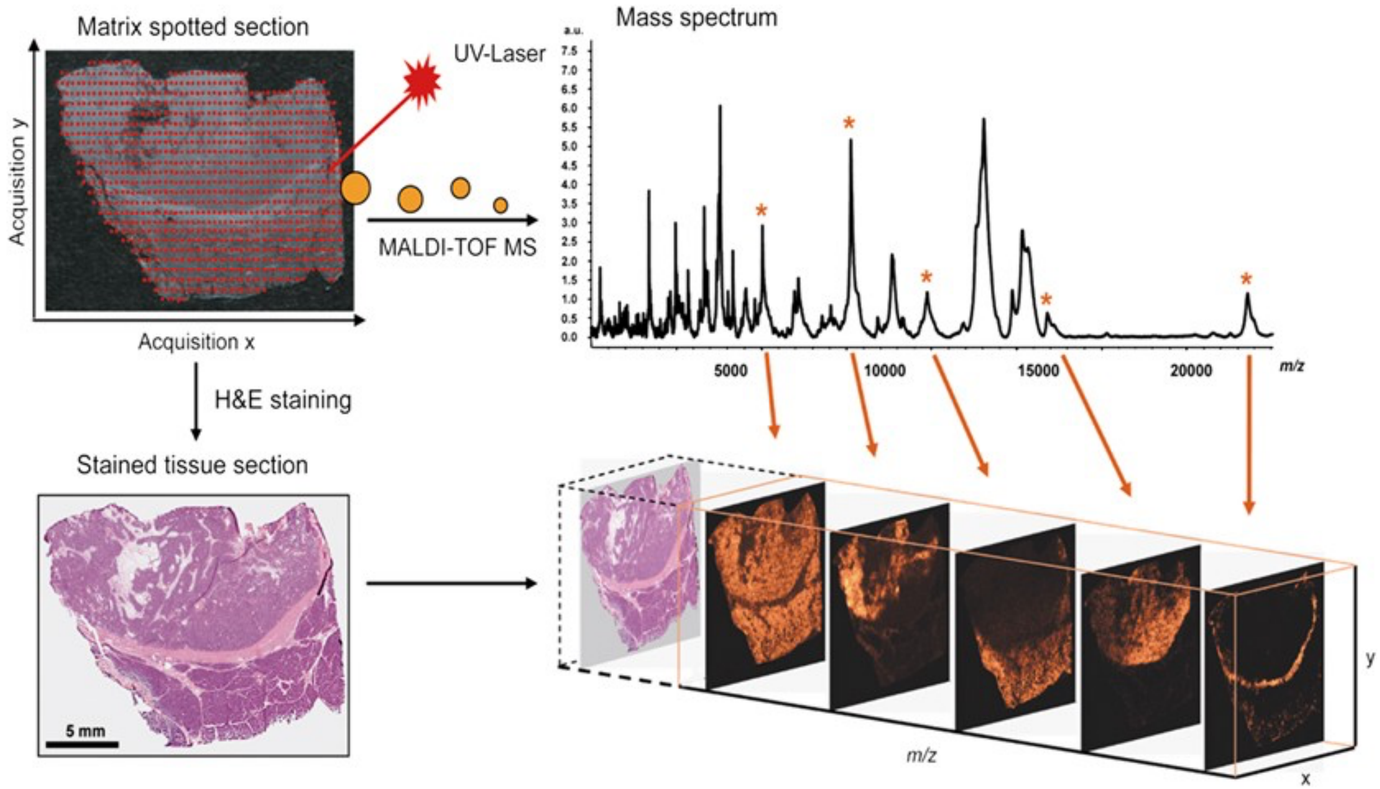
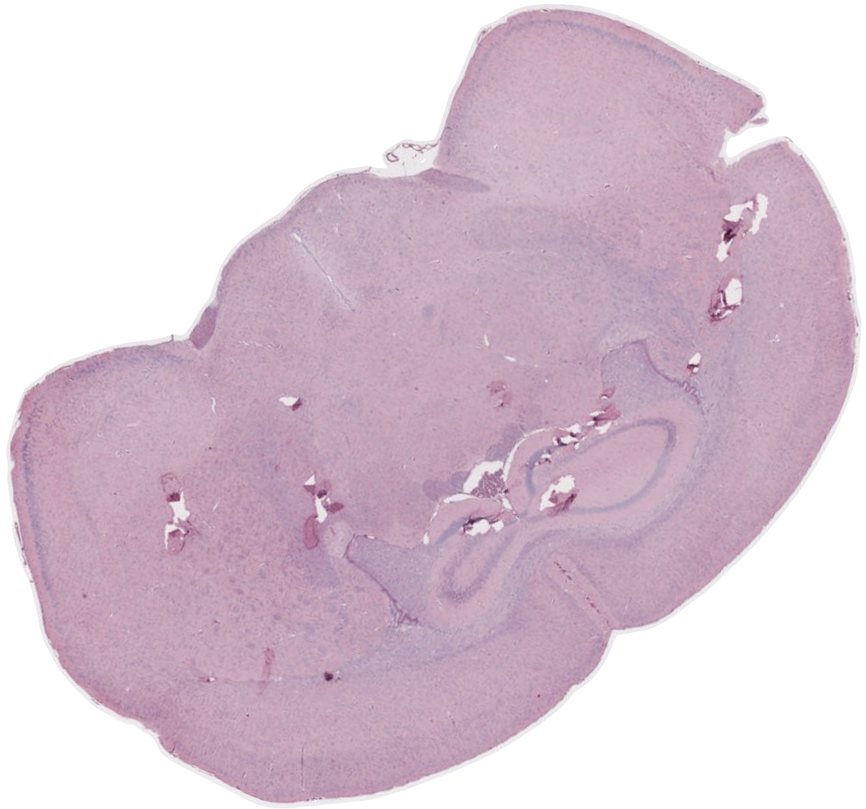


Image from *"MALDI Imaging mass spectrometry: current frontiers and perspectives in pathology research and practice"*, Aichler and Walch, Laboratory Investigation - volume 95, pages 422–431 (2015)

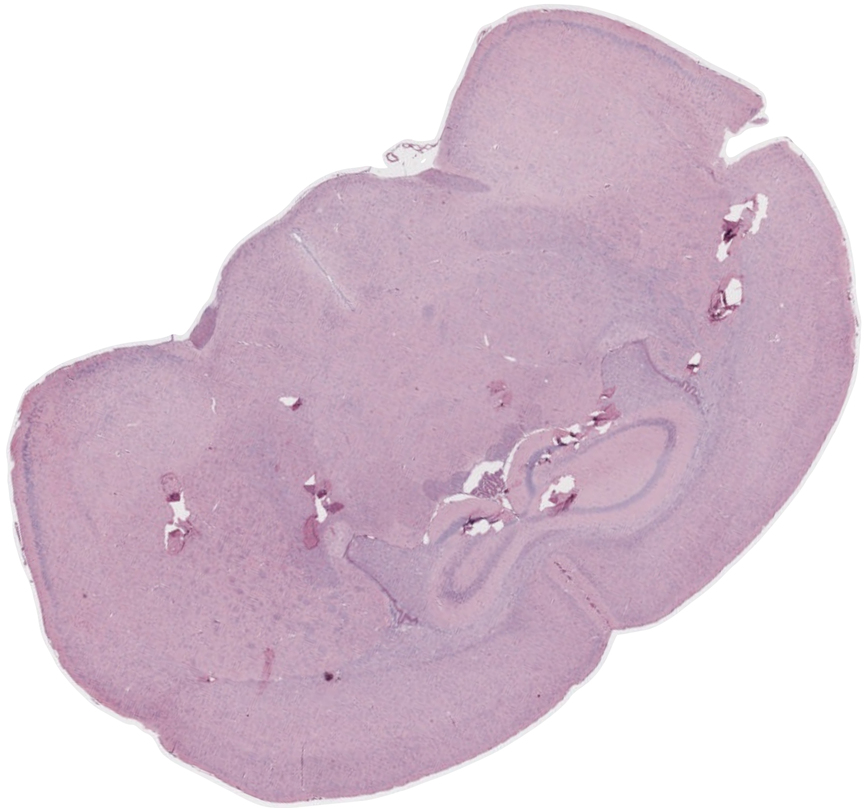
The motivating datasets



- It is obtained from a slice of the brain of a **healthy mouse**, to be used as a benchmark
- The atlas of the mouse brain is well known - this way, biologists can validate their preprocessing pipelines
- I will use this dataset as a tool to build the model

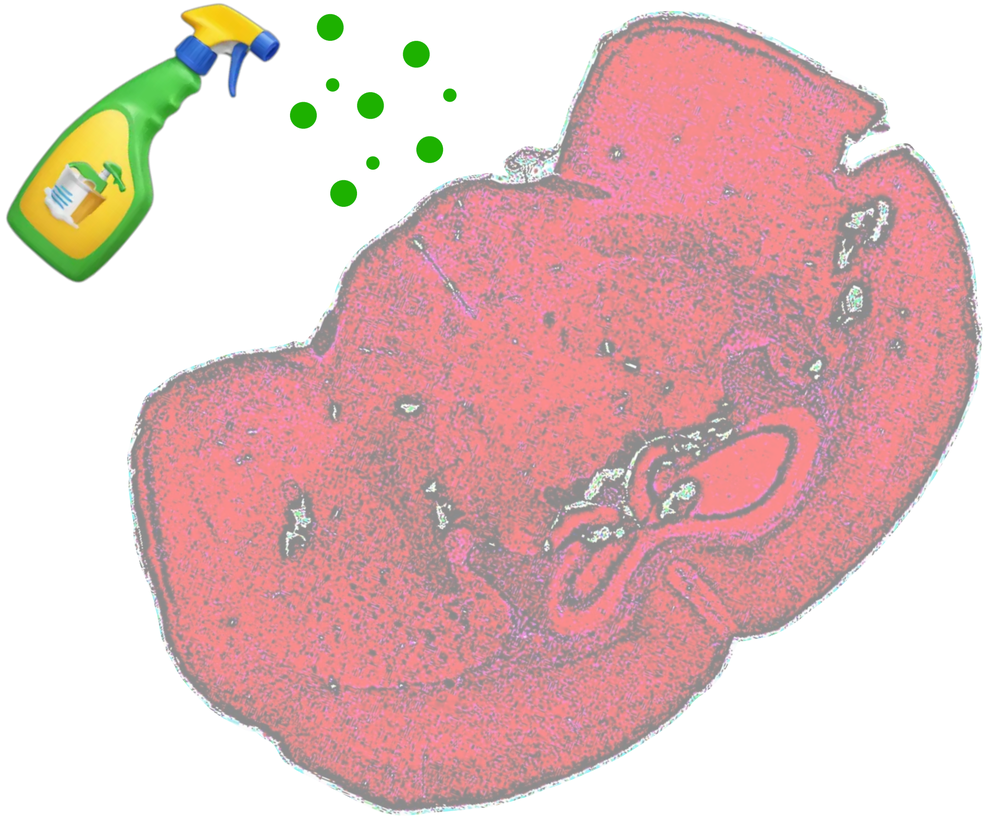
MALDI-MSI procedure

- We want to develop new statistical methods that take into account the information of these complex data



MALDI-MSI procedure

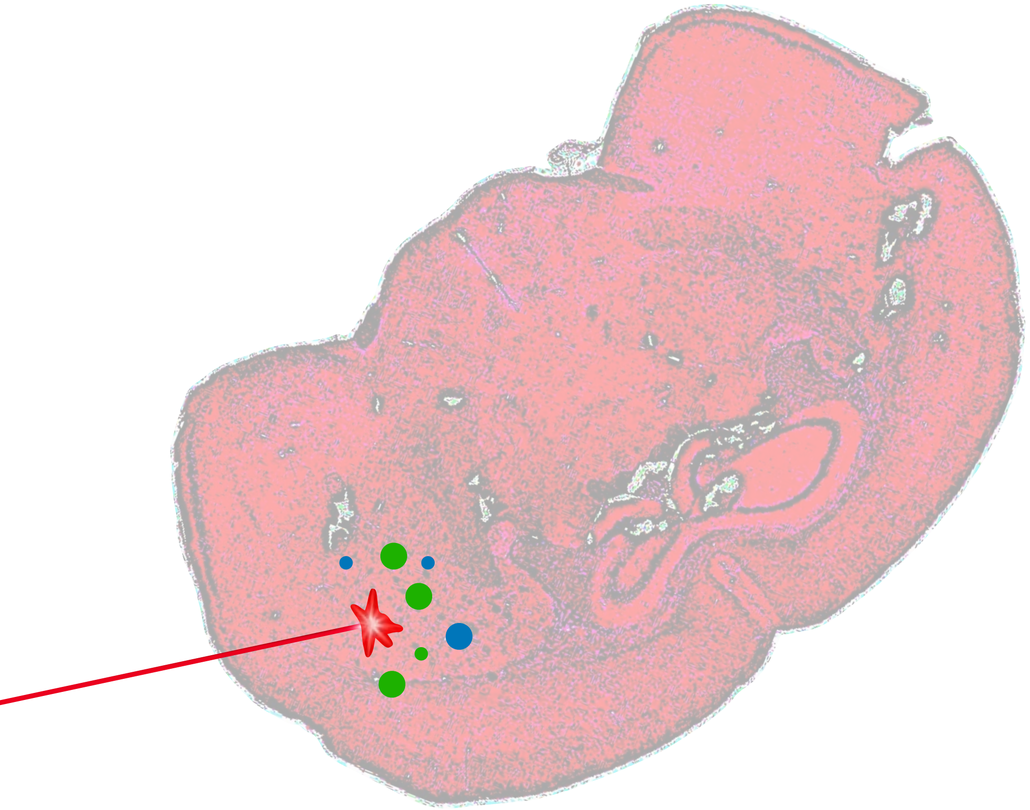
- We want to develop new statistical methods that take into account the information of these complex data



- The tissue is covered by a specific substance called *matrix*, which separates a specific molecule (here, **lipids**) from the rest and makes it “ready to be detected”

MALDI-MSI procedure

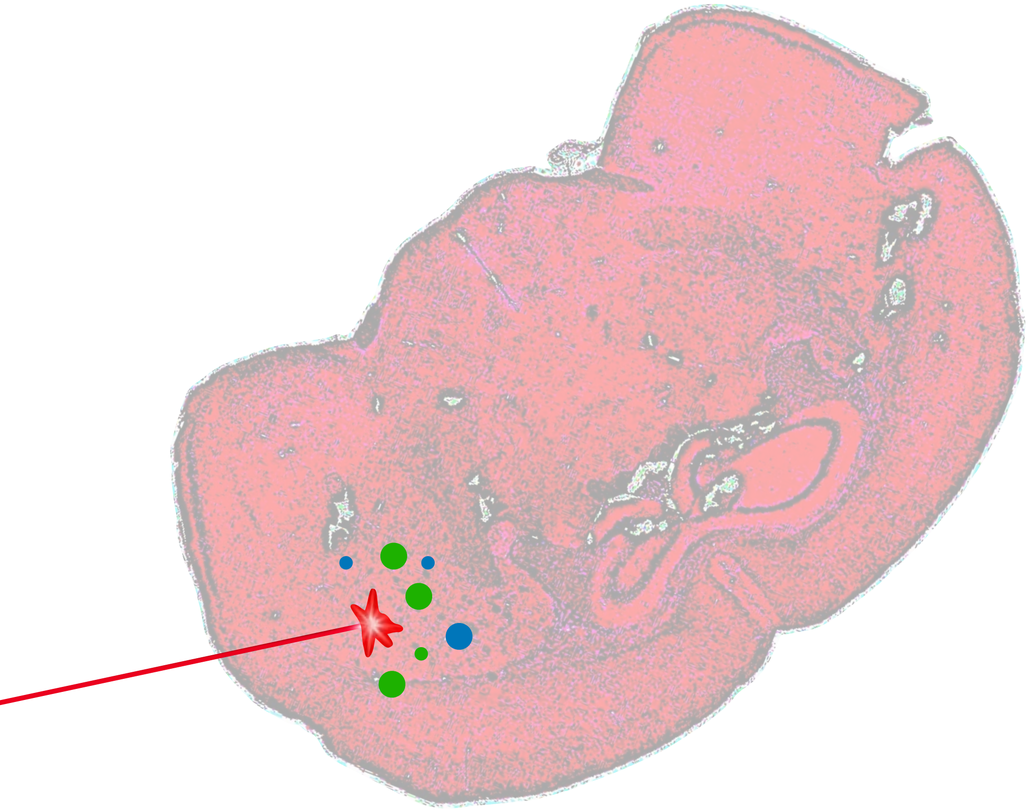
- We want to develop new statistical methods that take into account the information of these complex data

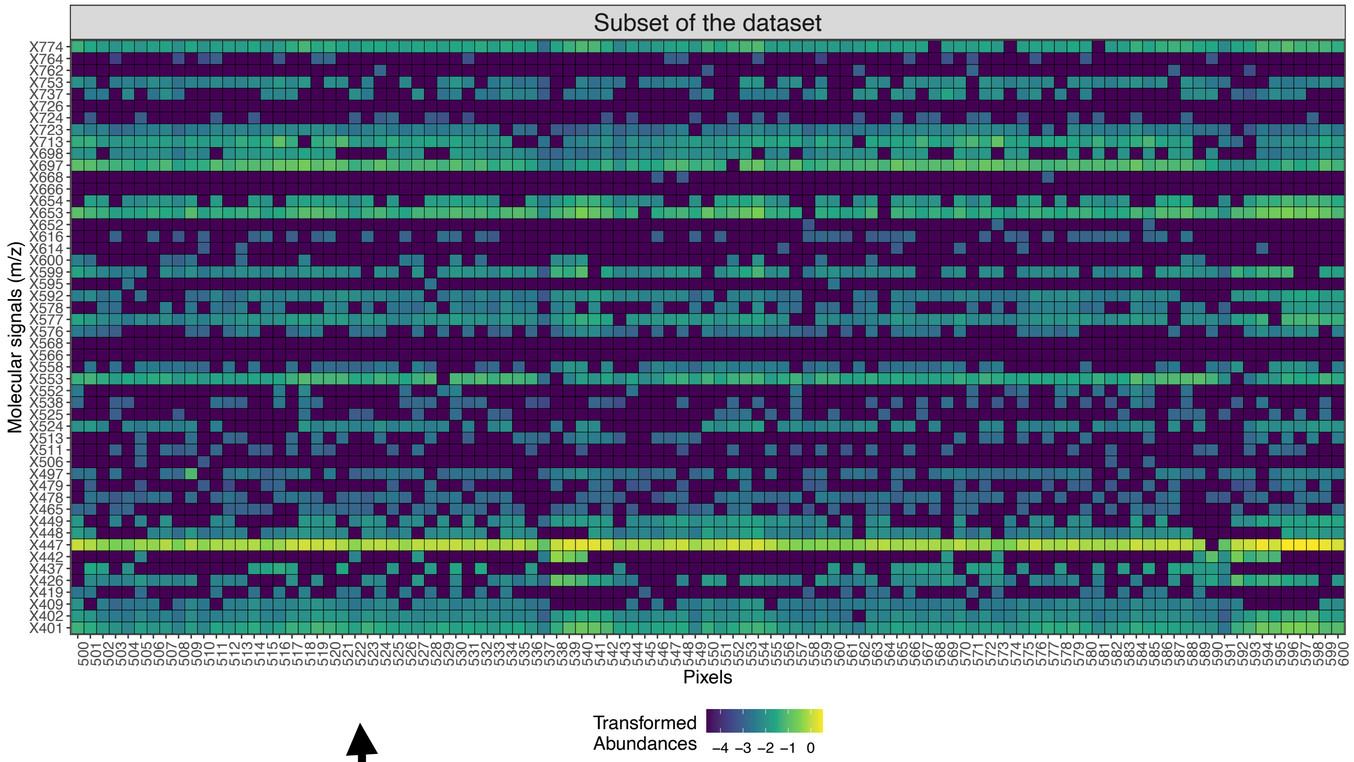


- The tissue is covered by a specific substance called *matrix*, which separates a specific molecule (here, **lipids**) from the rest and makes it “ready to be detected”
- A laser hits the tissue in a specific spot, and a mass spectrum is extracted (measuring signal abundance)
- Thus, the data can be organized into a large matrix. We have information for approximately more than **1500 pixels** (i.e., different locations)
- Each pixel contains the abundance for approximately **80 signals** (i.e., different lipids)
- Thus, data can be organized in a **large matrix**

Visual summary

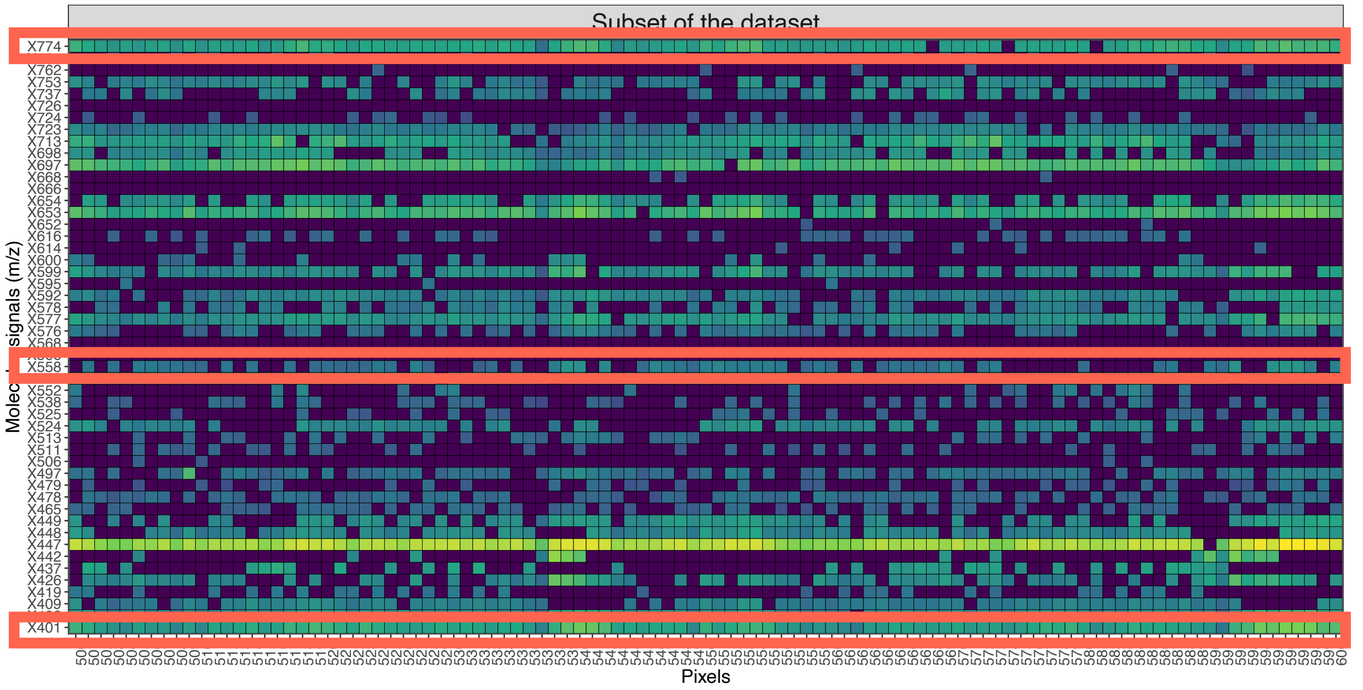
- The MALDI-MSI technique extracts the distributions of the abundance for different **lipid signals (rows)** in many **locations (pixels, columns)** given a biological sample

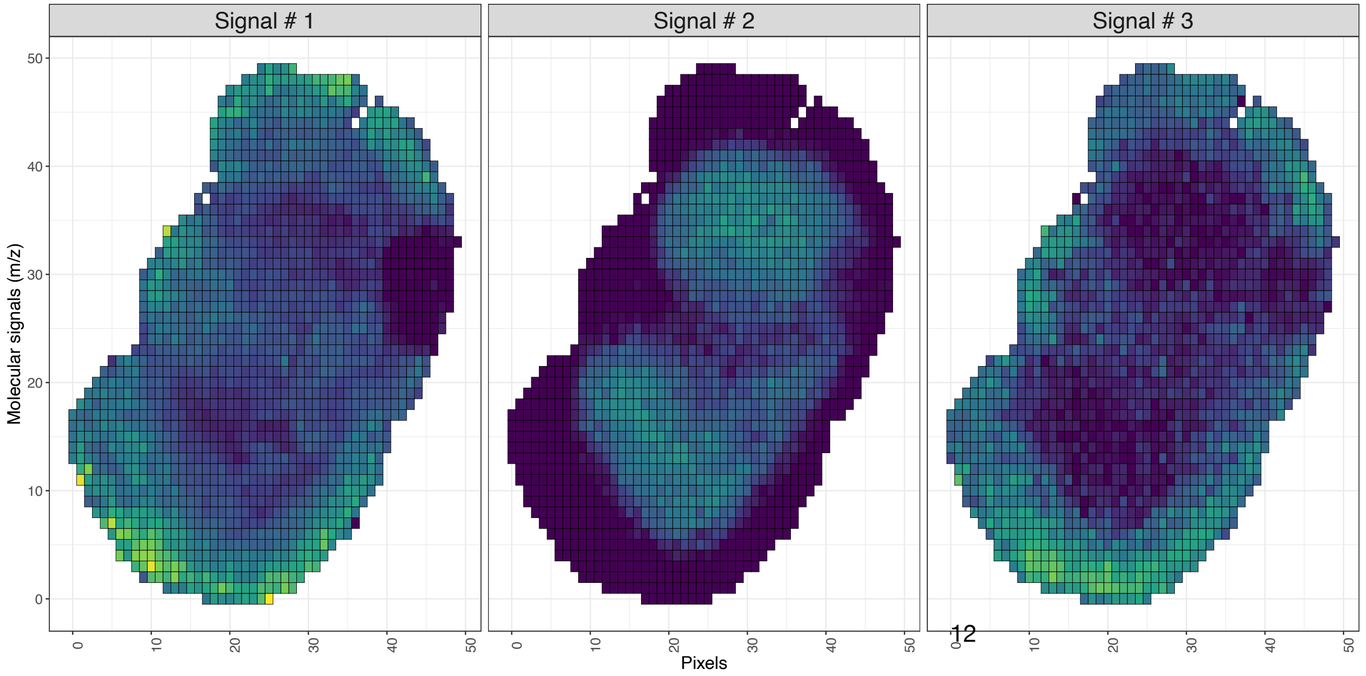




Visual summary

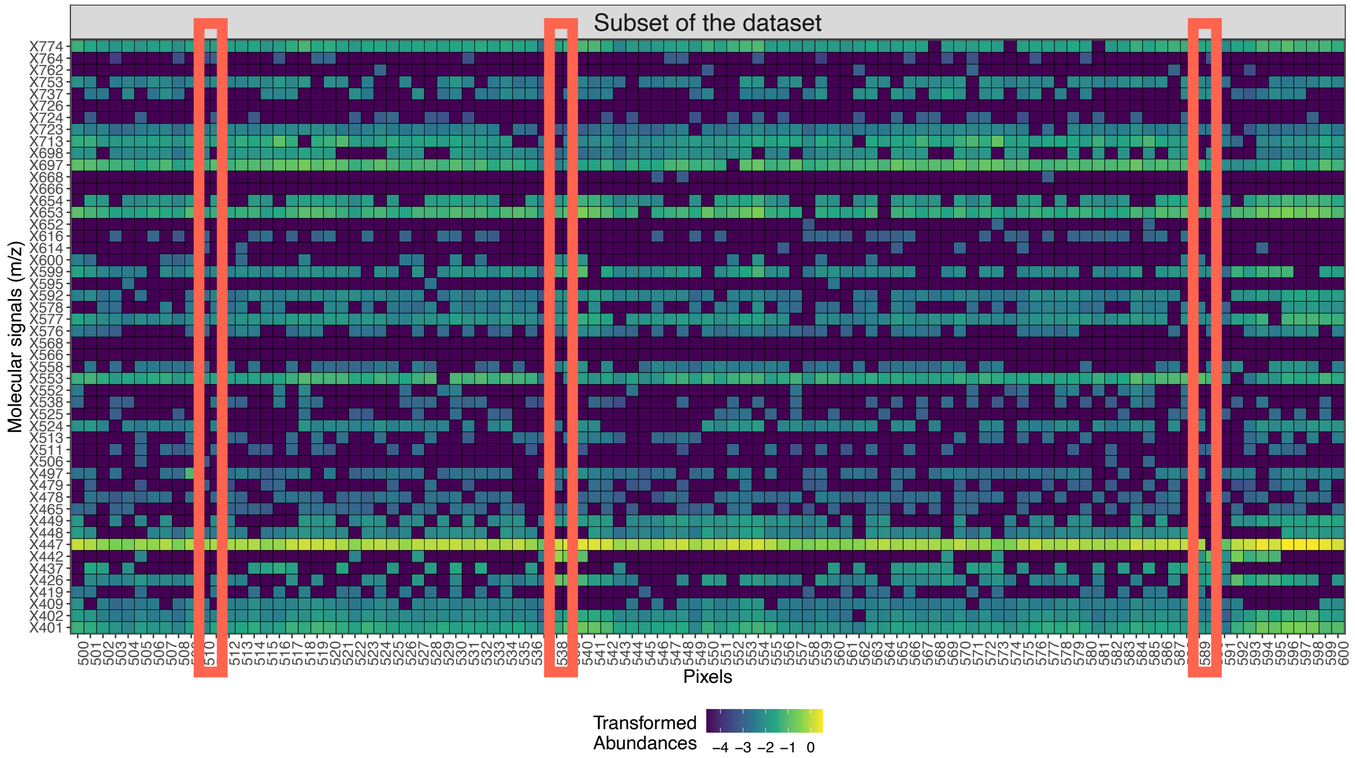
- For each lipid signal, we recover its expressions all over the brain slice
- There is spatial information to take into account!

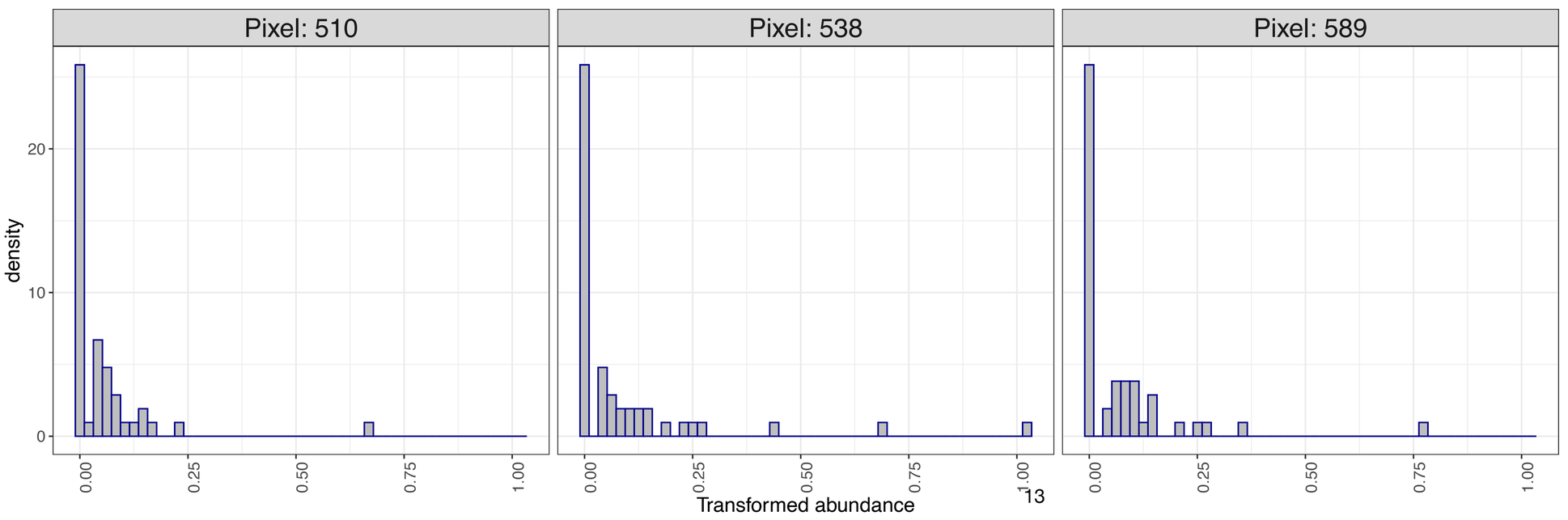




Visual summary

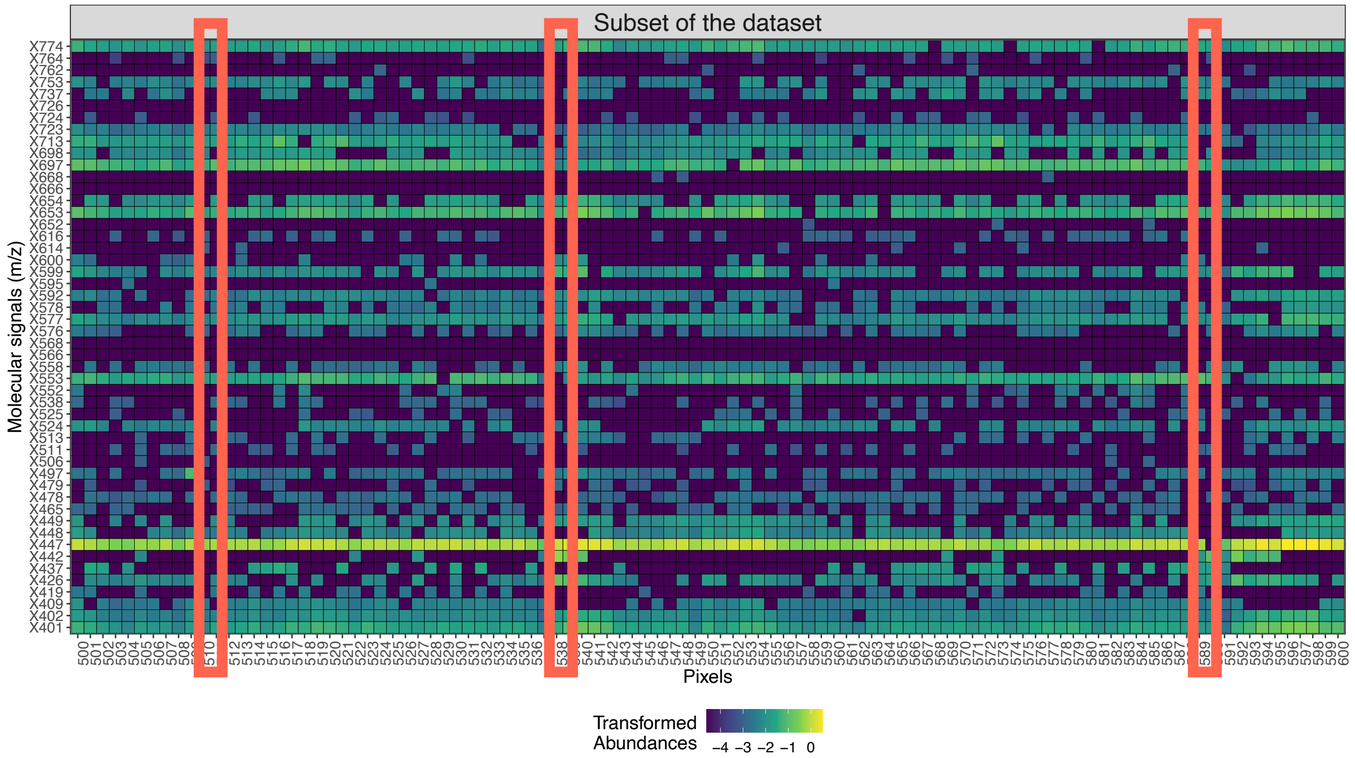
- Within each pixel, we observe a **distribution of abundances**
- Recall that the dataset contains around 1000 pixels that we want to cluster

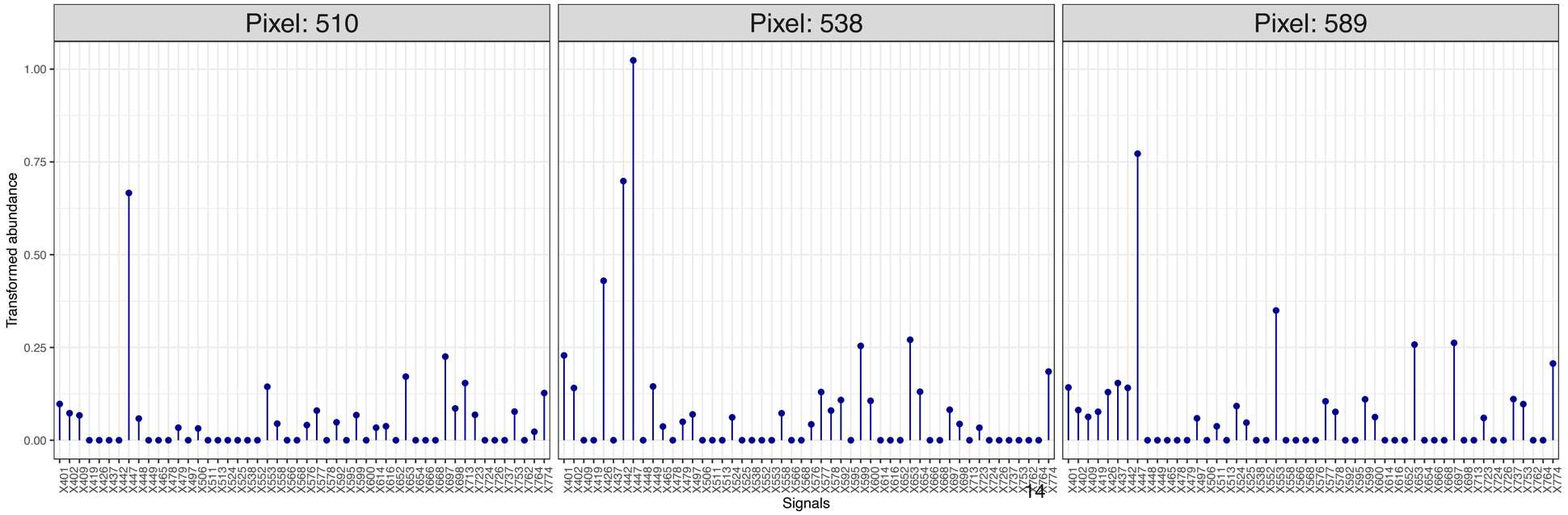




Visual summary

- Within each pixel, we observe an **abundance of particular signals**
- Recall that the dataset contains around 1000 pixels that we want to cluster

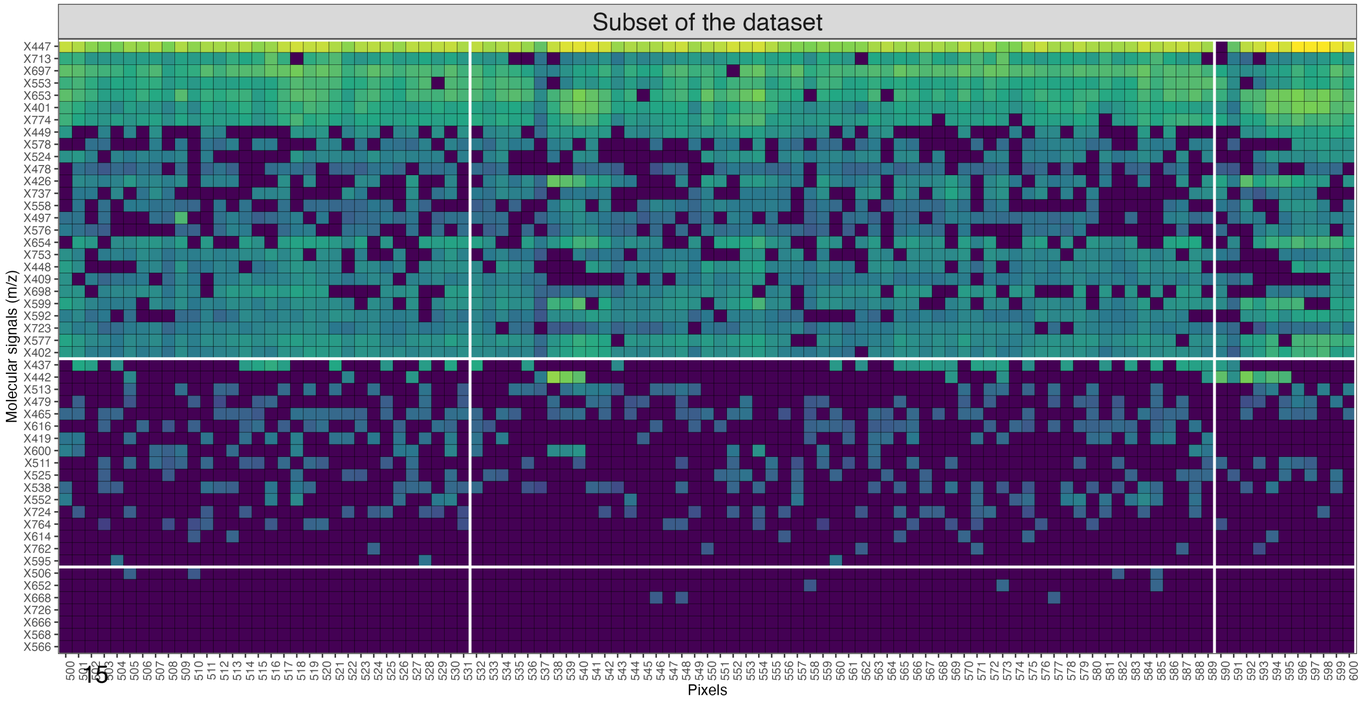




Biclustering algorithms

- We look for “two-dimensional” clustering
- **Biclustering** of the data: **simultaneously** group the pixels (columns) and, *within each column cluster*, group the different values of the signals (rows)
- The problem reminds us (or only me?) of the framework of **clustering grouped data**, for which lively literature has recently flourished (see, for example, Beraha et al. (2021), Balocchi et al. (2023), Camerlenghi et al. (2019), D’Angelo et al. (2023), Denti et al. (2023), Rebaudo et al. (2023), Rodriguez et al. (2011),…)
- All these models assume a **partially exchangeable** framework

heatmap() R function output on the subset dataset - simultaneous clustering of columns and rows

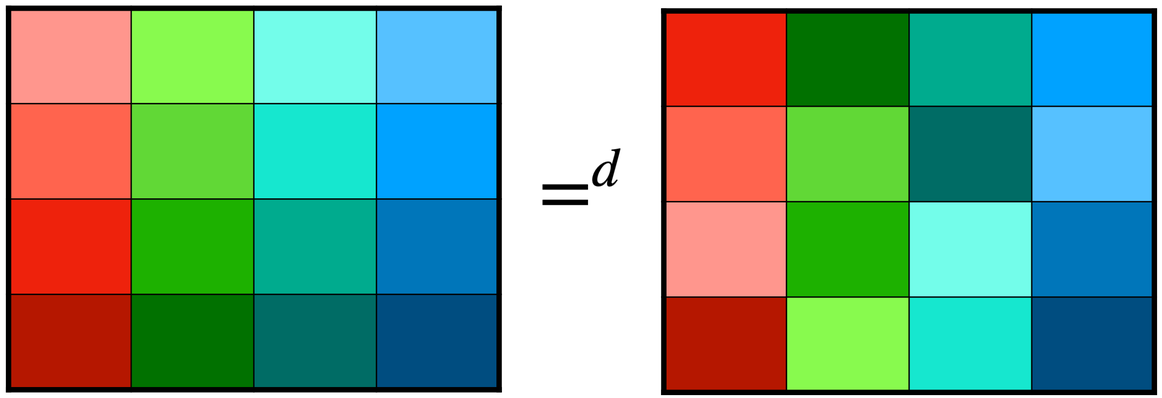


Partial exchangeability

- **Partially exchangeable data**: the observations are exchangeable within groups and conditionally independent across groups

- For partially exchangeable data, a **matrix structure is not needed**, as the framework **disregards the identities of the rows**

Partial Exchangeability



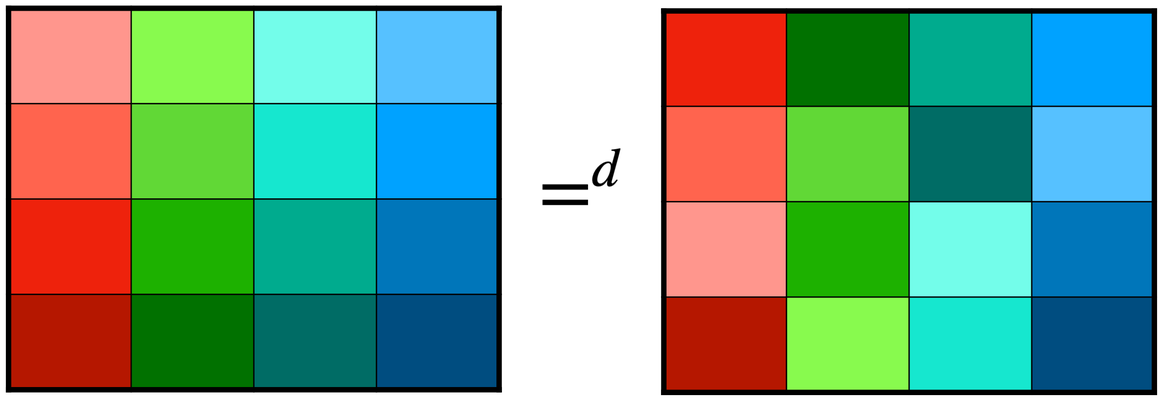
Partial exchangeability

- **Partially exchangeable data**: the observations are exchangeable within groups and conditionally independent across groups

- For partially exchangeable data, a **matrix structure is not needed**, as the framework **disregards the identities of the rows**

  - It's not ideal for this application, as we want to take the meaning of the rows (lipid signals) into account!

Partial Exchangeability



Separate exchangeability

- There are cases in which one may want to draw inference about the rows' characteristics: we use **separate exchangeability** (Rebaudo et al., 2021)

- Separate exchangeability is an ideal framework for modeling **matrices where rows and columns have specific meanings**: once the **columns** are clustered together, the model does not disregard any potential information conveyed by the **rows**

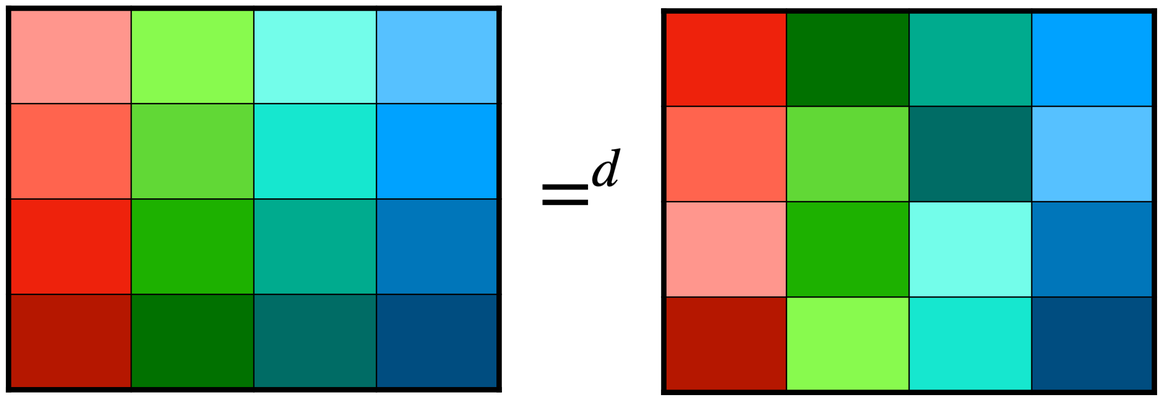
Statistics > Methodology

[Submitted on 14 Dec 2021 (v1), last revised 20 Jun 2024 (this version, v2)]

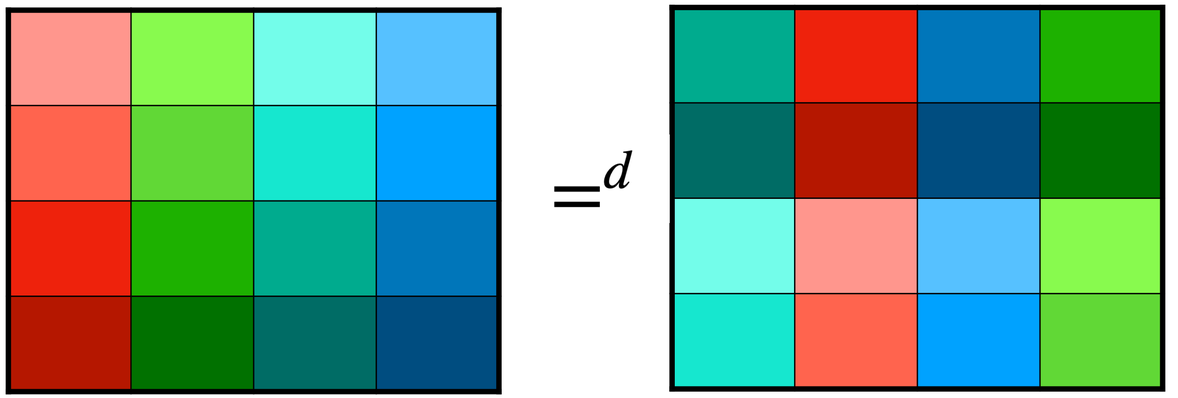
Separate Exchangeability as Modeling Principle in Bayesian Nonparametrics

Giovanni Rebaudo, Qiaohui Lin, Peter Mueller

Partial Exchangeability



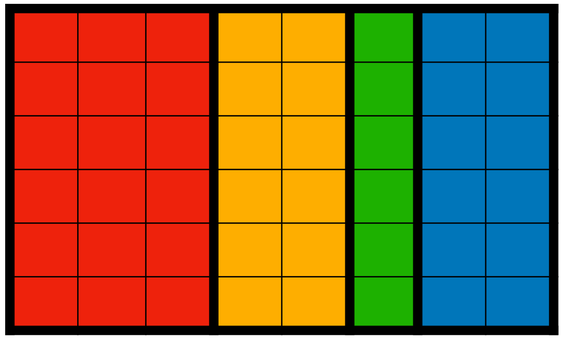
Separate Exchangeability



Separate Exchangeability through Nested Partitions

- Let $\boldsymbol{Y}$ indicate the MALDI-MSI abundance matrix, characterized by N rows (signals) and J columns (pixels)
- We need column-specific membership labels, denoted with $\boldsymbol{C} = \left(C_1, \ldots, C_J\right)$, where $C_j \in \{1,\ldots,K\}$ for $j = 1,\ldots,J$. We will refer to the values in $\boldsymbol{C}$ as **distributional cluster** (DC) labels.

$$p(\boldsymbol{C} \mid \boldsymbol{\pi}) = \prod_{j=1}^{J} \pi_{C_j}$$



Statistics > Methodology

[Submitted on 14 Dec 2021 (v1), last revised 20 Jun 2024 (this version, v2)]

Separate Exchangeability as Modeling Principle in Bayesian Nonparametrics

Giovanni Rebaudo, Qiaohui Lin, Peter Mueller

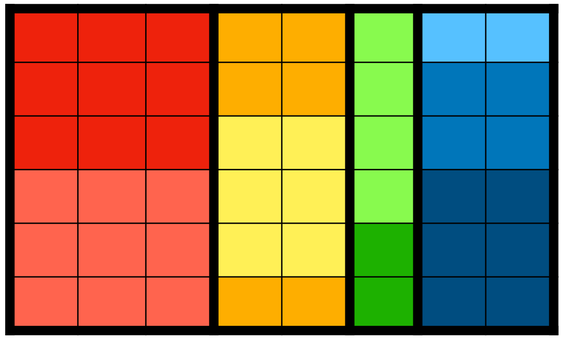
Separate Exchangeability through Nested Partitions

- Let $\boldsymbol{Y}$ indicate the MALDI-MSI abundance matrix, characterized by N rows (signals) and J columns (pixels)
- We need column-specific membership labels, denoted with $\boldsymbol{C} = \left(C_1, \ldots, C_J\right)$, where $C_j \in \{1,\ldots,K\}$ for $j = 1,\ldots,J$. We will refer to the values in $\boldsymbol{C}$ as **distributional cluster** (DC) labels

$$p(\boldsymbol{C} \mid \boldsymbol{\pi}) = \prod_{j=1}^{J} \pi_{C_j}$$

- We consider a set of membership labels for the rows *within every possible column cluster*, defining $\boldsymbol{R}_k = \left(R_{1,k}, \ldots, R_{N,k}\right)$, with $R_{i,k} \in \{1,\ldots,L\}$ for all $i = 1,\ldots,N$ and $k = 1,\ldots,K$.

$$p(\boldsymbol{R} \mid \boldsymbol{\omega}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \omega_{R_{i,k},k}$$



Statistics > Methodology

[Submitted on 14 Dec 2021 (v1), last revised 20 Jun 2024 (this version, v2)]

Separate Exchangeability as Modeling Principle in Bayesian Nonparametrics

Giovanni Rebaudo, Qiaohui Lin, Peter Mueller

Separate Exchangeability through Nested Partitions

- Let $\boldsymbol{Y}$ indicate the MALDI-MSI abundance matrix, characterized by N rows (signals) and J columns (pixels)
- We need column-specific membership labels, denoted with $\boldsymbol{C} = \left(C_1, \ldots, C_J\right)$, where $C_j \in \{1,\ldots,K\}$ for $j = 1,\ldots,J$. We will refer to the values in $\boldsymbol{C}$ as **distributional cluster** (DC) labels.

$$p(\boldsymbol{C} \mid \boldsymbol{\pi}) = \prod_{j=1}^{J} \pi_{C_j}$$

- We consider a set of membership labels for the rows *within every possible column cluster*, defining $\boldsymbol{R}_k = \left(R_{1,k}, \ldots, R_{N,k}\right)$, with $R_{i,k} \in \{1,\ldots,L\}$ for all $i = 1,\ldots,N$ and $k = 1,\ldots,K$.

$$p(\boldsymbol{R} \mid \boldsymbol{\omega}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \omega_{R_{i,k},k}$$

- Finally, the data distribution is

$$p(\boldsymbol{Y} \mid \boldsymbol{R}, \boldsymbol{C}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{j=1}^{J} f(y_{ij} \mid \theta_{R_{i,C_j}})$$

Statistics > Methodology

[Submitted on 14 Dec 2021 (v1), last revised 20 Jun 2024 (this version, v2)]

Separate Exchangeability as Modeling Principle in Bayesian Nonparametrics

Giovanni Rebaudo, Qiaohui Lin, Peter Mueller

Separate Exchangeability through Nested Partitions

- Let $\boldsymbol{Y}$ indicate the MALDI-MSI abundance matrix, characterized by N rows (signals) and J columns (pixels)
- We need column-specific membership labels, denoted with $\boldsymbol{C} = \left(C_1, \ldots, C_J\right)$, where $C_j \in \{1,\ldots,K\}$ for $j = 1,\ldots,J$. We will refer to the values in $\boldsymbol{C}$ as **distributional cluster** (DC) labels.

$$p(\boldsymbol{C} \mid \boldsymbol{\pi}) = \prod_{j=1}^{J} \pi_{C_j}$$

- We consider a set of membership labels for the rows *within every possible column cluster*, defining $\boldsymbol{R}_k = \left(R_{1,k}, \ldots, R_{N,k}\right)$, with $R_{i,k} \in \{1,\ldots,L\}$ for all $i = 1,\ldots,N$ and $k = 1,\ldots,K$.

$$p(\boldsymbol{R} \mid \boldsymbol{\omega}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \omega_{R_{i,k},k}$$

- Finally, the data distribution is

$$p(\boldsymbol{Y} \mid \boldsymbol{R}, \boldsymbol{C}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \prod_{j=1}^{J} f(y_{ij} \mid \theta_{R_{i,C_j}})$$

The column cluster is the second index of the row cluster! The row cluster is shared across all the elements of the column cluster!

Reminiscent of the biclustering model by Lee et al. (2013)

Presence of **Common Atoms**, as in Denti et al. (2023) and Chandra et al. (2023)

Statistics > Methodology

[Submitted on 14 Dec 2021 (v1), last revised 20 Jun 2024 (this version, v2)]

Separate Exchangeability as Modeling Principle in Bayesian Nonparametrics

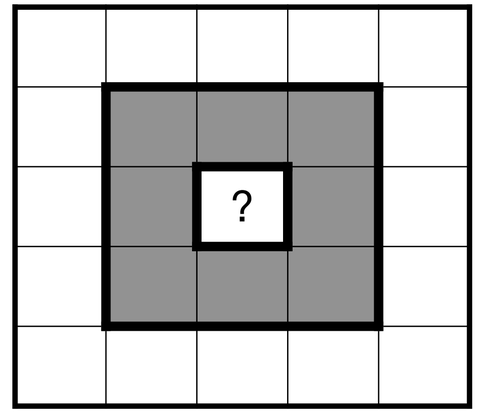
Giovanni Rebaudo, Qiaohui Lin, Peter Mueller

Including spatial information in the model

- Include in the model the spatial correlation that may incur across pixels
- To do so, we use **hidden Markov random fields** (Besag, 1986) to specify a clustering distribution over the columns
- In particular, we use a **Potts model** with K potential clusters

$$p(C_j = k \mid C_{-j}) \propto \exp\left[\beta \sum_{q \sim \mathcal{N}_j} \mathbb{1}_{\{C_q = k\}}\right], \quad k = 1, \ldots, K$$

- β is the inverse temperature parameter (crucial, but hard to estimate!) and $\mathcal{N}_j$ denotes the neighborhood of pixel j
- The larger the inverse temperature, the higher the probability of a column to be in the same cluster of its neighbors

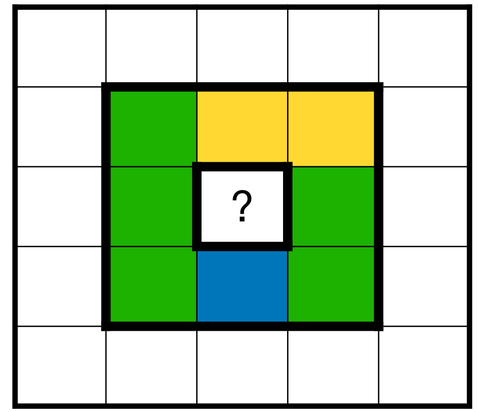


Including spatial information in the model

- Include in the model the spatial correlation that may incur across pixels
- To do so, we use **hidden Markov random fields** (Besag, 1986) to specify a clustering distribution over the columns
- In particular, we use a **Potts model** with K potential clusters

$$p(C_j = k \mid C_{-j}) \propto \exp\left[\beta \sum_{q \sim \mathcal{N}_j} \mathbb{1}_{\{C_q = k\}}\right], \quad k = 1, \ldots, K$$

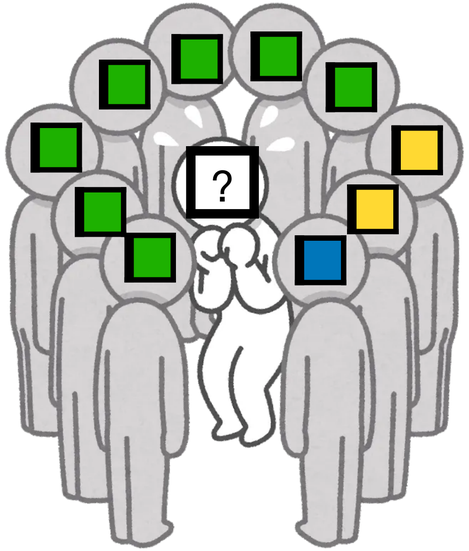
- β is the inverse temperature parameter (crucial, but hard to estimate!) and $\mathcal{N}_j$ denotes the neighborhood of pixel j
- The larger the inverse temperature, the higher the probability of a column to be in the same cluster of its neighbors



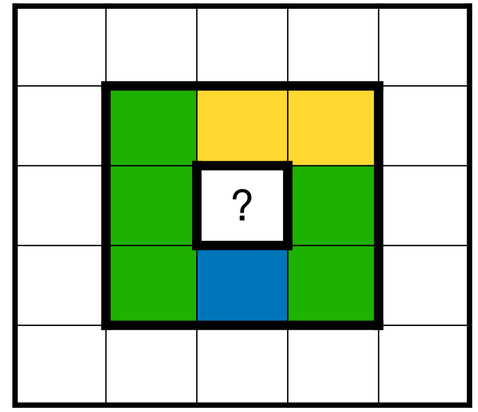
Including spatial information in the model

- Include in the model the spatial correlation that may incur across pixels
- To do so, we use **hidden Markov random fields** (Besag, 1986) to specify a clustering distribution over the columns
- In particular, we use a **Potts model** with K potential clusters

$$p(C_j = k \mid C_{-j}) \propto \exp\left[\beta \sum_{q \sim \mathcal{N}_j} \mathbb{1}_{\{C_q = k\}}\right], \quad k = 1, \ldots, K$$



https://medium.com/@takashifurukawa1122
Yes, there are more green pixels here, sorry



- β is the inverse temperature parameter (crucial, but hard to estimate!) and $\mathcal{N}_j$ denotes the neighborhood of pixel j
- The larger the inverse temperature, the higher the probability of a column to be in the same cluster of its neighbors

Including spatial information in the model

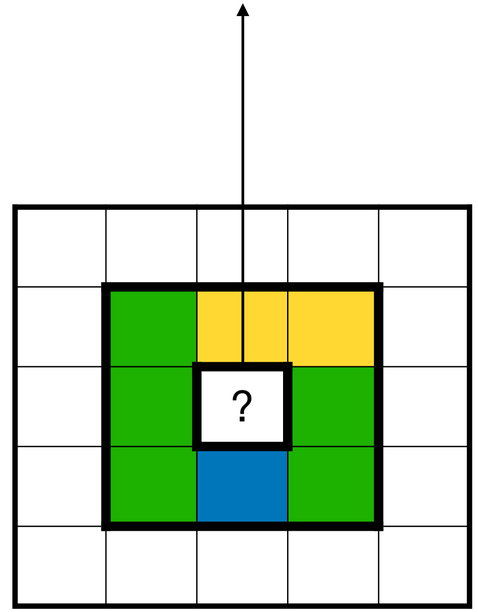
- Include in the model the spatial correlation that may incur across pixels
- To do so, we use **hidden Markov random fields** (Besag, 1986) to specify a clustering distribution over the columns
- In particular, we use a **Potts model** with K potential clusters

Prob of green $\propto \exp\left[5\beta\right]$

Prob of blue $\propto \exp\left[\beta\right]$

Prob of yellow $\propto \exp\left[2\beta\right]$

$$p(C_j = k \mid C_{-j}) \propto \exp\left[\beta \sum_{q \sim \mathcal{N}_j} \mathbb{1}_{\{C_q = k\}}\right], \quad k = 1, \ldots, K$$



- β is the inverse temperature parameter (crucial, but hard to estimate!) and $\mathcal{N}_j$ denotes the neighborhood of pixel j
- The larger the inverse temperature, the higher the probability of a column to be in the same cluster of its neighbors

Including spatial information in the model

- We can also extend the basic Potts model with the **BNP-HMRF prior** by Lü et al. (2021)

- Clever idea: adding as **first-order potentials** in the energy functions the element of a **log-stick breaking sequence**, combining two types of weights

Article · Publisher preview available

Bayesian nonparametric priors for hidden Markov random fields

July 2020 · Statistics and Computing 30(11) · Follow journal

DOI: 10.1007/s11222-020-09935-9

Hongliang Lü · Julyan Arbel · Florence Forbes

First order potentials

Second order potentials

$$p(\boldsymbol{C} \mid \boldsymbol{\pi}) = \frac{1}{\mathcal{K}(\boldsymbol{\pi}, \beta)} \exp\left[\sum_{j=1}^{J} \log(\pi_{C_j}) + \beta \sum_{q \sim \mathcal{N}_j} \mathbb{1}_{\{C_q = C_j\}}\right]$$

- **Key point**: the distribution above is valid without specifying the number of clusters. Thus, $\boldsymbol{\pi}$ can be specified non-parametrically as a Dirichlet Process or Pitman-Yor process

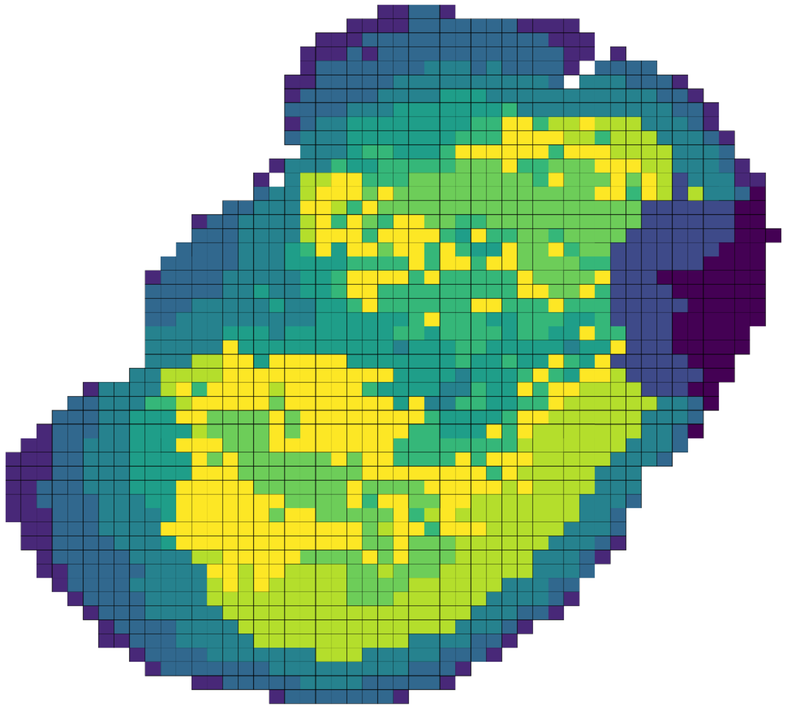
- Adding $\boldsymbol{\pi}$ allows to estimate the number of clusters nonparametrically

- We are also exploring the performance using sparse finite mixtures

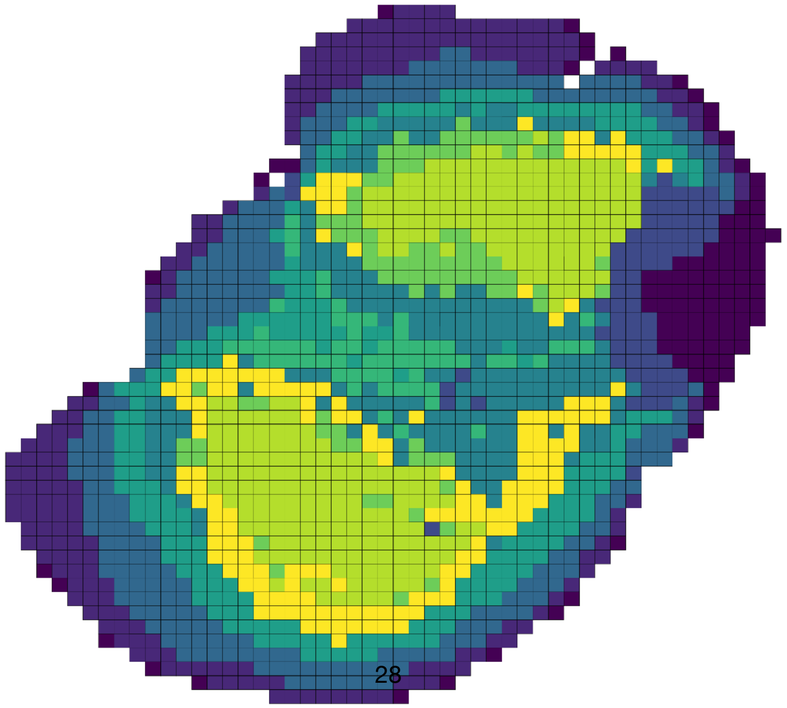
Results

- We developed an efficient CAVI algorithm to perform fast mean-field variational inference
- The model runs in a matter of minutes - R packages are under development
- Really promising results:

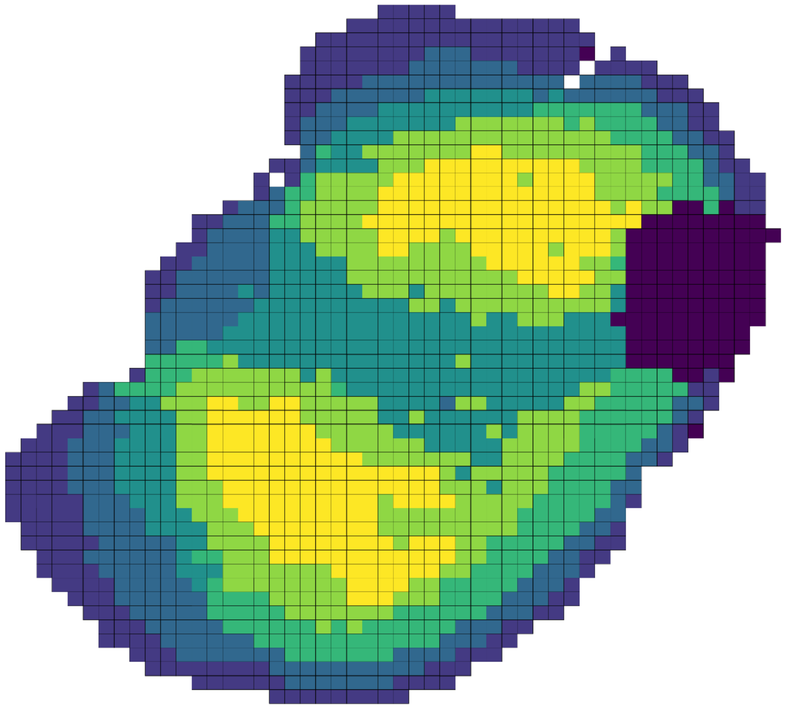
K-means++



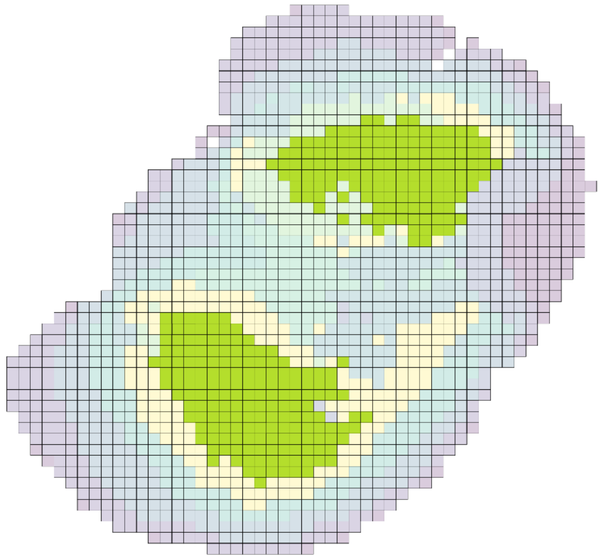
Biclustering model

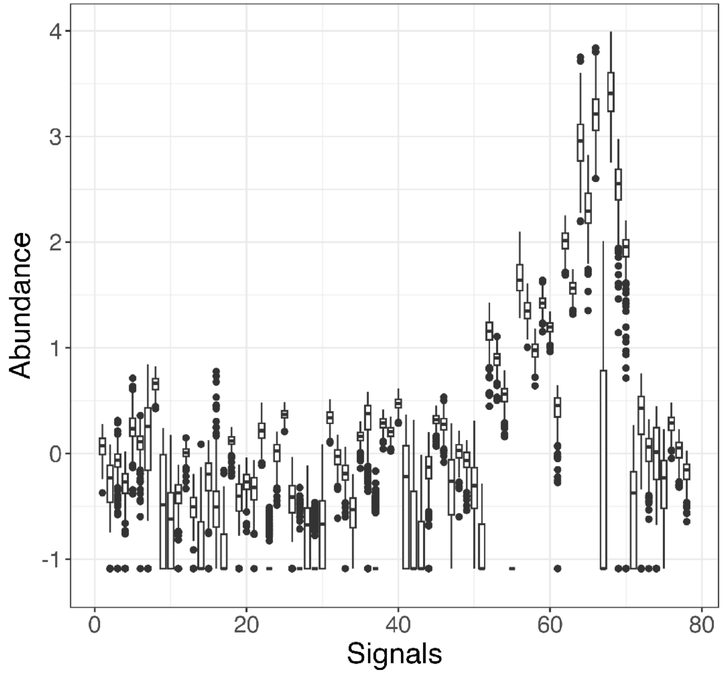


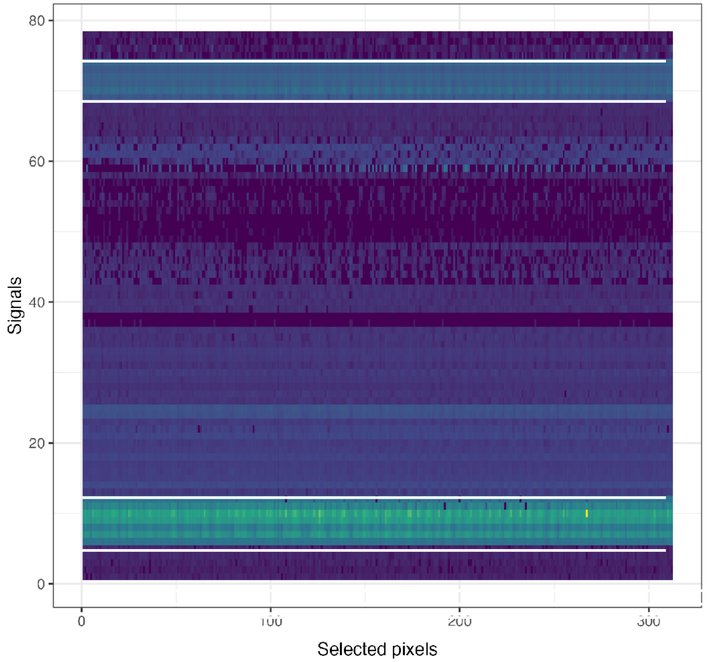
Biclustering model w/ HMRF

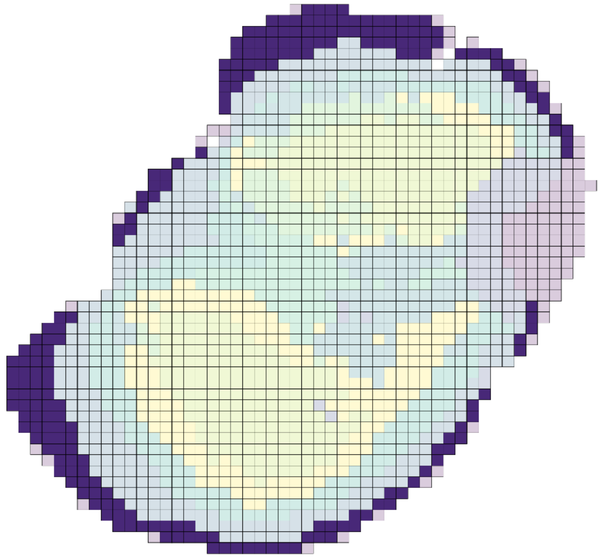


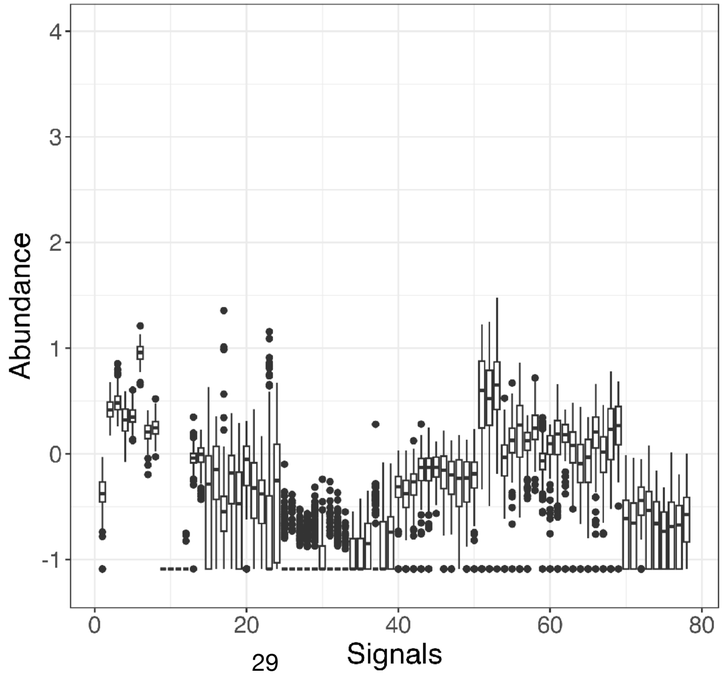
Comparing the content of two column clusters

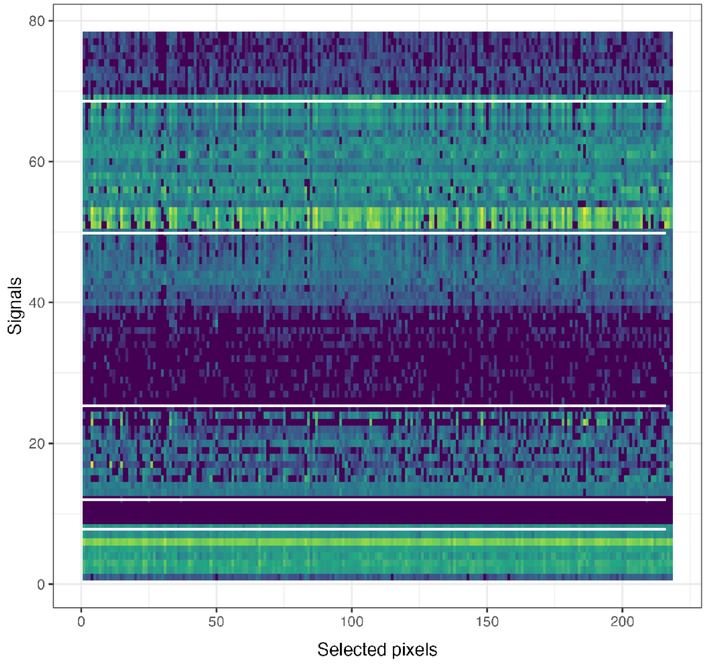












Rinse and repeat! There is more than just lipids out there

- Denti et al. 2022, introduced a laboratory protocol that enables the **cleaning of the sample** from the matrix used after an initial MALDI analysis.
    - This step is essential to removing residues from the initial matrix, which was applied to break specific bonds. Thus, the next analysis can take place.

Denti et al., 2022, “Spatial Multiomics of Lipids, N-Glycans, and Tryptic Peptides on a Single FFPE Tissue Section”, Journal of Proteome Research

Rinse and repeat! There is more than just lipids out there

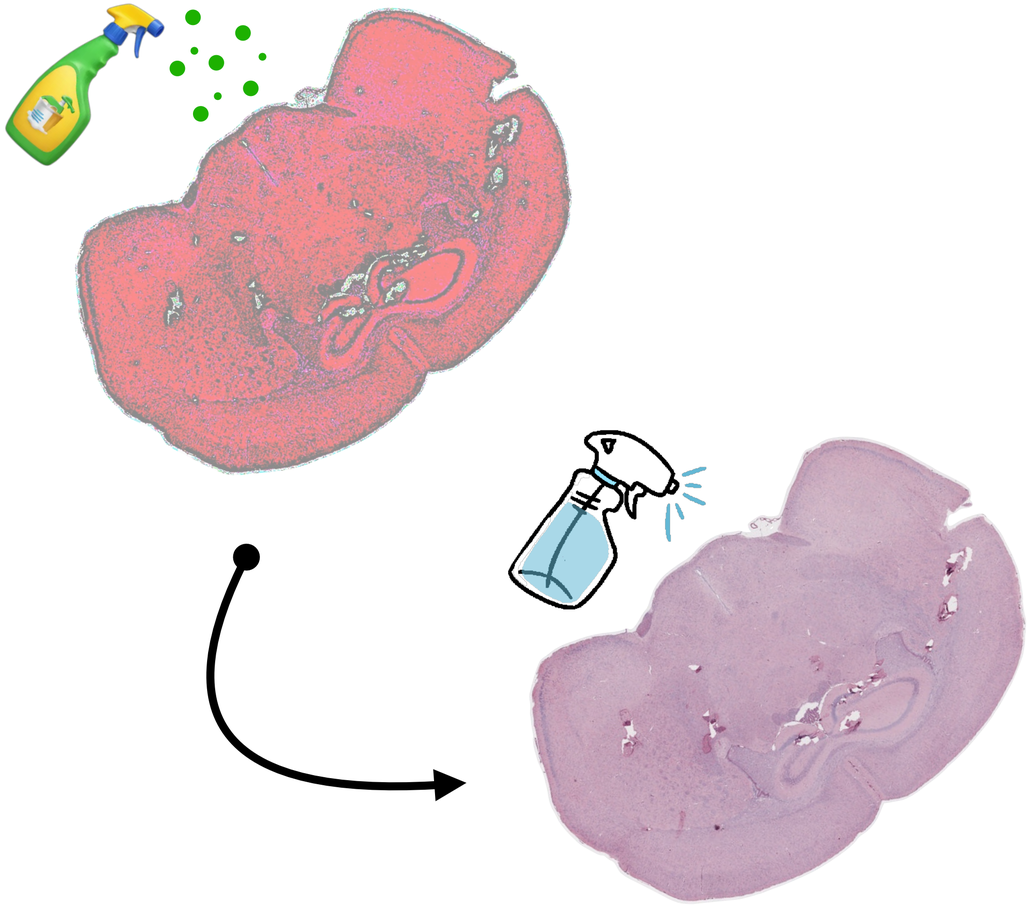
- Denti et al. 2022, introduced a laboratory protocol that enables the **cleaning of the sample** from the matrix used after an initial MALDI analysis.
  - This step is essential to removing residues from the initial matrix, which was applied to break specific bonds. Thus, the next analysis can take place.

DISCLAIMER: I did not invent anything here,
the first author is simply a very smart homonymous

Denti et al., 2022, “Spatial Multiomics of Lipids, N-Glycans, and Tryptic Peptides on a Single FFPE Tissue Section”, Journal of Proteome Research

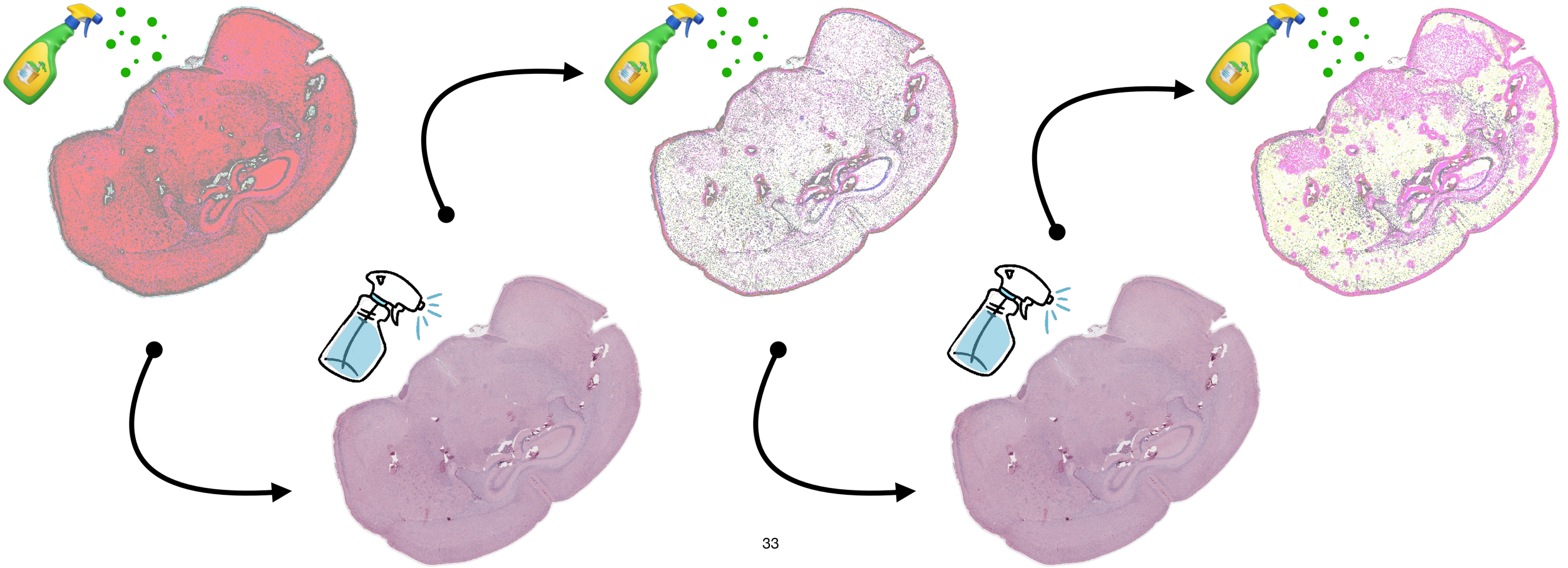
Rinse and repeat! There is more than just lipids out there

- Denti et al. 2022, introduced a laboratory protocol that enables the **cleaning of the sample** from the matrix used after an initial MALDI analysis.
  - This step is essential to removing residues from the initial matrix, which was applied to break specific bonds. Thus, the next analysis can take place.



Rinse and repeat! There is more than just lipids out there

- Once "washed," a new matrix can be applied to the sample, designed to break a different type of bond and extract other classes of molecules, such as lipids.
  - The extraction order follows a biological rationale in which molecular bonds are broken: lipids, glycans, and finally, peptides.

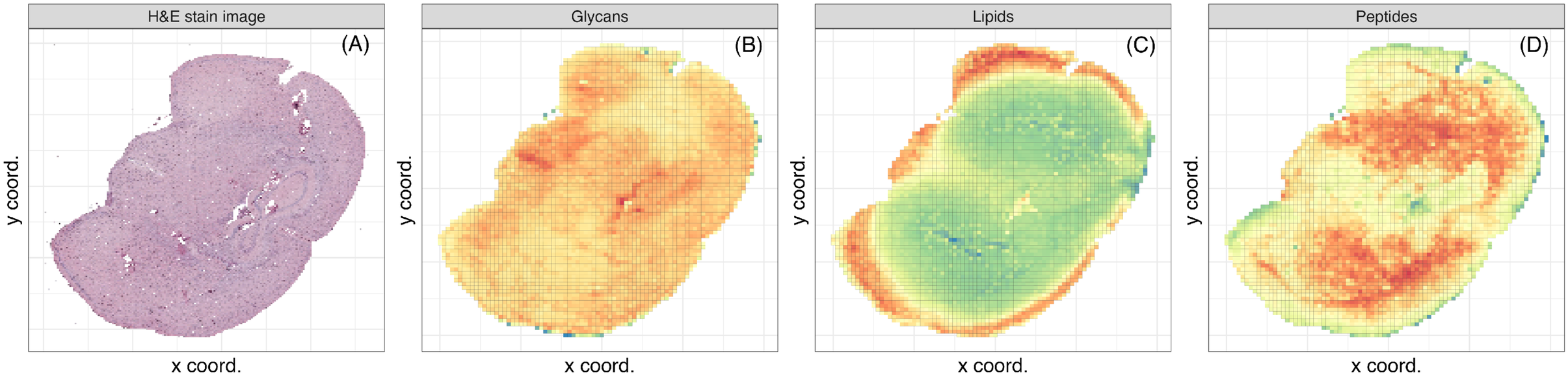


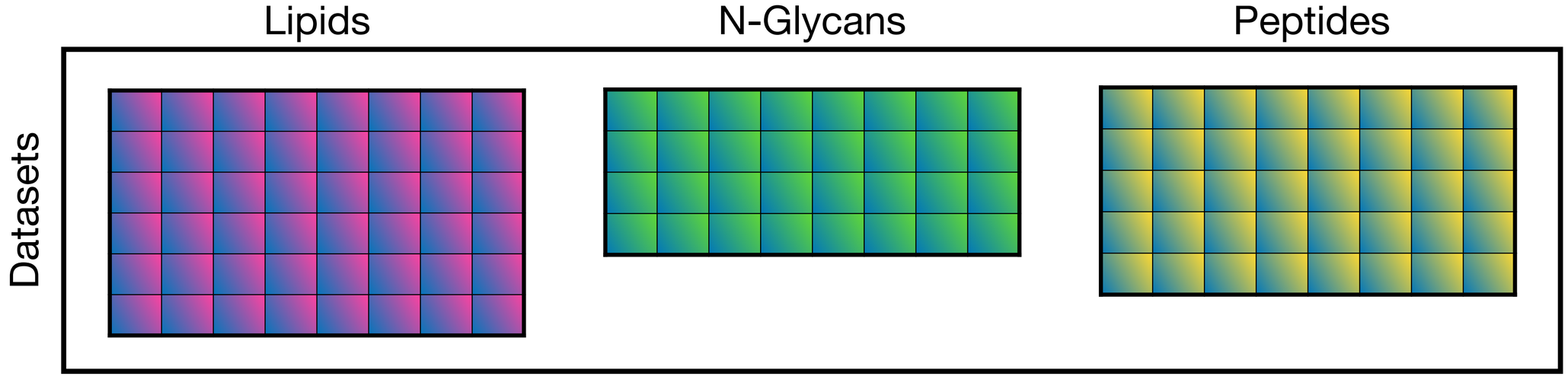
Next step: Poseidon

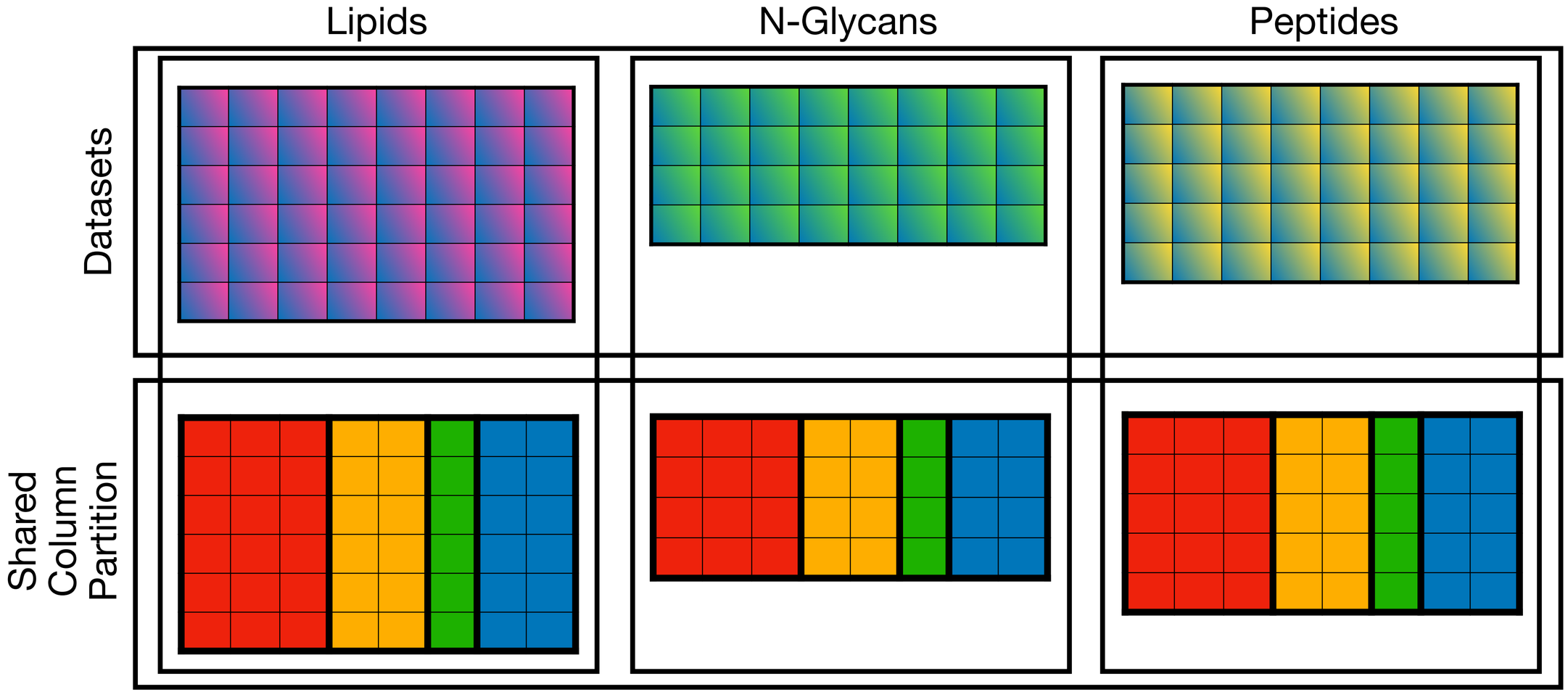
- Our data are, in reality, even more complex: the MALDI-MSI extracts datasets for **three different molecules from the same tissue**: lipids, N-glycans, peptides

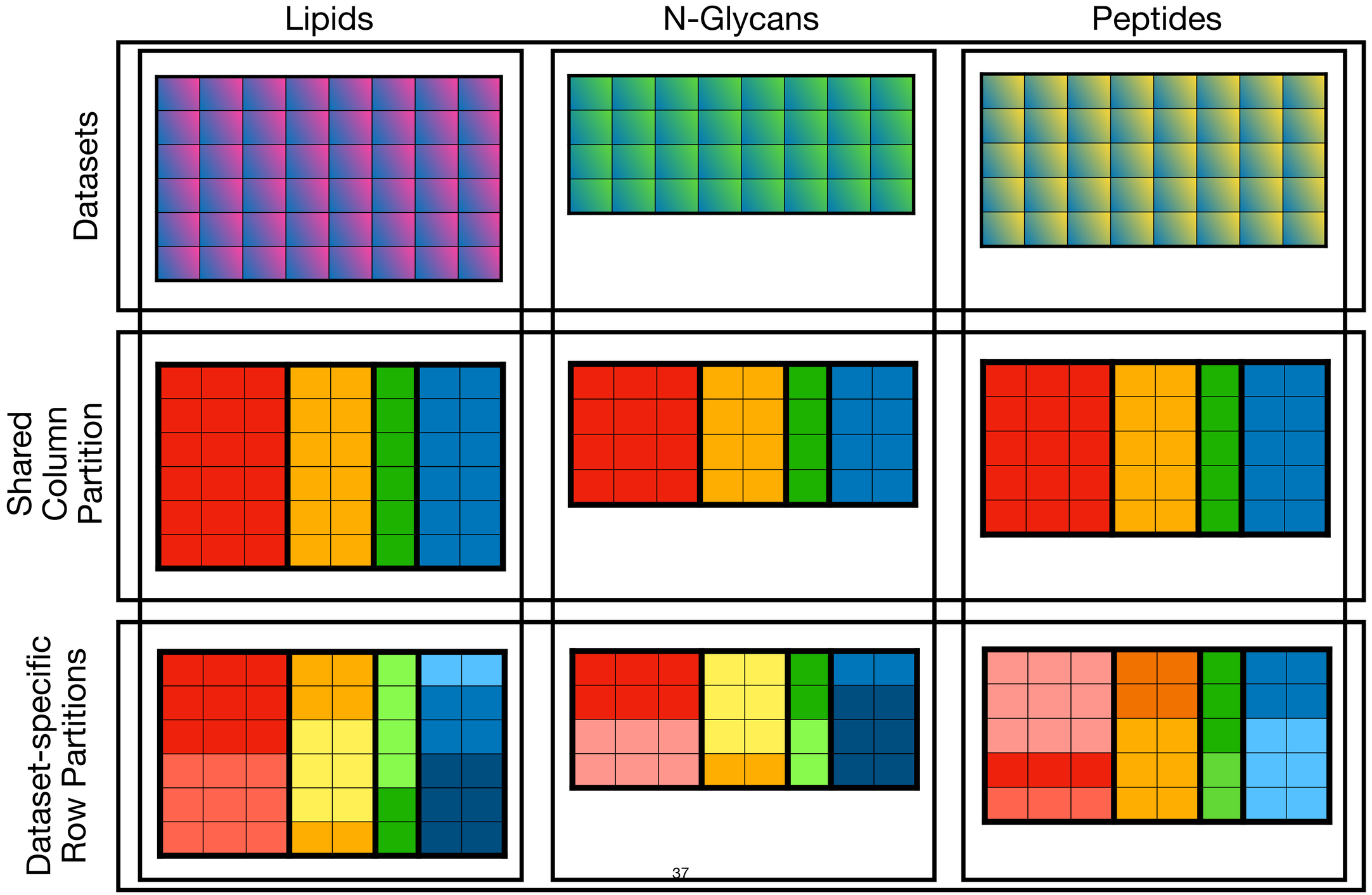
- They all **share the same set of pixels**, and potentially have a different number of signals

- We are developing **POSEIDON**: **POtts model** over **S**eparate **E**xchangeability **Integrating** **D**ifferent **O**mics **N**ested data to perform image segmentation led by multiple sources



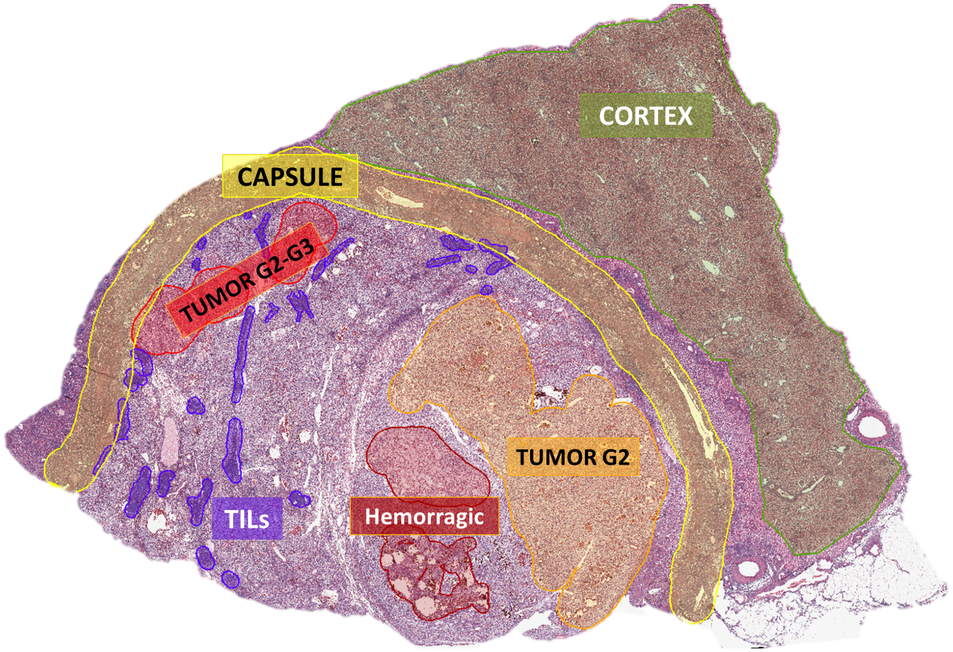

Lipids
N-Glycans
Peptides
Datasets

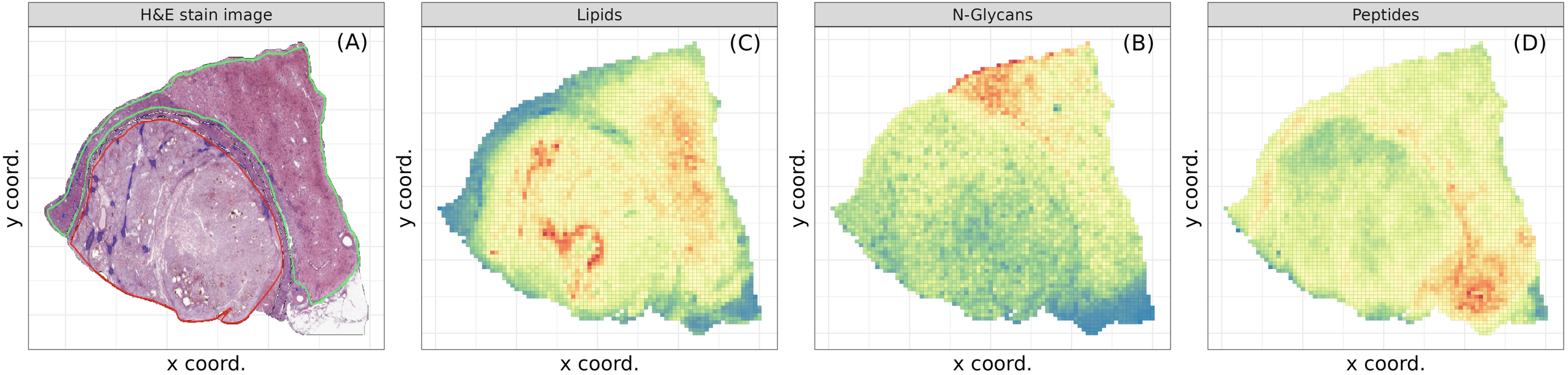

Lipids
N-Glycans
Peptides
Datasets
Shared
Column
Partition


Lipids
N-Glycans
Peptides
Datasets
Shared Column Partition
Dataset-specific Row Partitions

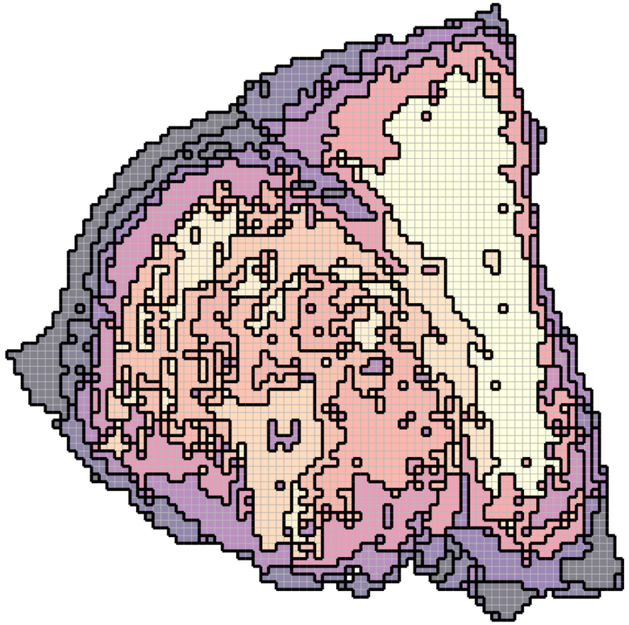
Next step: Poseidon on ccRCC data

- A more compelling application:
  - Clear cell Renal Cell Carcinoma
  - Lipids, N-glycans, and Peptides

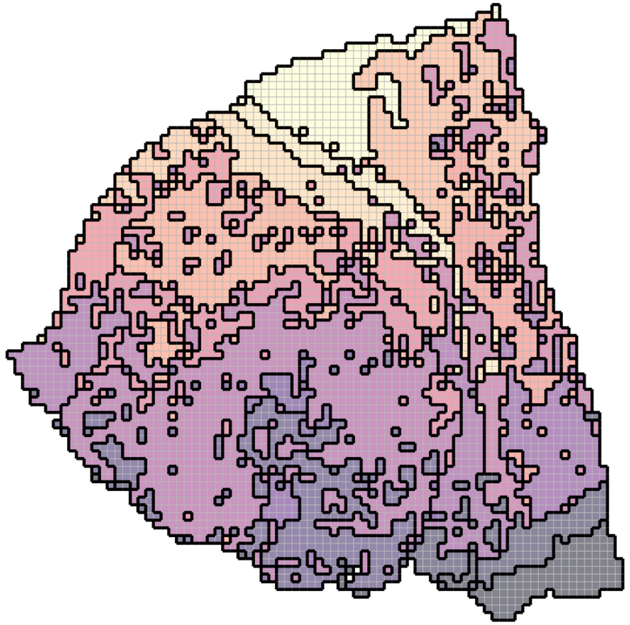




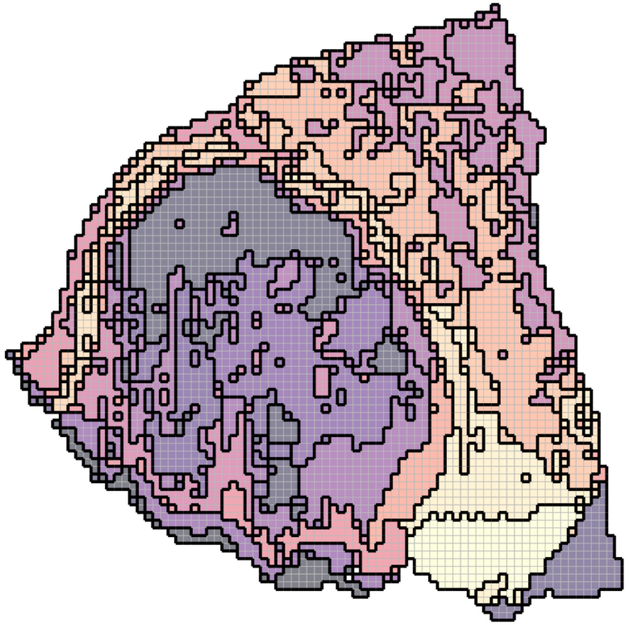
(A) Lipids



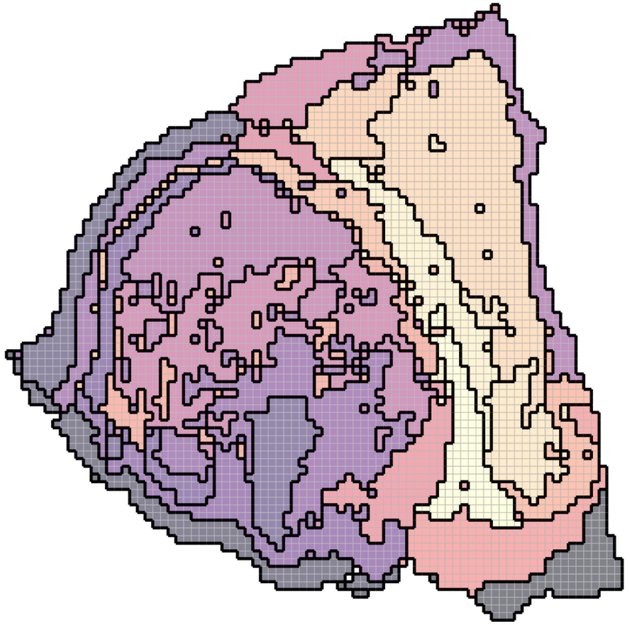
(B) N-Glycans



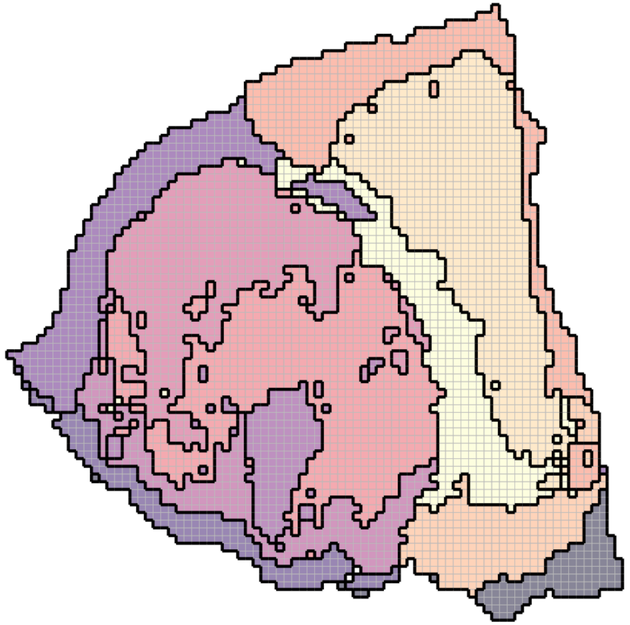
(C) Peptides

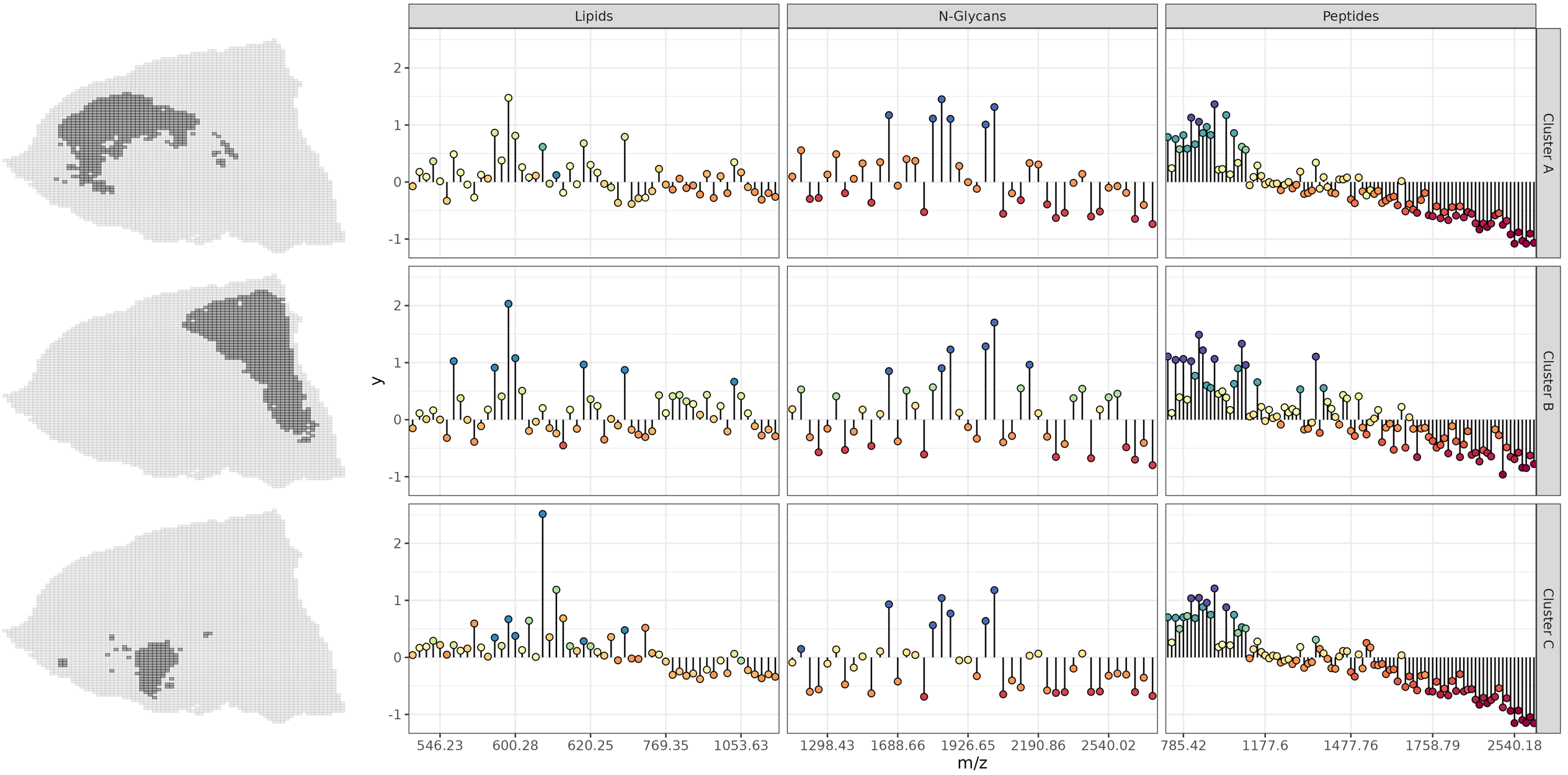


(D) Pose - Stacked datasets



(E) Poseidon




Lipids
N-Glycans
Peptides
Cluster A
Cluster B
Cluster C
y
2
1
0
-1
546.23
600.28
620.25
769.35
1053.63
1298.43
1688.66
1926.65
2190.86
2540.02
785.42
1177.6
1477.76
1758.79
2540.18
m/z

Calcium imaging

A Bayesian nonparametric model for

Simultaneous **C**lustering and
Deconvolution of **C**alcium traces

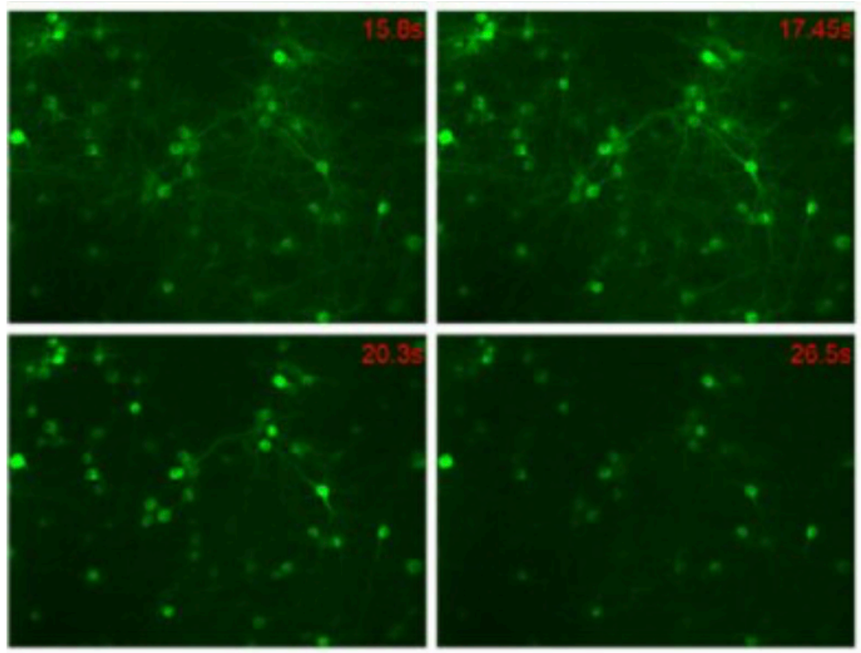
Calcium imaging

A Bayesian nonparametric model for

Simultaneous **C**lustering and
Deconvolution of **C**alcium traces

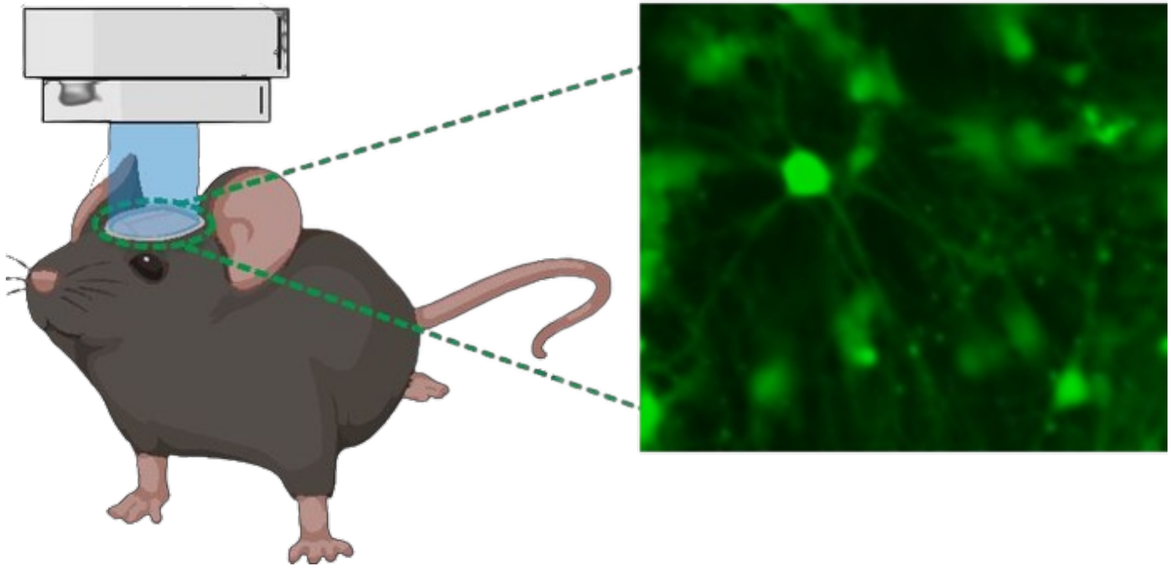
Mapping Neuronal Activity in Real Time

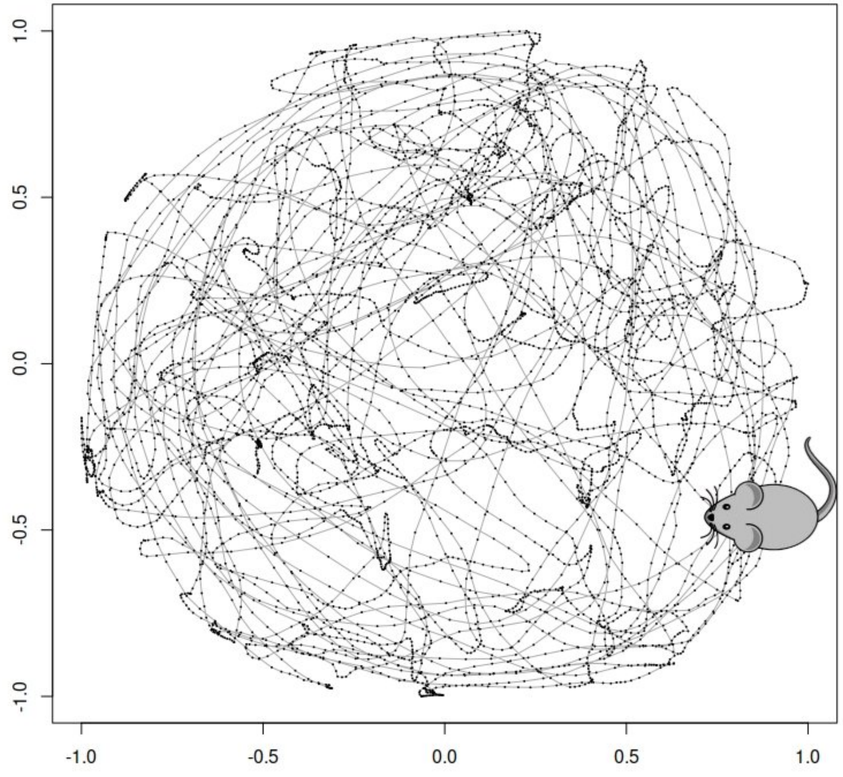
- **Neuroimaging** uses advanced physiological mechanisms to **visually capture and measure brain function**.
  - Reveals how individuals and **groups of neurons respond** during tasks.
  - It helps decode how neurons work together to produce cognition, behavior, and perception
  - While the anatomical structure is well-known, the **functional properties** are still unknown.
  - Neurons operate at incredible speed - tracking their real-time activity is complex but crucial for breakthroughs in neuroscience.



Calcium imaging measurements

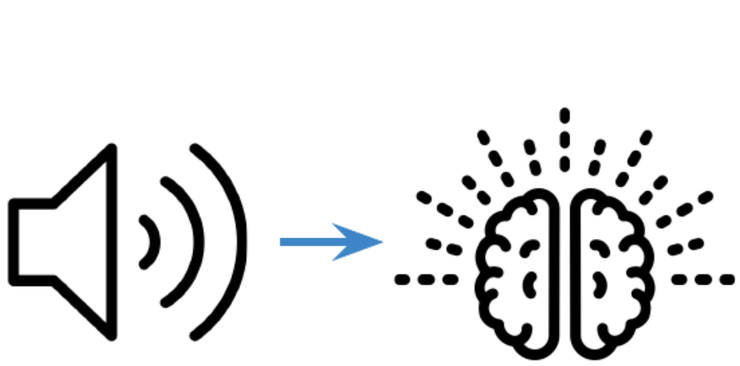
- **Miniaturized Microscopes**: Compact, high-resolution imaging systems
  - Allows real-time neural recording in freely moving subjects
- **Fluorescent Calcium Indicators**: Chemical markers that bind to calcium ions during neuronal activation
  - Enables precise measurement of the activity of an individual neuron

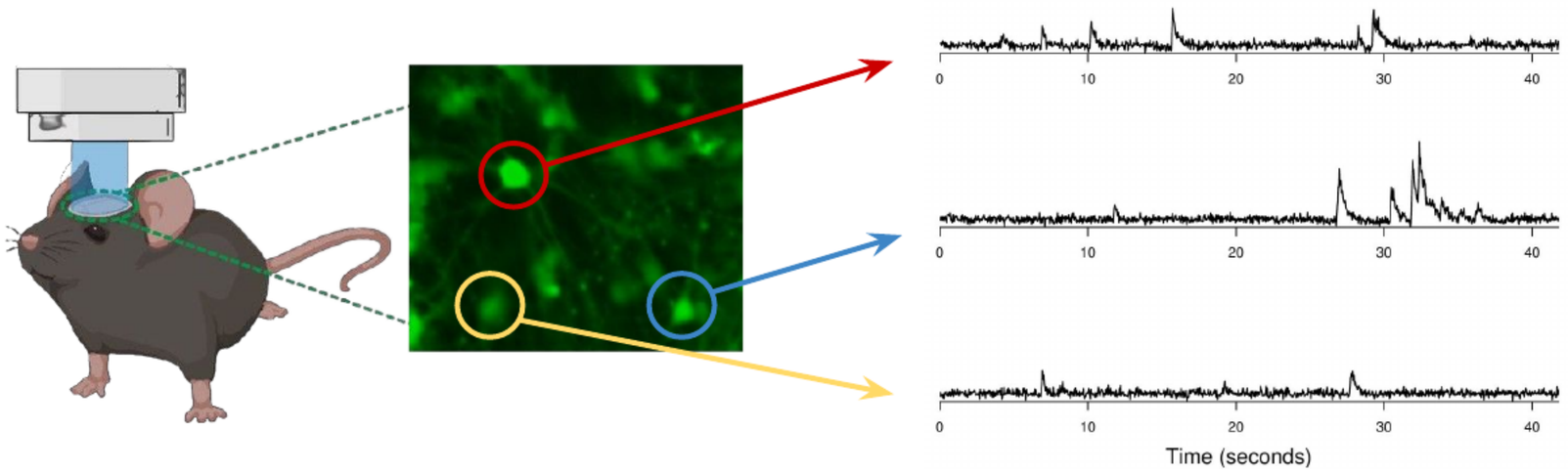




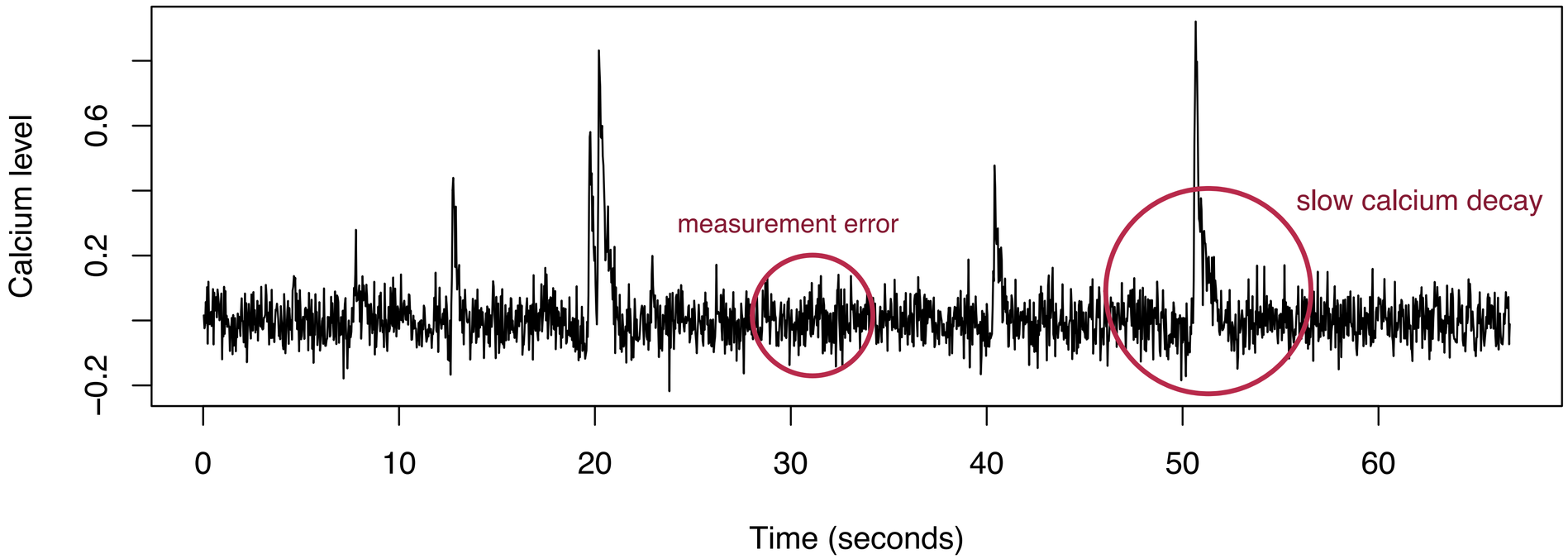
Calcium imaging measurements

- Physiological process behind calcium imaging:
  - **External Stimulus:** Triggers neuronal response
  - **Neuronal Activation**: Calcium floods the cell, causing a temporary increase in intracellular calcium concentration
  - **Return to Baseline**: When the neuron is at rest, calcium levels return to their normal state
- The fluorescent calcium traces can be used as **proxies of the activity** over time of individual neurons





- Example of (a **short** chunk of) a calcium trace for an individual neuron: it represents the level of calcium over time, the **spikes** in the observed concentration correspond to the **neuron's activation**.



- **Two-step procedure**:
  - **Deconvolution**: extraction of the spike train (i.e., the series of recorded firing times). The spike train is then regarded as the *true cleaned signal*.
  - **Analysis of the spike train**: definition of summary statistics, models, etc., to link the neuronal activity with the external conditions that induced it.

Deconvolution

- Many deconvolution methods are based on a **biophysical model** that relates the **observed fluorescence trace** to the **calcium dynamic** and the **spiking activity**.

- For $i = 1,\dots,n$ and $t = 1,\dots,T$, $y_{i,t}$ is the observed fluorescence of neuron i at time t, $c_{i,t}$ is the intracellular calcium concentration of neuron i at time t.

- A popular model* considers the observed fluorescence $y_{i,t}$ as a **noisy function of the intracellular calcium concentration** $c_{i,t}$.

$$y_{i,t} = b_i + c_{i,t} + \epsilon_{i,t}, \quad \epsilon_{i,t} \sim \mathcal{N}(0,\sigma^2)$$

- The calcium dynamic is then modeled using an **autoregressive process with jumps** at the neuron's **firing events.**

$$c_{i,t} = \gamma c_{i,t-1} + a_{i,t} \cdot s_{i,t} + \omega_{i,t}, \quad \omega_{i,t} \sim \mathcal{N}(0,\tau^2)$$

- Here, b_i denotes the baseline level, $\epsilon_{i,t}$ measurement error, γ the decay parameter.

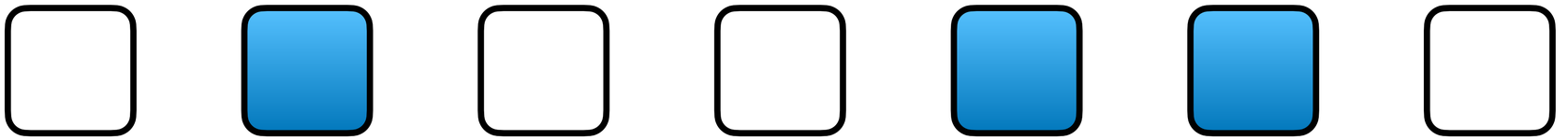
Vogelstein, J. T., Packer, A. M., Machado, T. A., Sippy, T., Babadi, B., Yuste, R. and Paninski, L. (2010) Fast nonnegative deconvolution for spike train inference from population calcium imaging. J. Neurophysiol. 104(6), 3691–3704

The specification of the latent neuronal activity

$$c_{i,t} = \gamma c_{i,t-1} + a_{i,t} \cdot s_{i,t} + \omega_{i,t}, \quad \omega_{i,t} \sim \mathcal{N}(0, \tau^2)$$

- An autoregressive component models the calcium dynamics
- An additional component is considered for modeling the calcium spikes
  - We distinguish the **binary signal indicators** and the **amplitudes of the non-zero spikes**:
    - $s_{i,t} \in \{0,1\}$ describes the absence/presence of activation for a neuron i at the time t
    - $a_{i,t} \in \mathbb{R}_+$ describes its magnitude

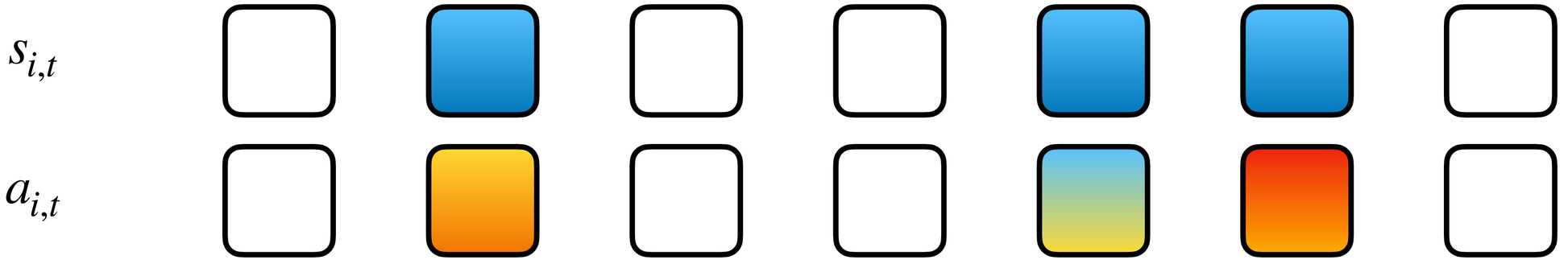
$s_{i,t}$



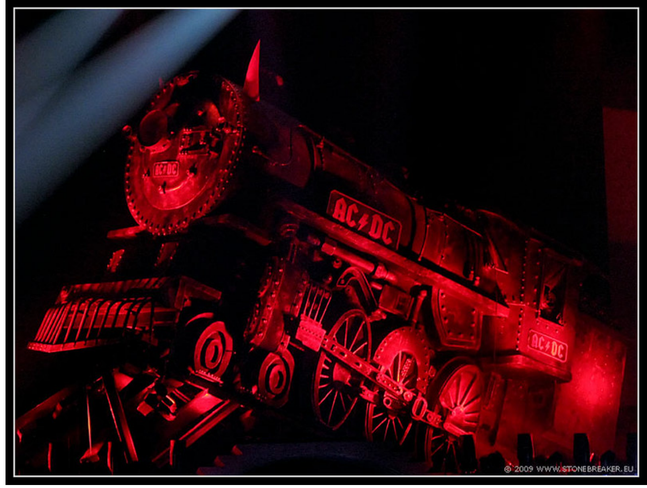
The specification of the latent neuronal activity

$$c_{i,t} = \gamma c_{i,t-1} + a_{i,t} \cdot s_{i,t} + \omega_{i,t}, \quad \omega_{i,t} \sim \mathcal{N}(0, \tau^2)$$

- An autoregressive component models the calcium dynamics
- An additional component is considered for modeling the calcium spikes
  - We distinguish the **binary signal indicators** and the **amplitudes of the non-zero spikes**:
    - $s_{i,t} \in \{0,1\}$ describes the absence/presence of activation for a neuron i at the time t
    - $a_{i,t} \in \mathbb{R}_+$ describes its magnitude



Spike trains

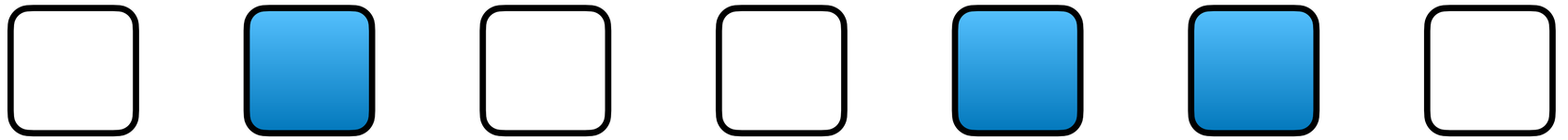


- The spike trains are the indicators of the neurons' activity over time
- They can be viewed as binary time series: for a neuron i

$$\boldsymbol{s}_i = [s_{i,1}, \ldots, s_{i,t}, \ldots, s_{i,T}] \in \{0,1\}^T$$

- Neurons with **synchronized activity** should be characterized by **similarities in the spike trains**
- These series drive the identification of the clusters of co-activating neurons.

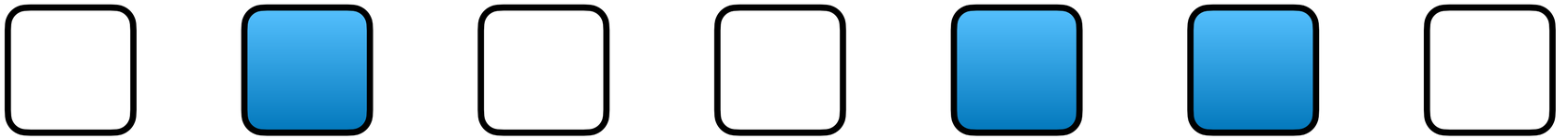
$s_{i,t}$



Spike trains

- Difficulties:
  - Often the firing events are **not strictly synchronous or overlapping in time**
  - Some **observed spikes might be due to measurement error**, or they might be random
  - Firing events are **not uniformly distributed over time** (an observed spike is produced by several consecutive activations): temporal dependence;
  - Not immediate how to cluster binary time series

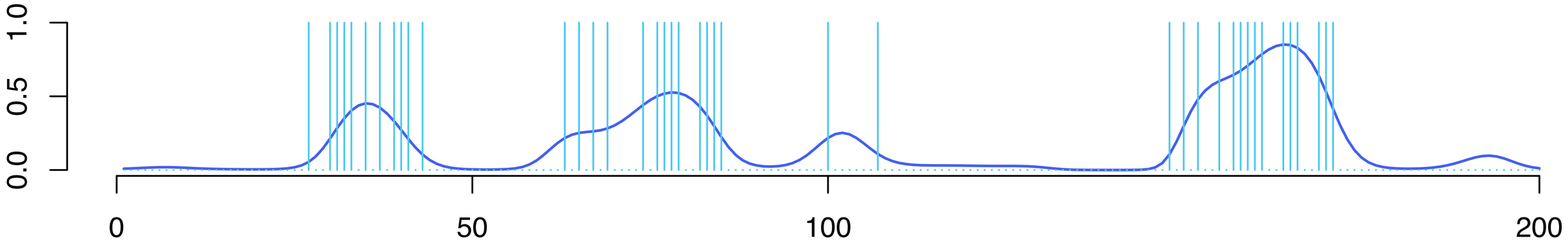
$s_{i,t}$



Data augmentation: the spike probability

- We introduce an underlying continuous process that describes the evolution of the **spike probability** over time:
  - The spike probability is modeled by a **Probit transformation of a Gaussian process** $\tilde{s}_i(t)$.
- Conditionally on $\tilde{s}_i(t)$, the spikes are generated as independent Bernoulli r.v.

$$s_{i,t} \mid \tilde{s}_i(t) \sim \text{Bernoulli}\Big(\Phi(\tilde{s}_i(t))\Big) \quad \text{for } t = 1, \ldots, T$$



Clustering spike trains

- **Idea**: similar evolutions of the spike probability will produce similar spike trains. We can cluster neurons through these latent series.

$$\tilde{s}_i \sim \sum_k \pi_k(\ell_i)\delta_{\tilde{s}_k^*}$$

- **Mixture weights**:
  - We would like to **inform the clustering** using the **anatomical location** ℓ_i
  - We use the **probit stick-breaking process** as prior distribution on the mixture probabilities:
  - More similar set of weights are induced on neighboring neurons

$$\pi_k(\ell_i) = \Phi(\alpha_k(\ell_i))\prod_{r<k}\left(1-\Phi(\alpha_r(\ell_i))\right) \quad \text{for } k = 1,2,\ldots$$

$$\left[\alpha_k(\ell_1), \alpha_k(\ell_2), \ \ldots \ , \alpha_k(\ell_n)\right]^T \sim \mathsf{N}_n\left(\mathbf{0}, \Sigma\right).$$

Clustering spike trains

- **Idea**: similar evolutions of the spike probability will produce similar spike trains. We can cluster neurons through these latent series.

$$\tilde{s}_i \sim \sum_k \pi_k(\ell_i)\delta_{\tilde{s}_k^*}$$

- **Mixture atoms:**
  - Draws from a Gaussian process (indexed by time).
  - Its covariance function Ω describes the **temporal dependence between spikes**.

$$\tilde{s}_k^* \overset{iid}{\sim} \mathrm{GP}(\mu, \Omega)$$

What about the amplitudes?

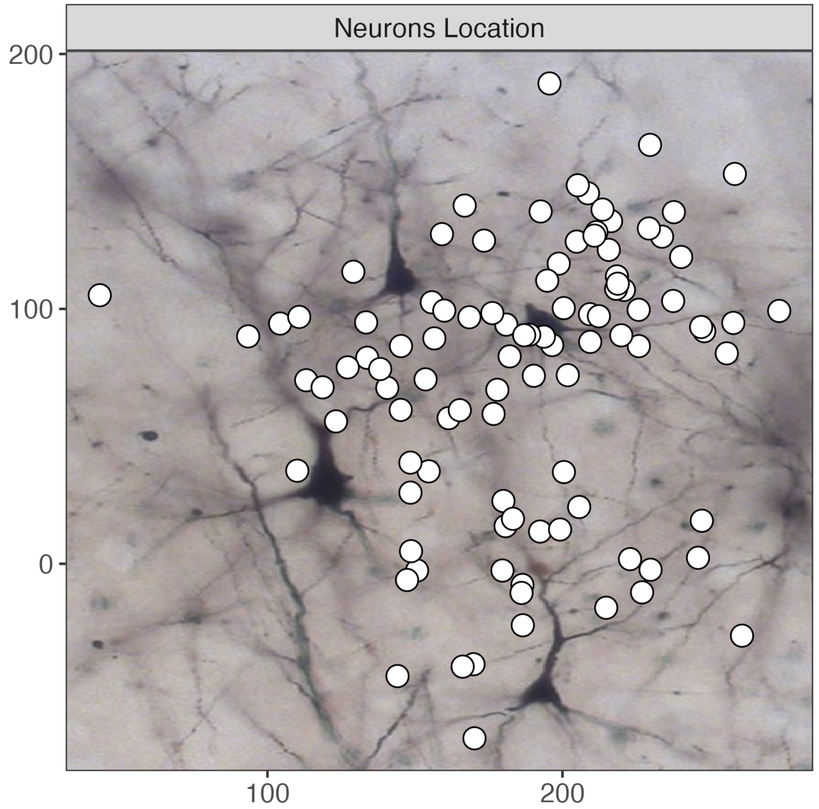
- We place an **additional DP** with a spike in zero on the distribution of the magnitudes, conditionally on the spike trains

$$a_{i,t} \mid s_{i,t}, P \sim \left(1 - s_{i,t}\right) \delta_0 + s_{i,t} P,$$
$$P \sim \mathrm{DP}\left(\alpha, P_0\right)$$

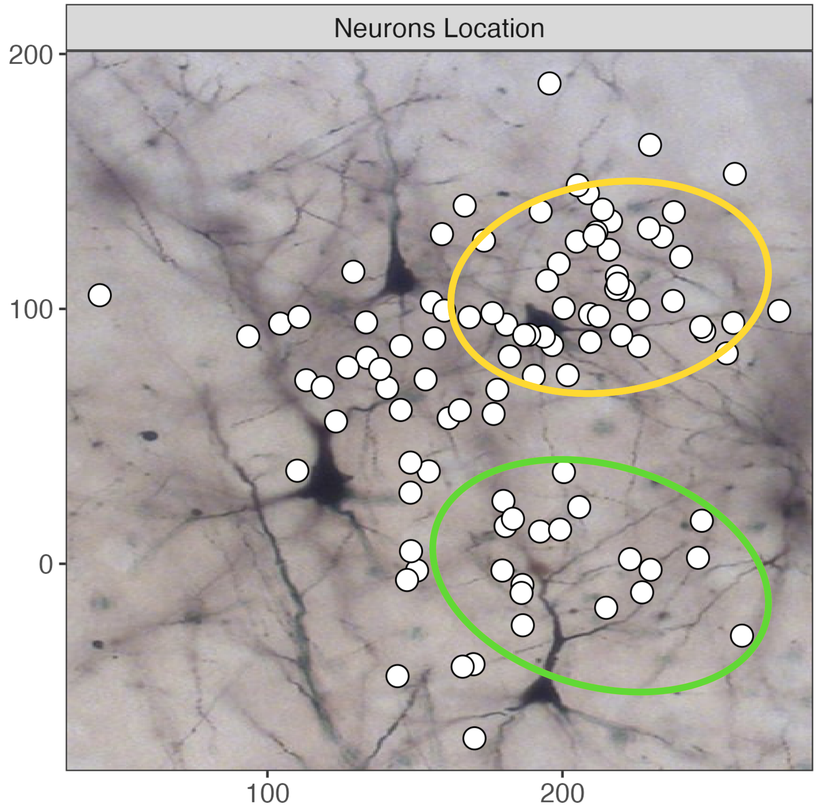
- The model is then completed with the following priors (mostly dictated by conjugacy):

$$c_{i,0} \sim \mathrm{N}\left(0, C_0\right), \quad b_i \sim \mathrm{N}\left(b_0, B_0\right),$$
$$1/\sigma^2 \sim \mathrm{Gamma}\left(h'_\sigma, h''_\sigma\right), \quad 1/\tau^2 \sim \mathrm{Gamma}\left(h'_\tau, h''_\tau\right),$$
$$\gamma \sim \mathrm{Beta}\left(h_{1\gamma}, h_{2\gamma}\right)$$

A generative look at our model



A generative look at our model



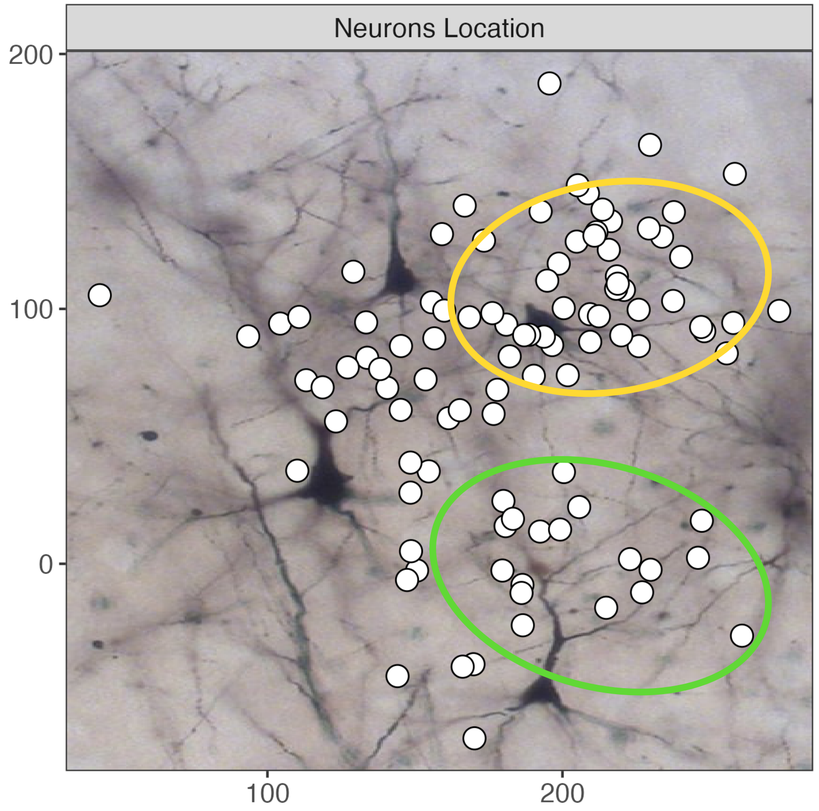
The neurons are clustered according to a PSB, taking space into account

$$\tilde{s}_i \mid \ell_i, G_{\ell_i} \sim G_{\ell_i}$$

$$G_{\ell_i} = \sum_{k=1}^{\infty} \pi_k(\ell_i) \cdot \delta_{\tilde{s}_k^*}$$

$$\pi_k(\ell_i) = \Phi(\alpha_k(\ell_i)) \prod_{r<k} \left(1 - \Phi(\alpha_r(\ell_i))\right)$$

A generative look at our model



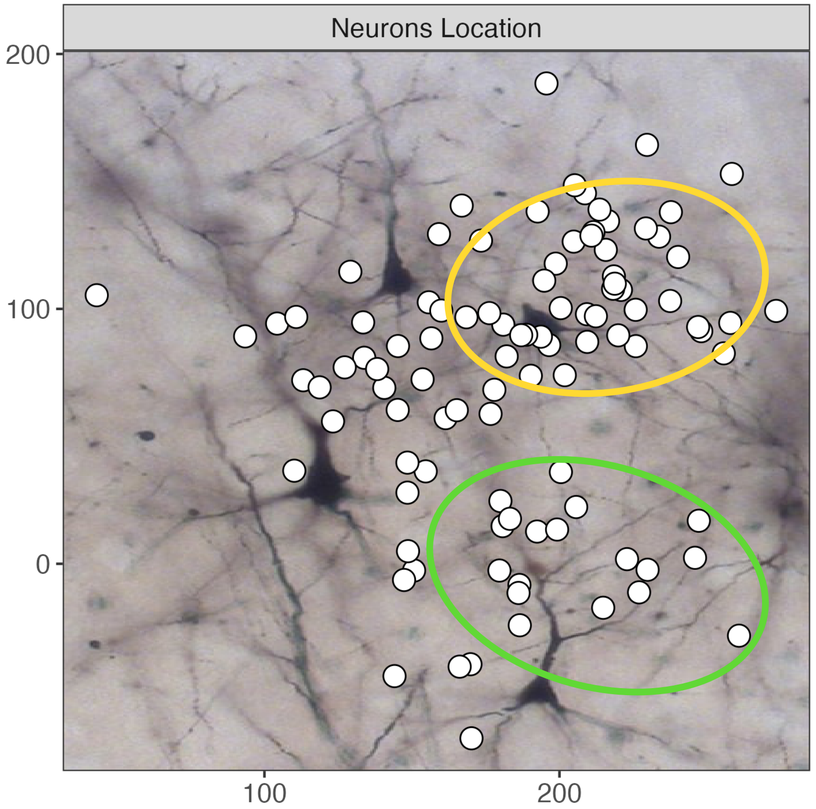
The neurons are clustered according to a PSB, taking space into account

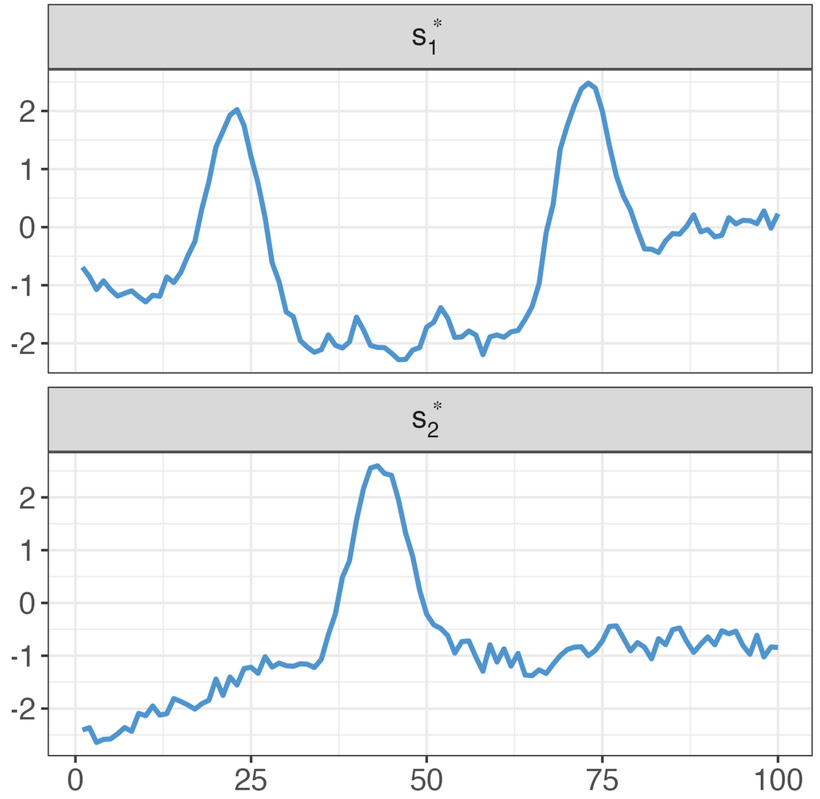
$$\tilde{s}_i \mid \ell_i, G_{\ell_i} \sim G_{\ell_i}$$

$$G_{\ell_i} = \sum_{k=1}^{\infty} \pi_k(\ell_i) \cdot \delta_{\tilde{s}_k^*}$$

The corresponding atoms are Gaussian processes…

A generative look at our model

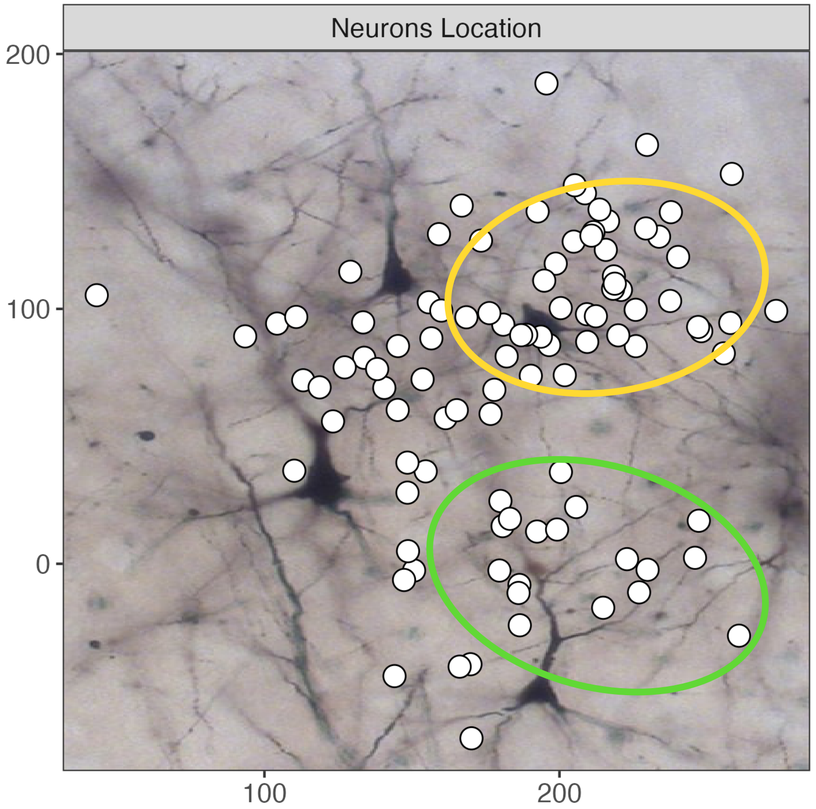


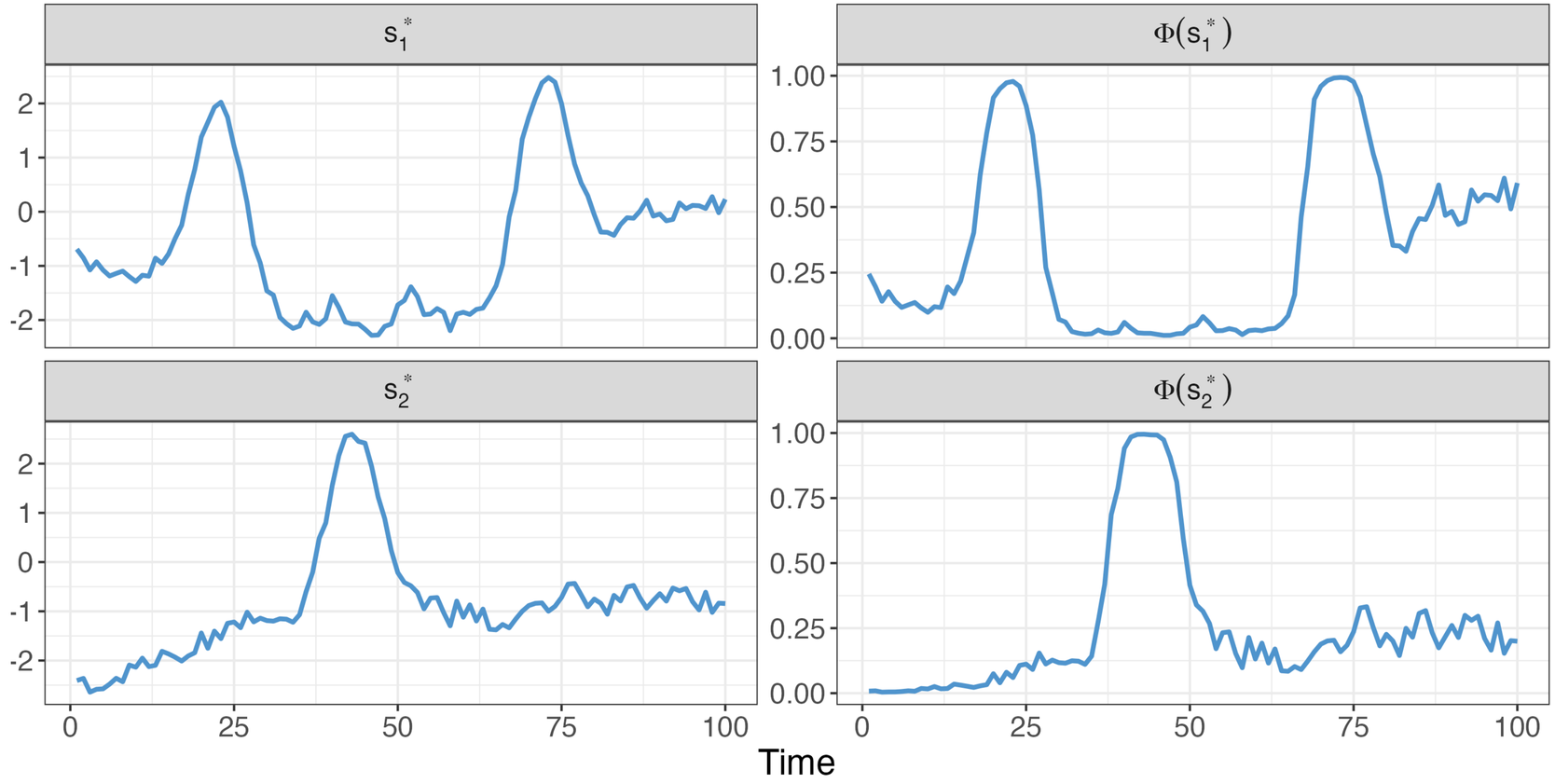


The corresponding atoms are Gaussian processes…

$$\tilde{s}_k^* \overset{iid}{\sim} \text{GP}(\mu, \Omega)$$

A generative look at our model

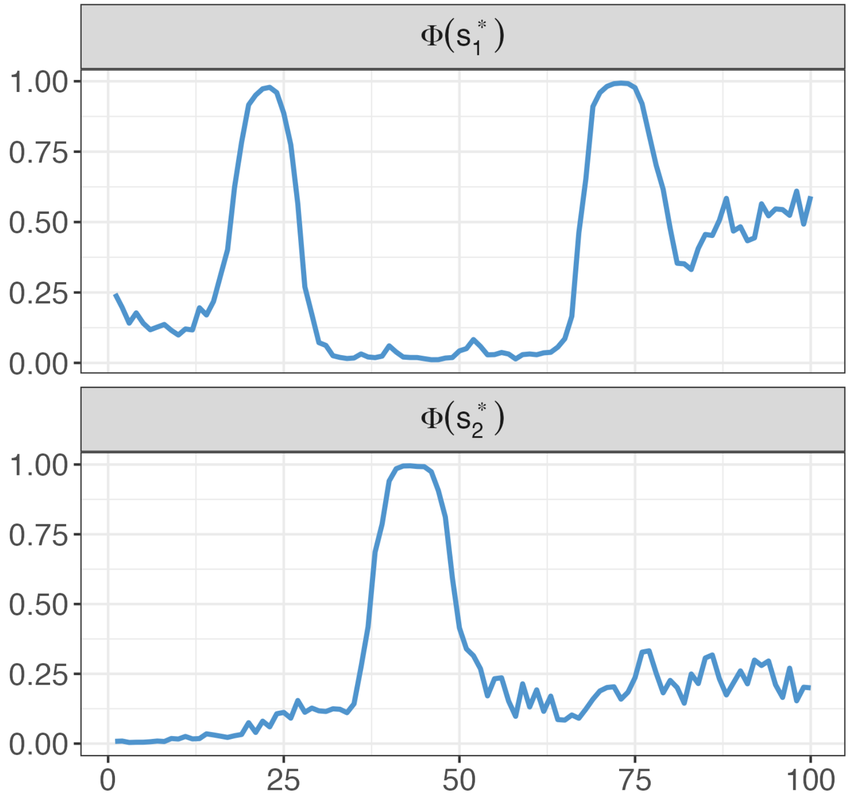




… that are transformed via a Probit transformation

$$\Phi(\tilde{s}_k^*(t))$$

A generative look at our model

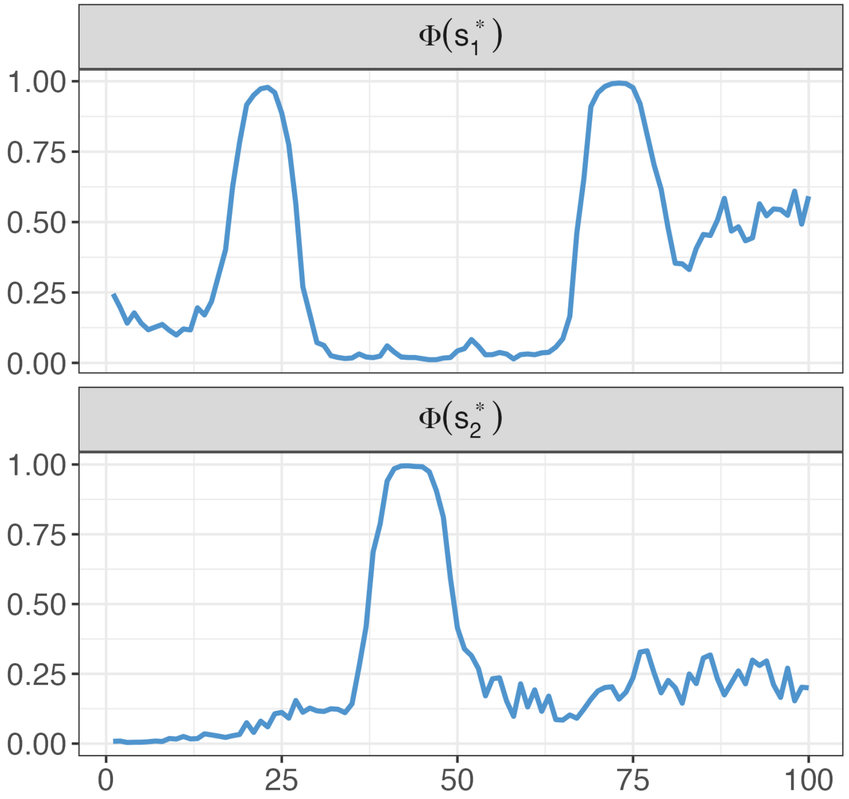


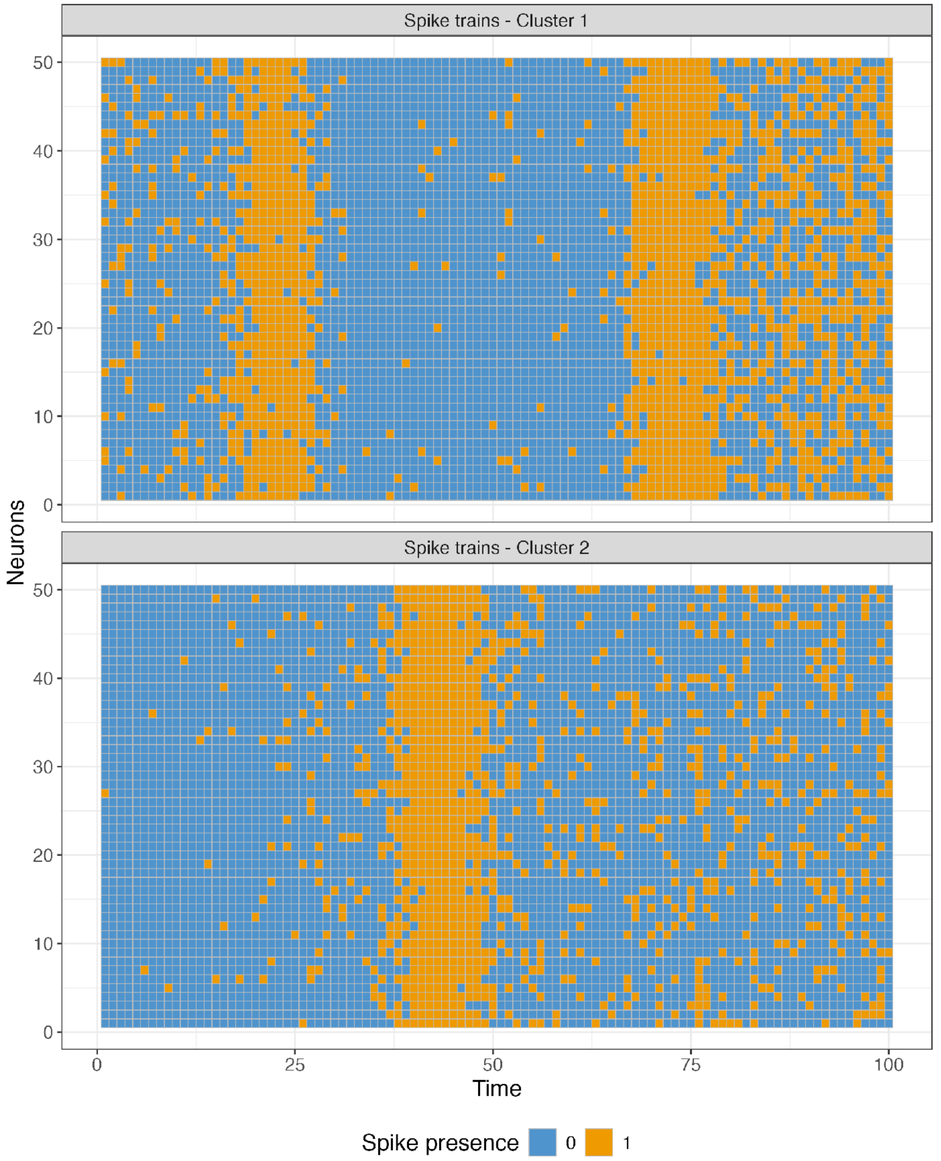
These transformed processes models the probability of spiking over time

$$s_{i,t} \mid \tilde{s}_i(t) \sim \text{Bernoulli}\Big(\Phi(\tilde{s}_i(t))\Big)$$

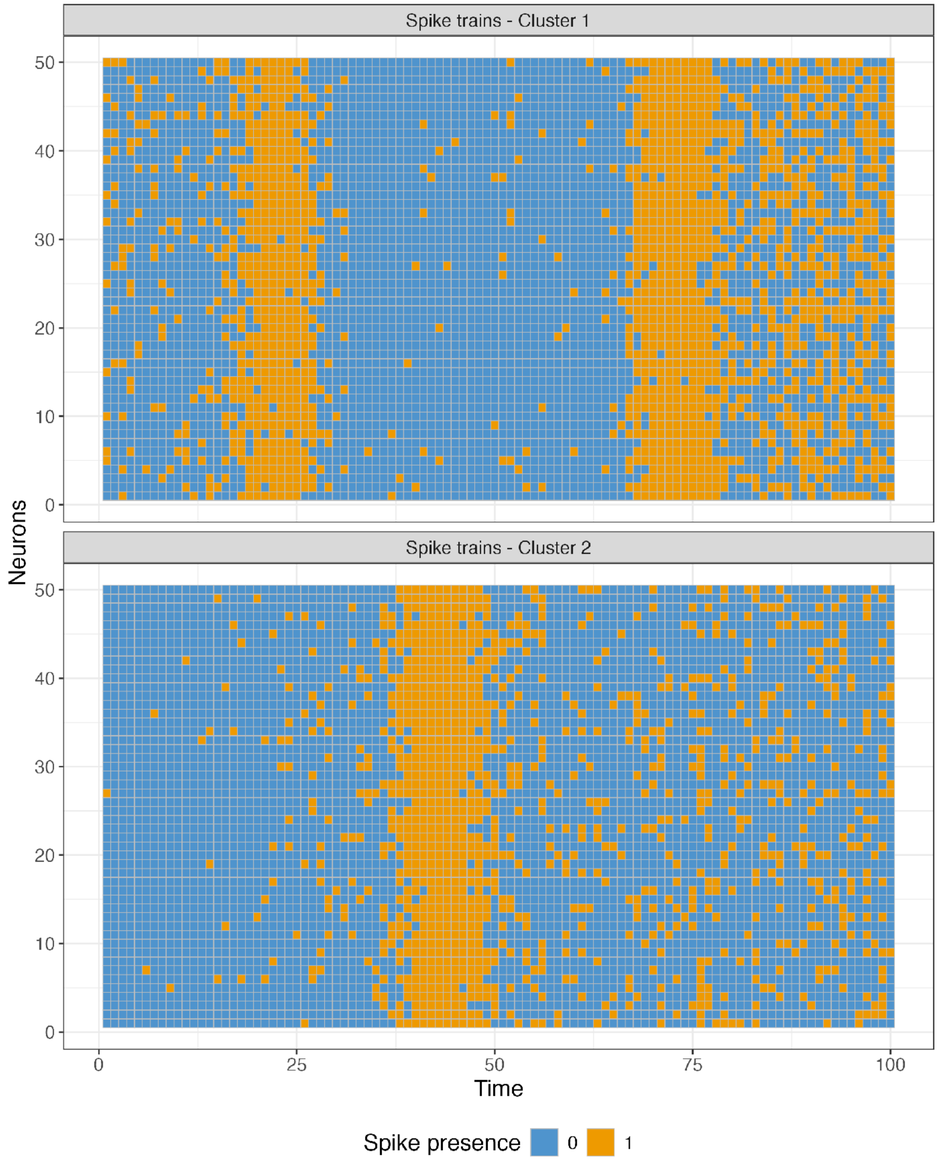
The probability is the same across all the neurons. From here, we generate the **spike trains**, one for each neuron

A generative look at our model





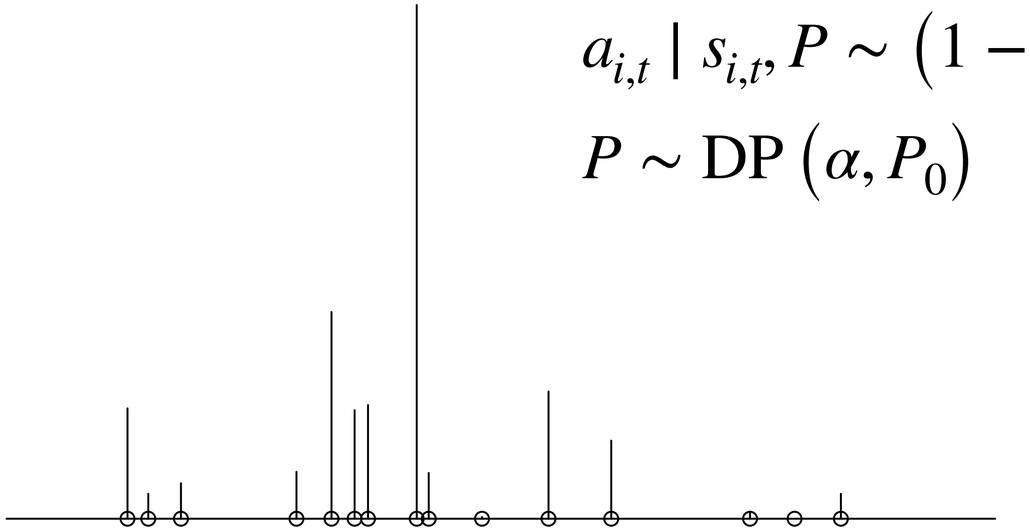
A generative look at our model



A second DP models the magnitude of each spike

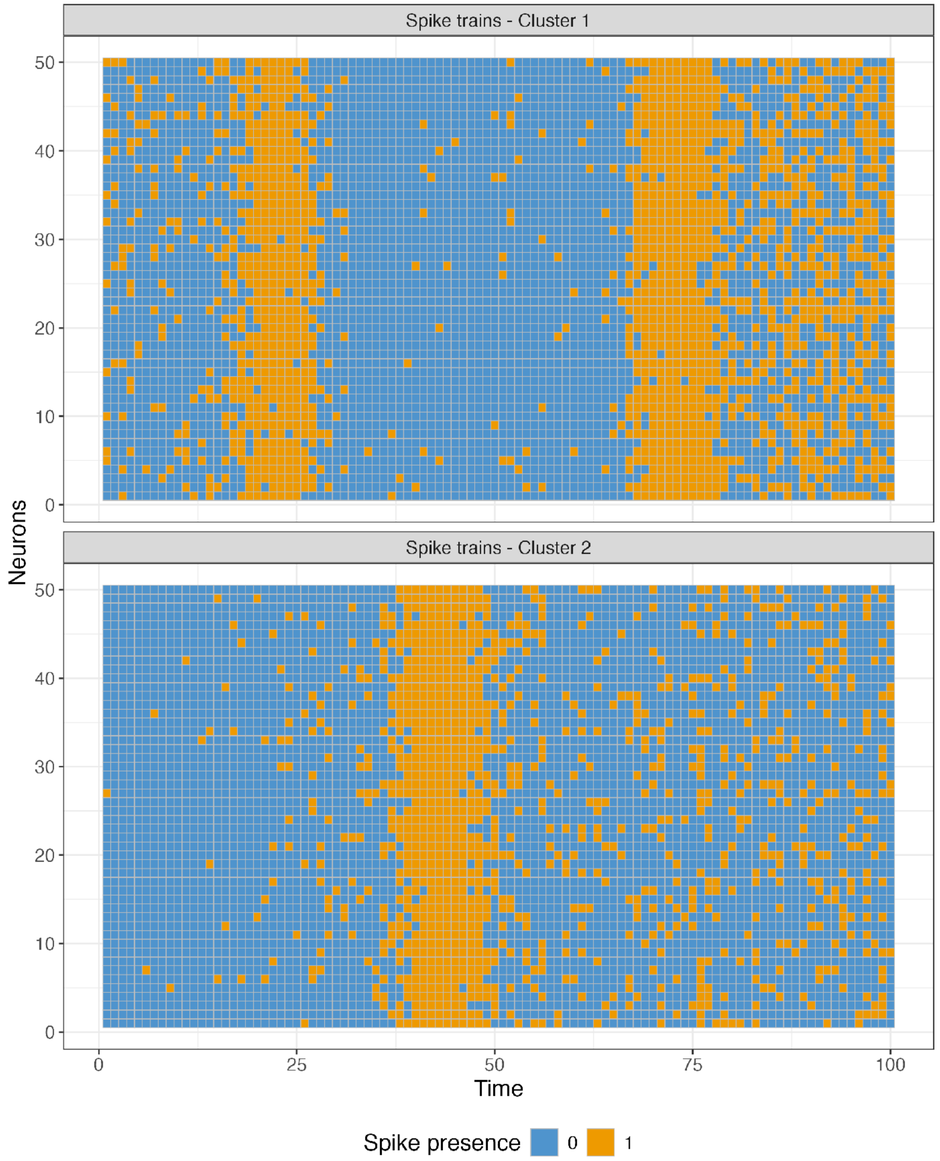
$$a_{i,t} \mid s_{i,t}, P \sim \left(1 - s_{i,t}\right) \delta_0 + s_{i,t} P,$$

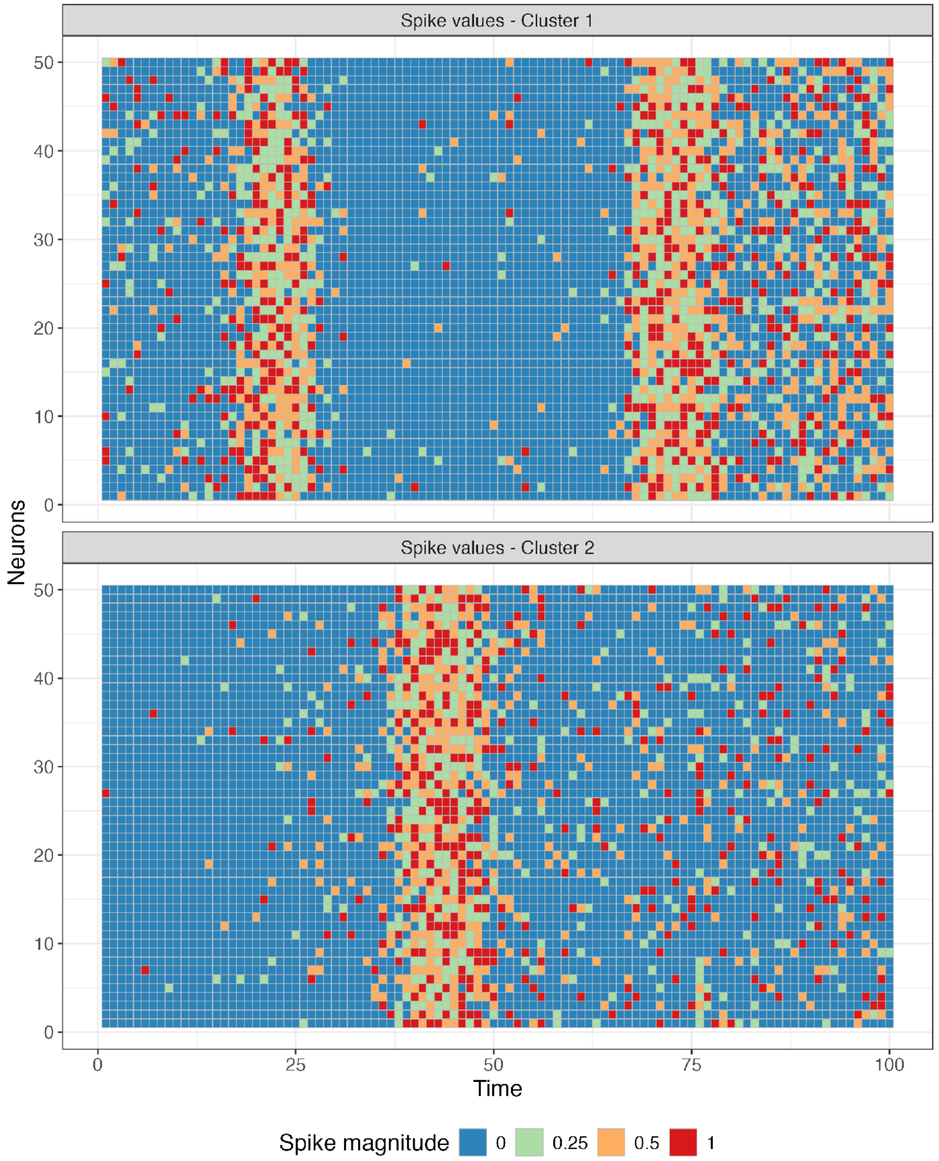
$$P \sim \mathrm{DP}\left(\alpha, P_0\right)$$



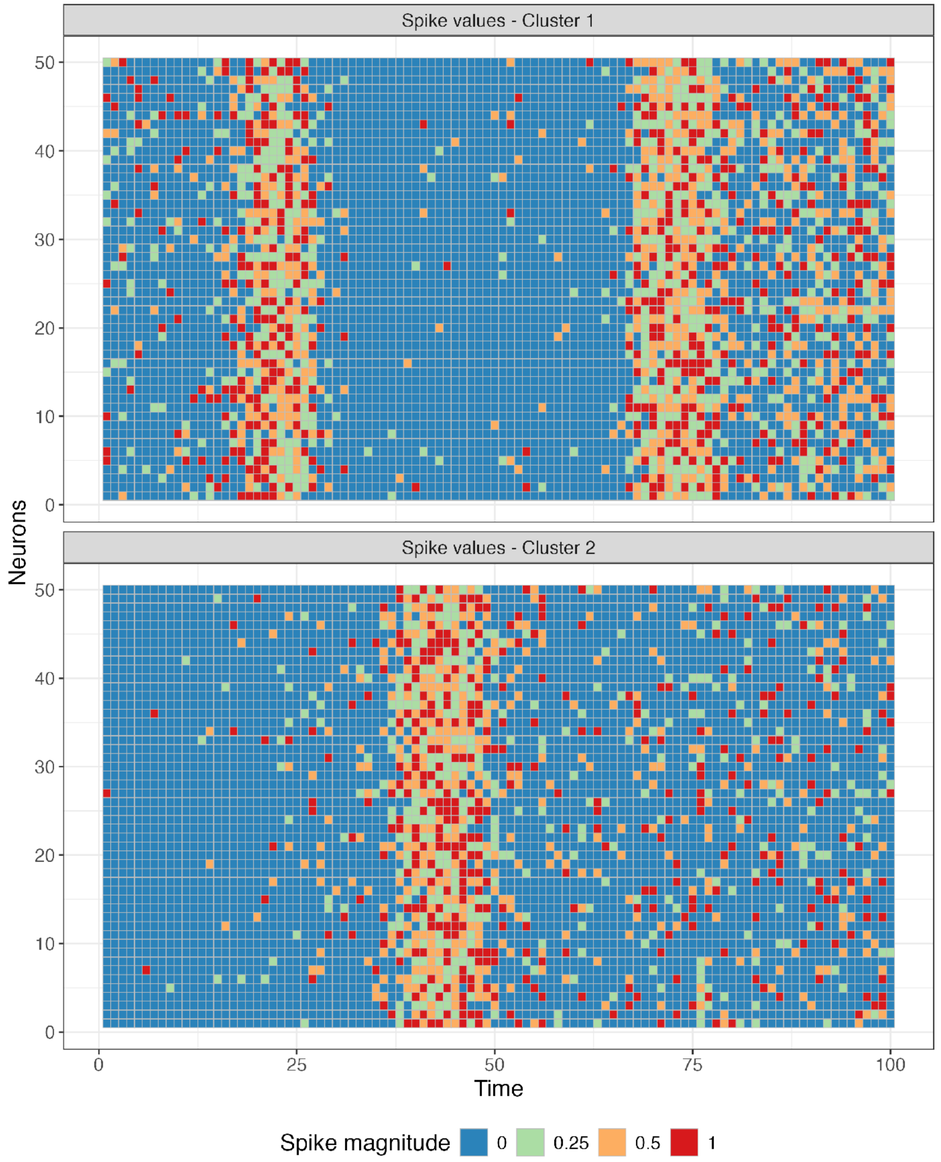
$$a^* \sim \bar{a} + Gamma(a, b)$$

A generative look at our model





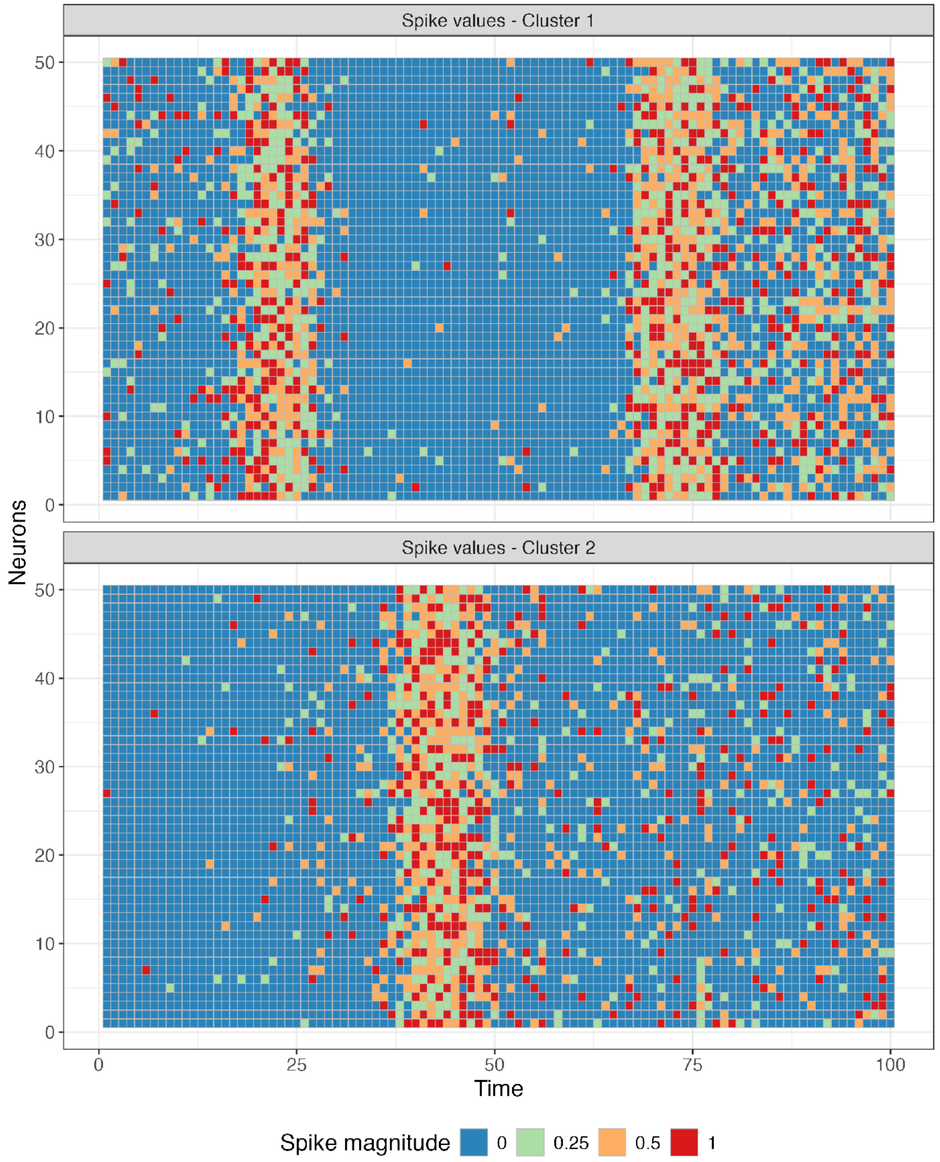
A generative look at our model

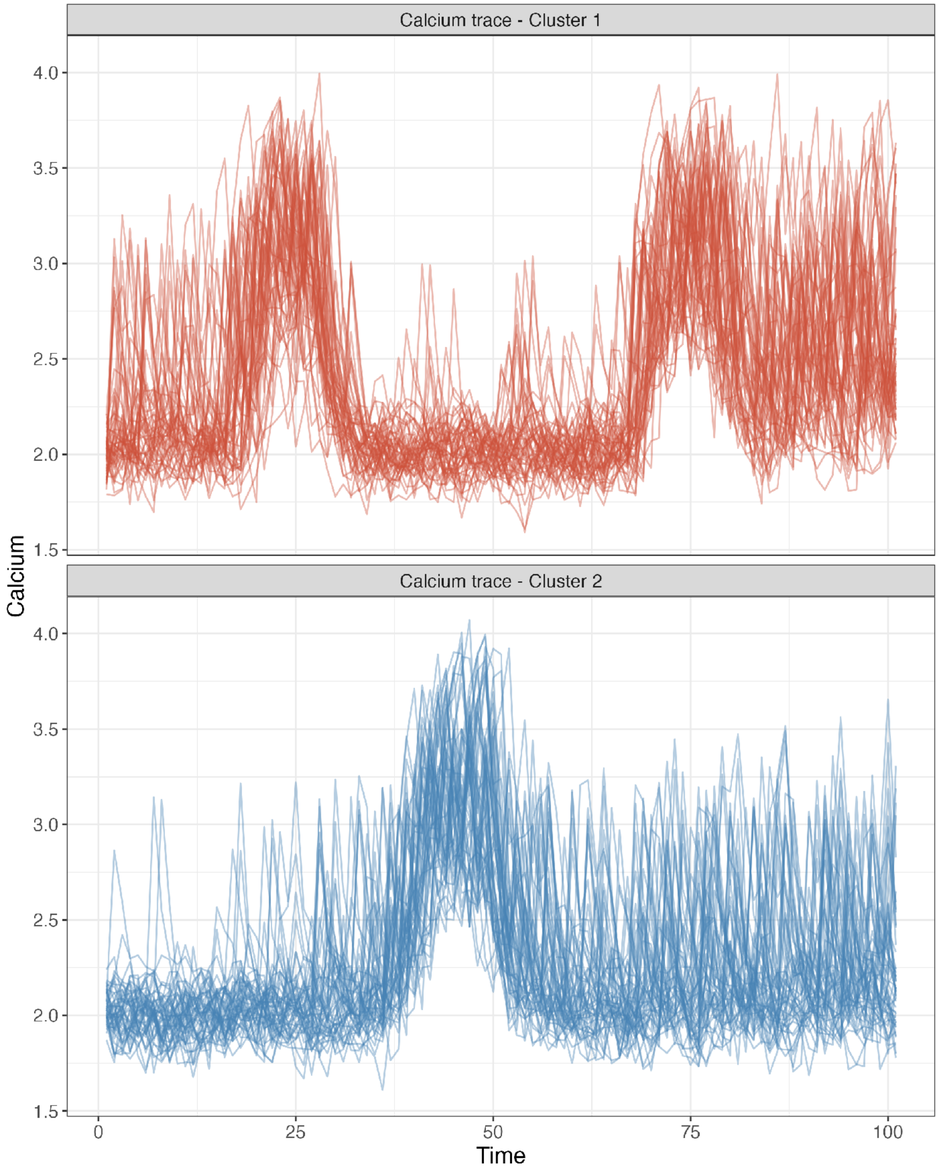


Finally, given the spike trains and the magnitudes, we generate the calcium traces series with

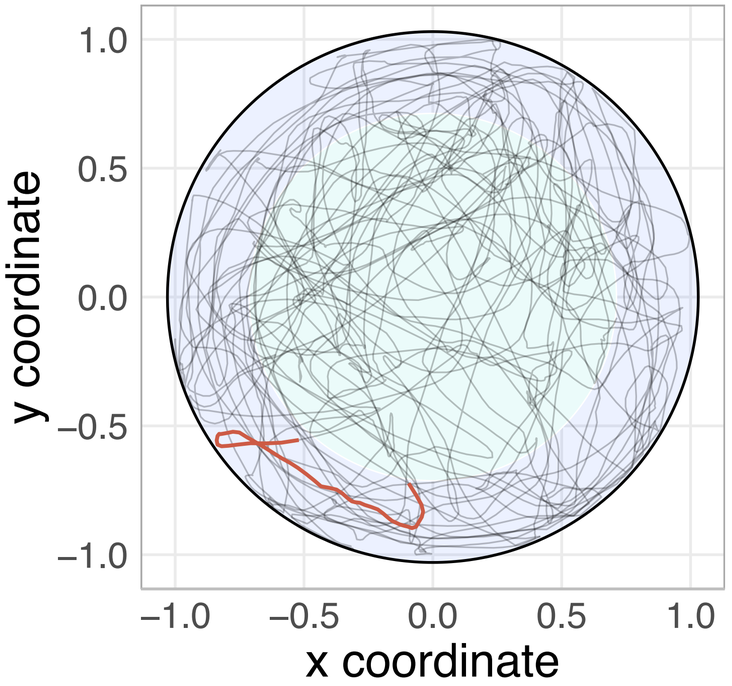
$$y_{i,t} \mid b_i, \gamma, c_{i,t-1}, s_{i,t}, a_{i,t}, \sigma^2, \tau^2 \sim \mathrm{N}(b_i + \gamma c_{i,t-1} + s_{i,t} \cdot a_{i,t},\ \sigma^2 + \tau^2)$$

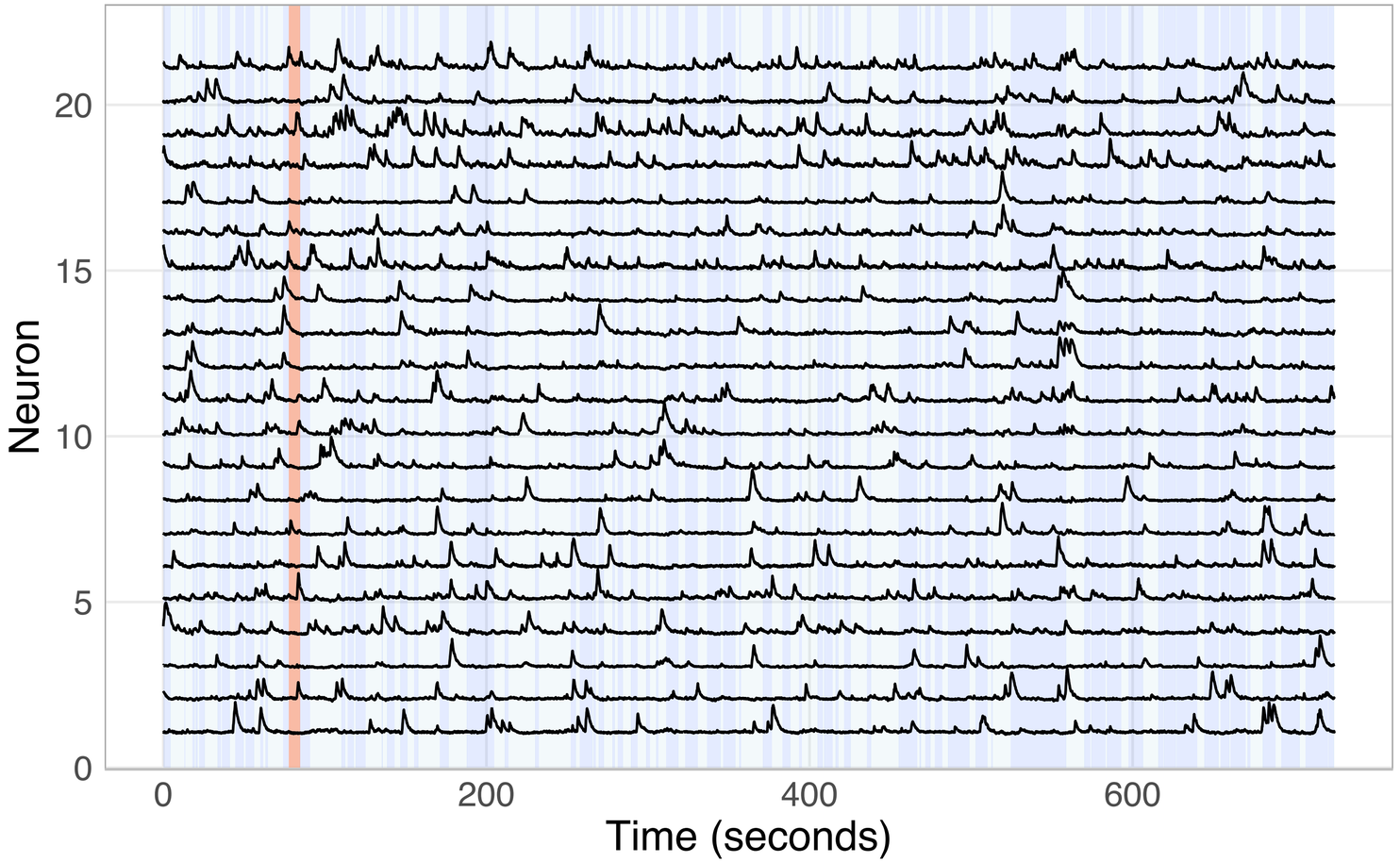
A generative look at our model





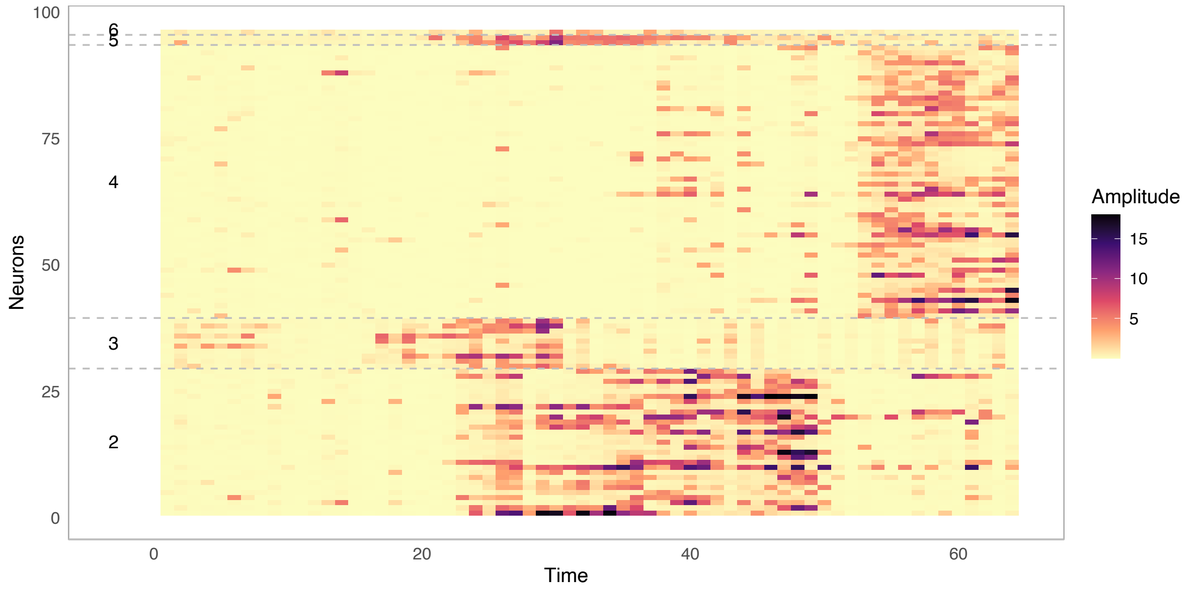
On real data



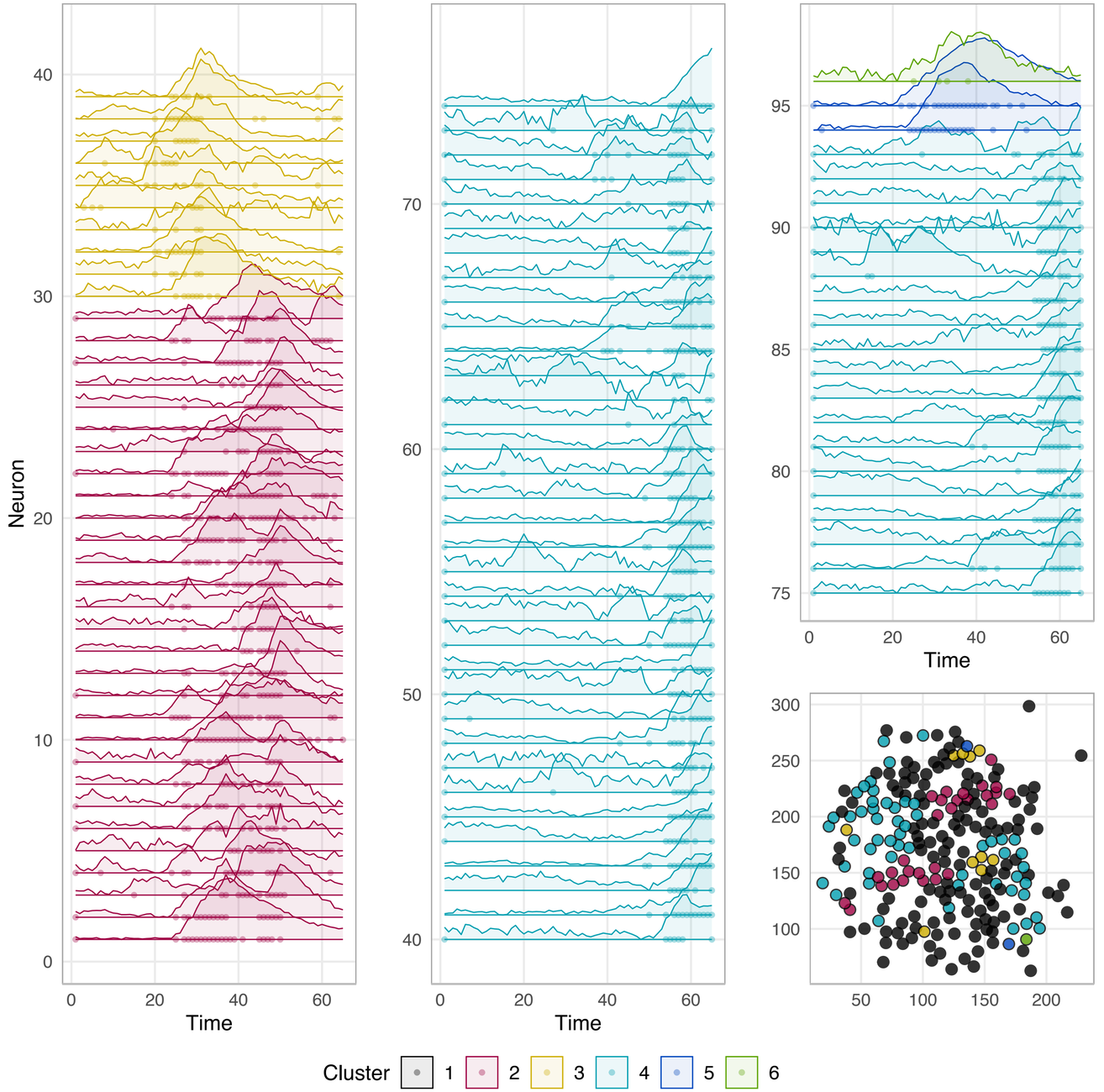


- We consider a short subset of the complete time series of the experiment
- The mouse is left to explore the circular arena
- Data: 229 neurons, more than 5000 time points that we divide into time windows according to the position of the mouse: center vs. outer ring

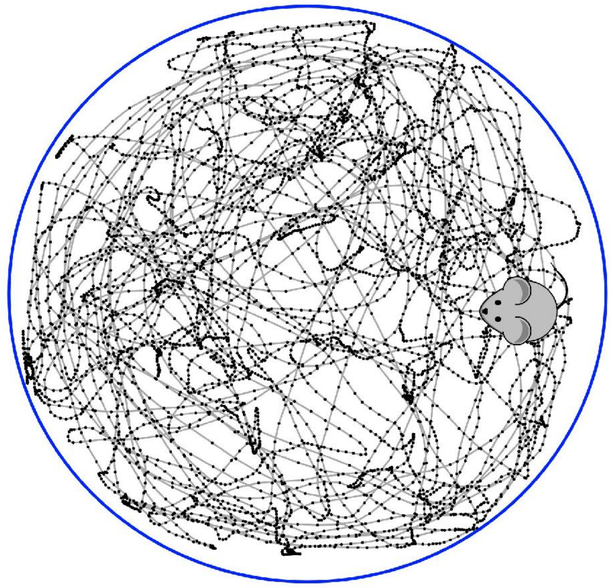
Results on a representative time window



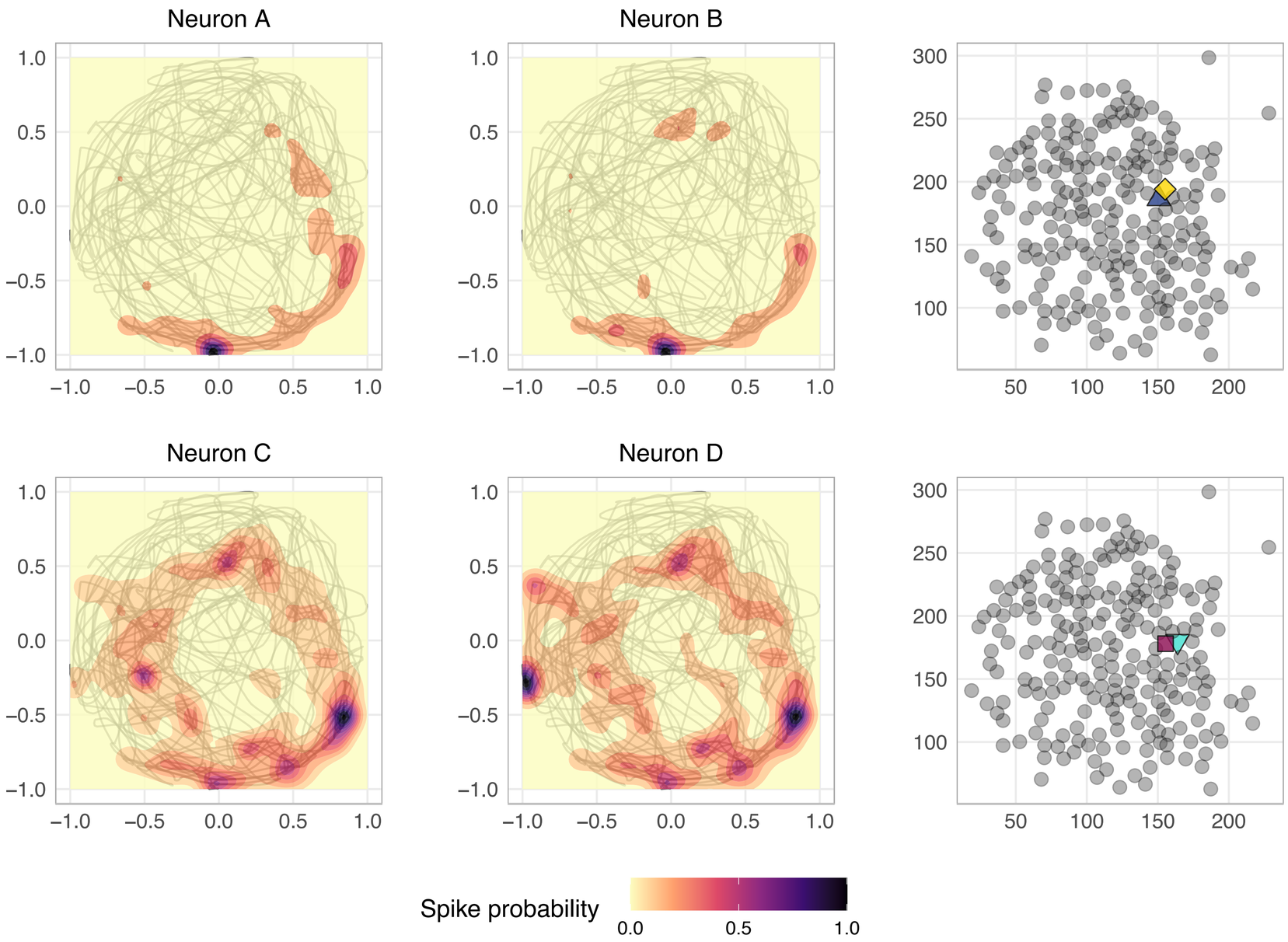
- We group the neurons in 6 clusters
- Top: spike trains + amplitudes
- Right: times series and neurons' locations colored by cluster (cluster 1 with 122 inactive neurons not reported)



Neuronal response to mouse position



- Postprocessing: analysis among time windows
- We select neurons frequently clustered together and try to understand their **spike patterns associated with the mouse position**



Discussion - Pose and Poseidon

- We introduced a model for biclustering matrix data, taking into account spatial information
- We developed a CAVI algorithm to obtain fast and accurate partitions + **easy to draw inference on nested partitions**
- We have extended the model by **integrating multiple datasets**
- **Open points**
  - CAVI is sensitive to initialization. How to pick the best starting configuration?
  - When to stop? Estimating β and $\boldsymbol{\pi}$ introduces approximation in the ELBO
  - Extend the framework to different patients, inducing an extra layer of clustering
    - Need for even model scalable computational solutions (stochastic VI?)

Discussion - SCDC

- We introduced a BNP model for flexibly modeling calcium imaging traces, performing simultaneous clustering of neurons and deconvolution.
  - Provide a one-step modeling approach to better quantify uncertainty

- **Open points**
  - C++ implementation of the MCMC algorithm. The time series are long, and the model could be slow. We focused on different subsets - suboptimal.
  - How can the Gaussian Processes be better approximated?
  - VI strategies for PSB?
  - Other modeling approaches: the firing of neurons suggests the use of Hawkes processes

THANK YOU FOR YOUR ATTENTION!

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Francesco Denti
francesco.denti@unipd.it
University of Padua

Next step: Poseidon

$$p(\boldsymbol{Y} \mid \boldsymbol{R}, \boldsymbol{C}, \boldsymbol{\theta}) = \prod_{t=1}^{T} \prod_{i=1}^{N^{(t)}} \prod_{j=1}^{J} \prod_{l=1}^{L} \prod_{k=1}^{K} \left[\phi \left(y_{i,j}^{(t)} \mid \theta_{l}^{(t)} \right)^{\mathbb{1}_{\{r_{i,k}^{(t)}=l\}} \cdot \mathbb{1}_{\{C_j=k\}}} \right]$$

Dataset-specific Row Partitions

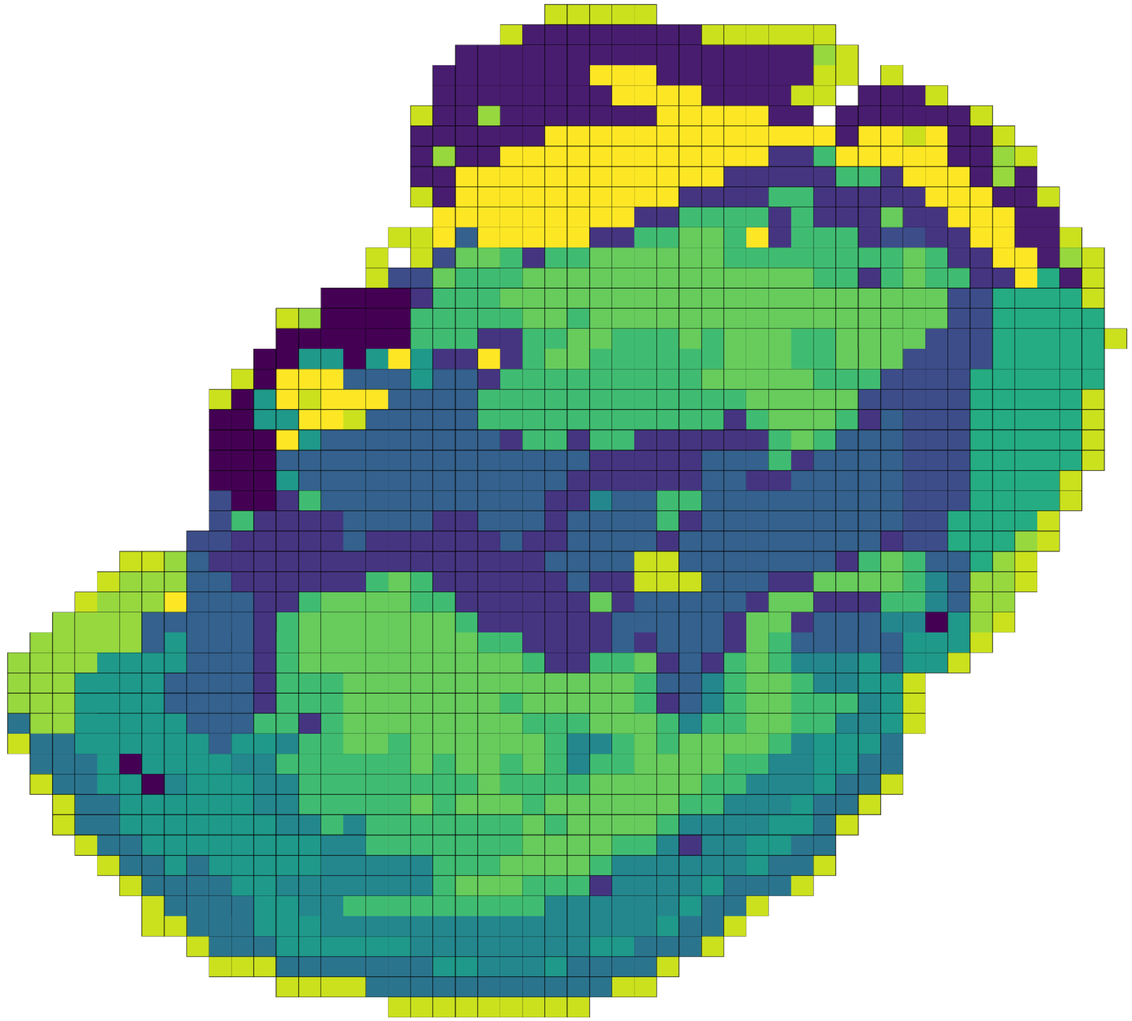
$$p(\boldsymbol{R} \mid \boldsymbol{C}, \boldsymbol{\theta}) = \prod_{t=1}^{T} \prod_{i=1}^{N^{(t)}} \prod_{l=1}^{L} \prod_{k=1}^{K} \left[\omega_{l,k}^{(t)} \right]^{\mathbb{1}_{\{r_{i,k}^{(t)}=l\}}}$$

Shared Column Partition

$$p(\boldsymbol{C} \mid \boldsymbol{\pi}) = \frac{1}{\mathcal{K}_{\boldsymbol{C}}(\boldsymbol{\pi}, \beta)} \exp \left[\sum_{j=1}^{J} \log(\pi_{C_j}) + \beta \sum_{q \in \mathcal{N}_j} \mathbb{1}_{\{C_j=C_q\}} \right]$$

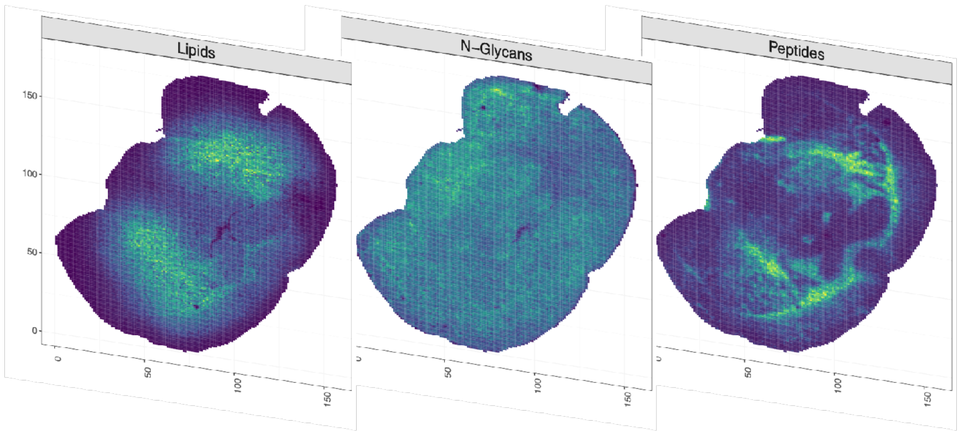
Some preliminary results

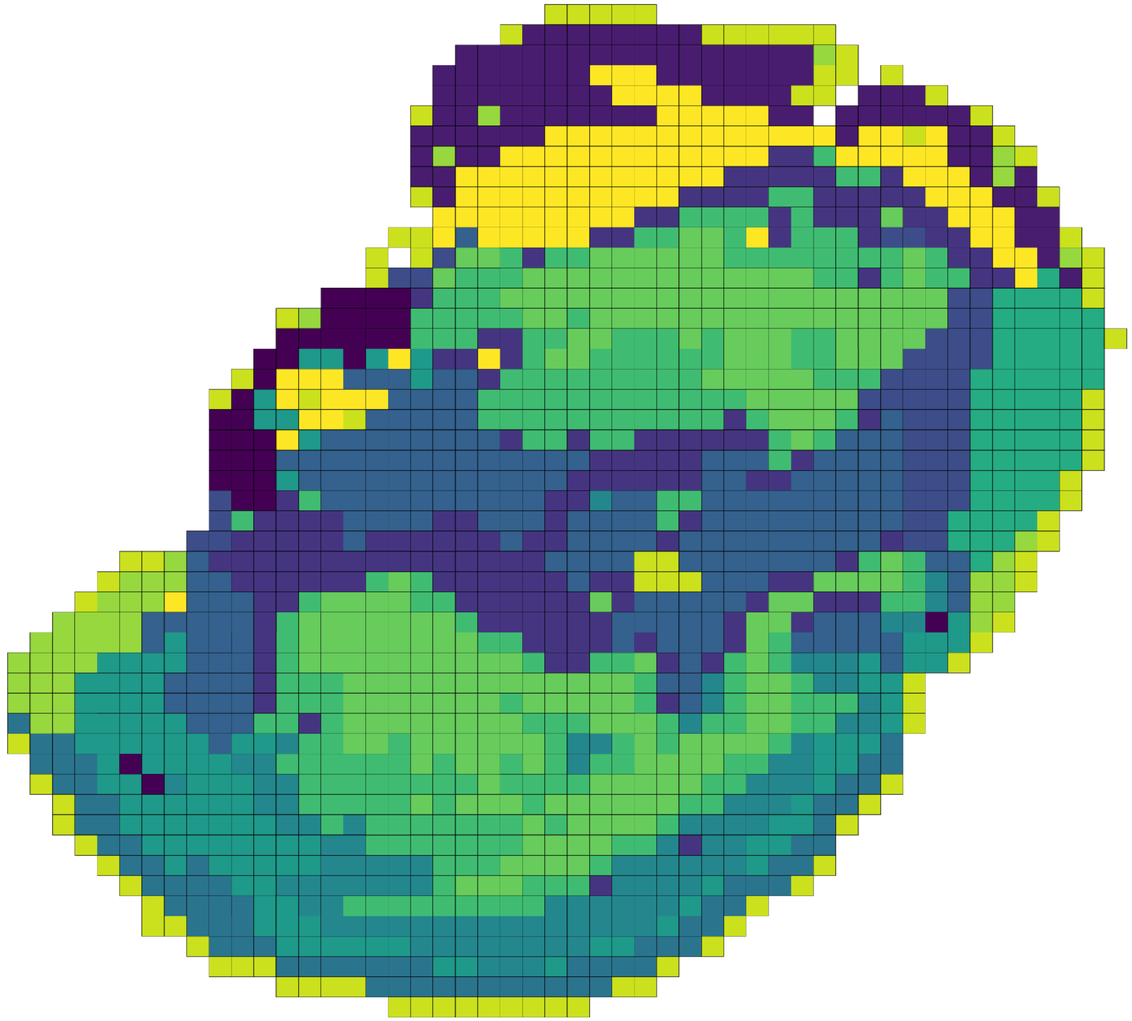
Shared column clustering



Some preliminary results

Shared column clustering





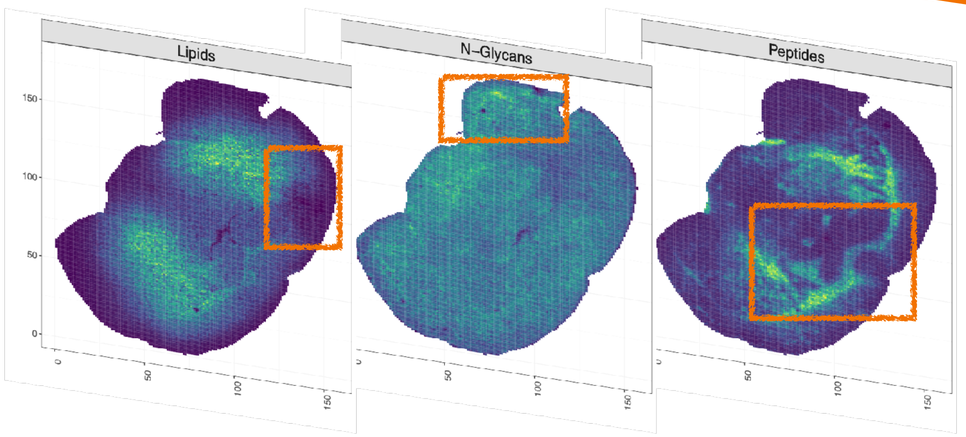
Some preliminary results

Shared column clustering

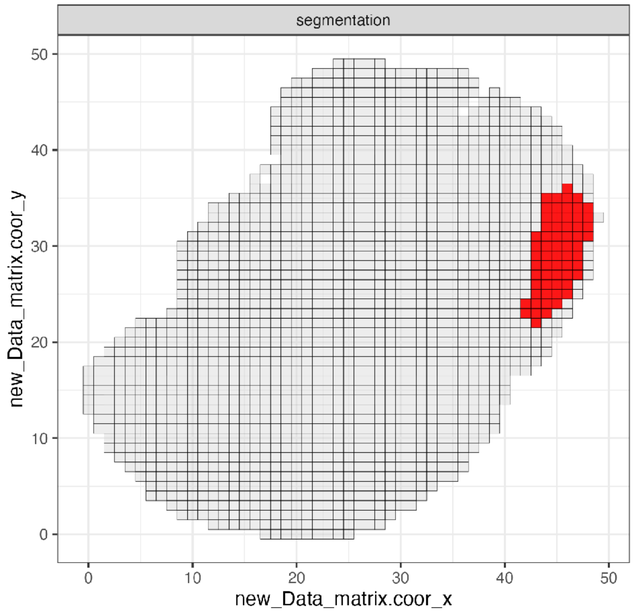
From glycans

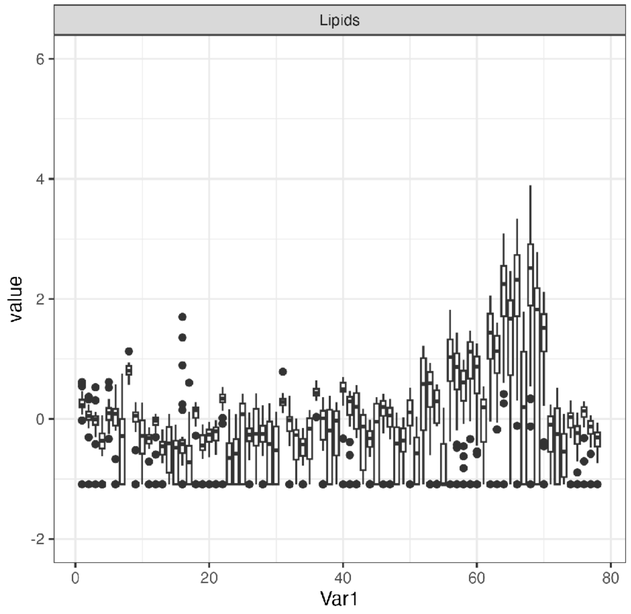
From lipids

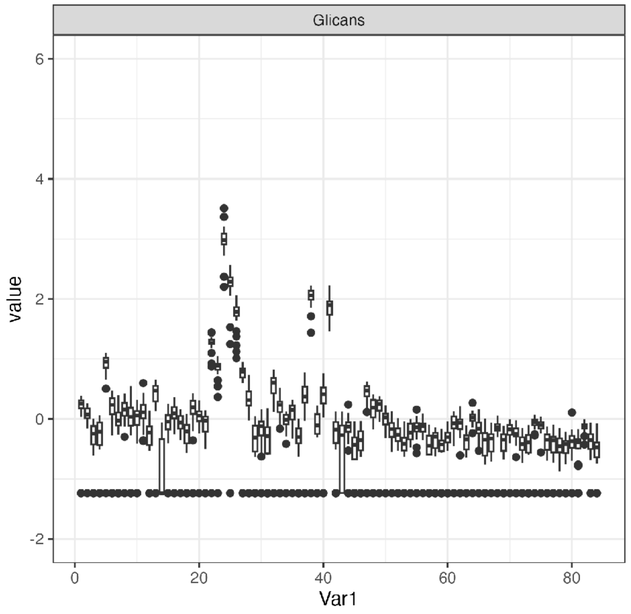
From peptides

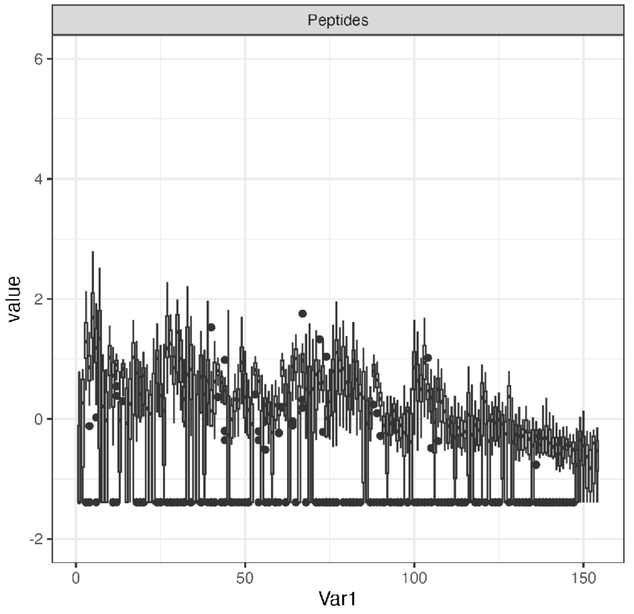


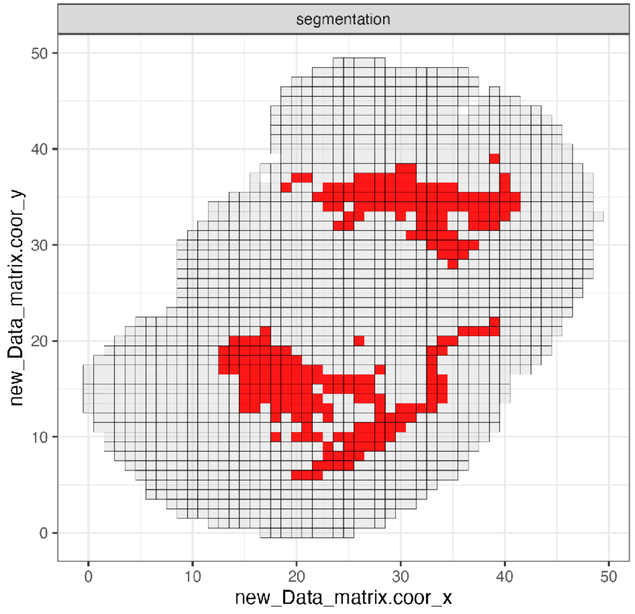
Some preliminary results

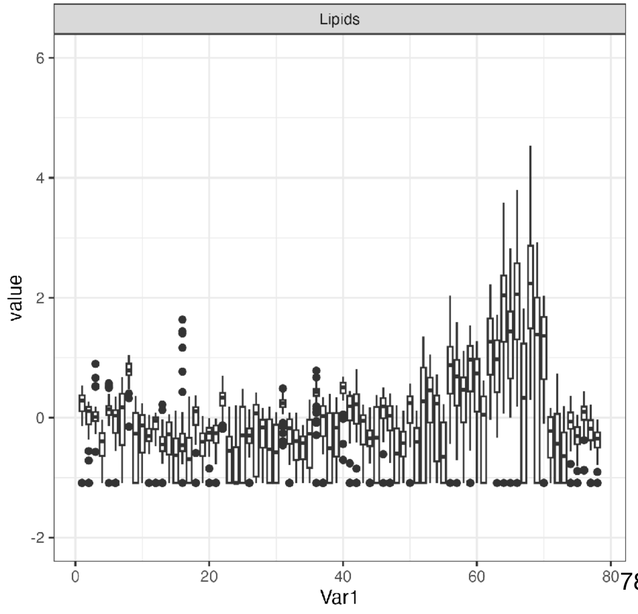


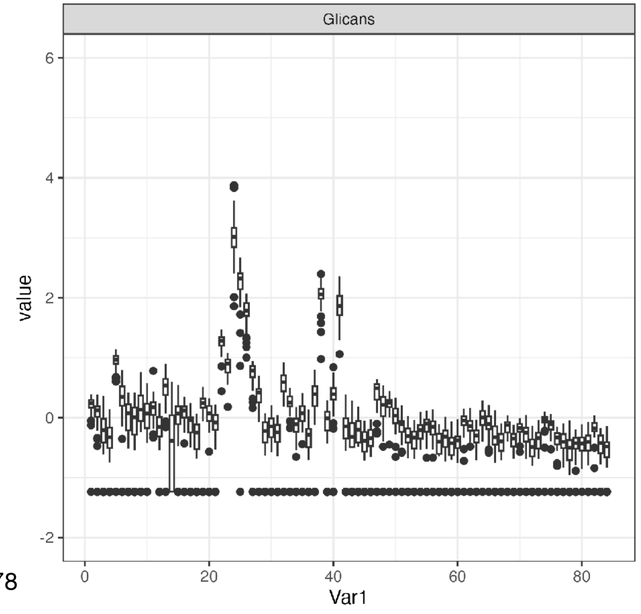


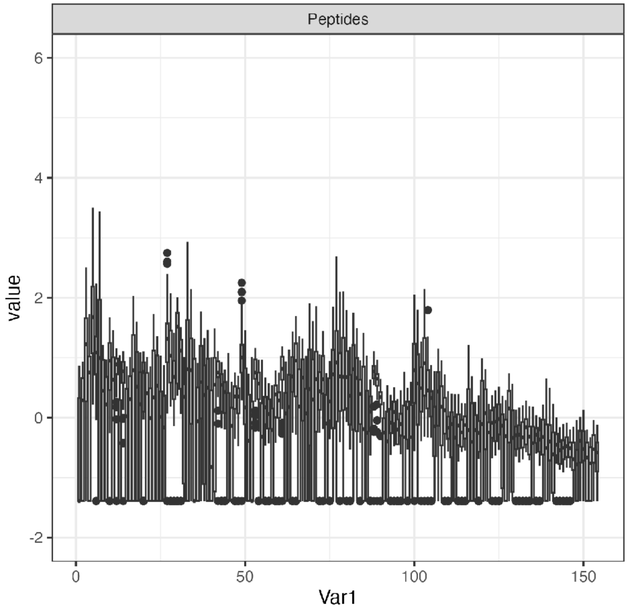












An Orwellian problem

An Orwellian problem

Bayesian Analysis (2024) **TBA**, Number TBA, pp. 1–34

A Finite-Infinite Shared Atoms Nested Model for the Bayesian Analysis of Large Grouped Data Sets

Laura D'Angelo* and Francesco Denti†

Contents lists available at ScienceDirect

Econometrics and Statistics

journal homepage: www.sciencedirect.com/journal/econometrics-and-statistics

The generalized nested common atoms model

Francesco Denti [a,*], Laura D'Angelo [b]

[a] University of Padua, via Cesare Battisti 241, Padua, 35121, Italy
[b] University of Milan-Bicocca, Piazza dell'Ateneo Nuovo 1, Milan, 20126, Italy

Common Atoms Model for nested data

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the scores of the songs of the j-th artist

$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\cdot) = \int_{\Theta} f(\cdot \mid \theta) G_j(d\theta)$$

Mixture model for the energy distribution of **each artist**

Common Atoms Model for nested data

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the scores of the songs of the j-th artist

$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\cdot) = \int_{\Theta} f(\cdot \mid \theta) G_j(d\theta)$$

$$G_1, \ldots, G_J \mid Q \sim Q \qquad Q = \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}$$

The group-specific mixing measures G_j are sampled from a common discrete distribution Q, where $\{\pi_k\}_{k \geq 1} \sim GEM(\alpha)$

Common Atoms Model for nested data

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the scores of the songs of the j-th artist

$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\cdot) = \int_\Theta f(\cdot \mid \theta) G_j(d\theta)$$

$$G_1, \ldots, G_J \mid Q \sim Q \qquad Q = \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}$$

The group-specific mixing measures G_j are sampled from a common discrete distribution Q, where $\{\pi_k\}_{k \geq 1} \sim GEM(\alpha)$

We will refer to G_k^*'s as the **distributional atoms**

Q is discrete, so ties will occur among the mixing distributions: **distributional clustering**

Common Atoms Model for nested data

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the scores of the songs of the j-th artist

$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\cdot) = \int_\Theta f(\cdot \mid \theta) G_j(d\theta)$$

$$G_1, \ldots, G_J \mid Q \sim Q \qquad Q = \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}$$

$$G_k^* = \sum_{l=1}^{\infty} \omega_{l,k} \delta_{\theta_l^*} \qquad \theta_1^*, \theta_2^*, \ldots \sim G_0$$

The distributional atoms are themselves discrete random measures, with

$$\{\omega_{l,k}\}_{l \geq 1} \sim GEM(\beta) \quad \forall k \geq 1$$

Common Atoms Model for nested data

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the scores of the songs of the j-th artist

$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\cdot) = \int_\Theta f(\cdot \mid \theta) G_j(d\theta)$$

$$G_1, \ldots, G_J \mid Q \sim Q \qquad Q = \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}$$

$$G_k^* = \sum_{l=1}^{\infty} \omega_{l,k} \delta_{\theta_l^*} \qquad \theta_1^*, \theta_2^*, \ldots \sim G_0$$

The distributional atoms are themselves discrete random measures, with

$$\{\omega_{l,k}\}_{l \geq 1} \sim GEM(\beta) \quad \forall k \geq 1$$

Crucial is the presence of **common observational atoms** sampled from a base measure G_0

G_k^* is a discrete mixing distribution, so multiple observations will be assigned to the same atom θ_l^*: **observational clustering**

Common Atoms Model for nested data

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the gene expressions in the j-th pixel

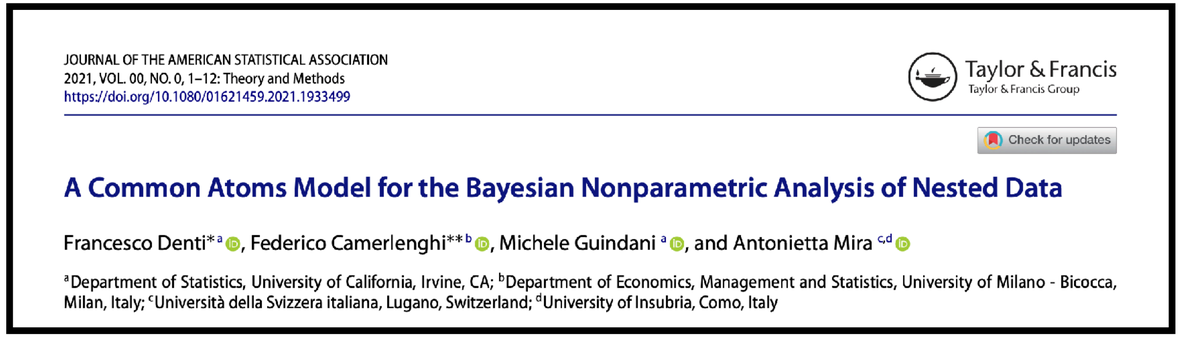
$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\cdot) = \int_\Theta f(\cdot \mid \theta) G_j(d\theta)$$

$$G_1, \ldots, G_J \mid Q \sim Q \qquad Q = \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}$$

$$G_k^* = \sum_{l=1}^{\infty} \omega_{l,k} \delta_{\theta_l^*} \qquad \theta_1^*, \theta_2^*, \ldots \sim G_0$$

$$\{\pi_k\}_{k \geq 1} \sim GEM(\alpha) \qquad \{\omega_{l,k}\}_{l \geq 1} \sim GEM(\beta) \quad \forall k \geq 1$$

The CAM is one of the many proposals devised to address the degeneracy property of the nested DP (see, for example, Beraha et al. (2021), Balocchi et al. (2023), Camerlenghi et al. (2019), D'Angelo et al. (2023), Denti et al. (2023), Rebaudo et al. (2023), Rodriguez et al. (2011),…)


JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2021, VOL. 00, NO. 0, 1–12: Theory and Methods
https://doi.org/10.1080/01621459.2021.1933499
Taylor & Francis
Taylor & Francis Group
Check for updates

A Common Atoms Model for the Bayesian Nonparametric Analysis of Nested Data

Francesco Denti*[a], Federico Camerlenghi**[b], Michele Guindani [a], and Antonietta Mira [c,d]

[a]Department of Statistics, University of California, Irvine, CA; [b]Department of Economics, Management and Statistics, University of Milano - Bicocca, Milan, Italy; [c]Università della Svizzera italiana, Lugano, Switzerland; [d]University of Insubria, Como, Italy

CAM and correlation

- To measure the **dependence across random measures** generated by the process, one can look at $Corr(G_j(A), G_{j'}(A))$ with A generic Borel set

- In CAMs, the correlation does not depend on the specific set A, so it is used as a measure of **generic dependence** among the generated $G_j's$, reflecting the **flexibility** of the process

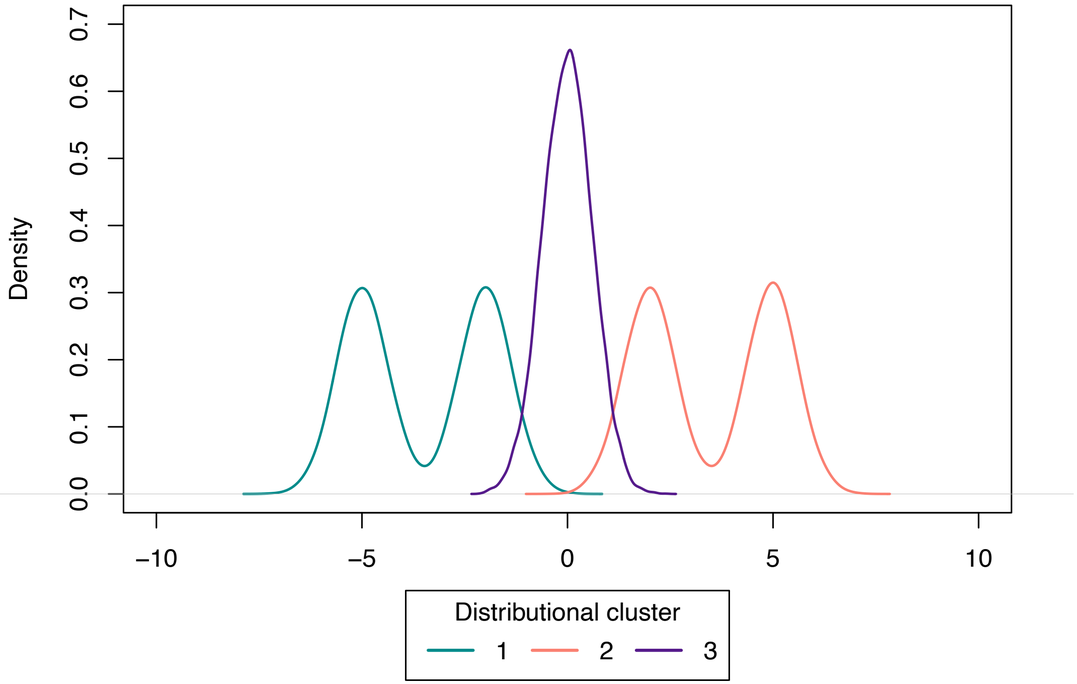
- Thus we write $\rho_{j,j'} = Corr(G_j, G_{j'})$

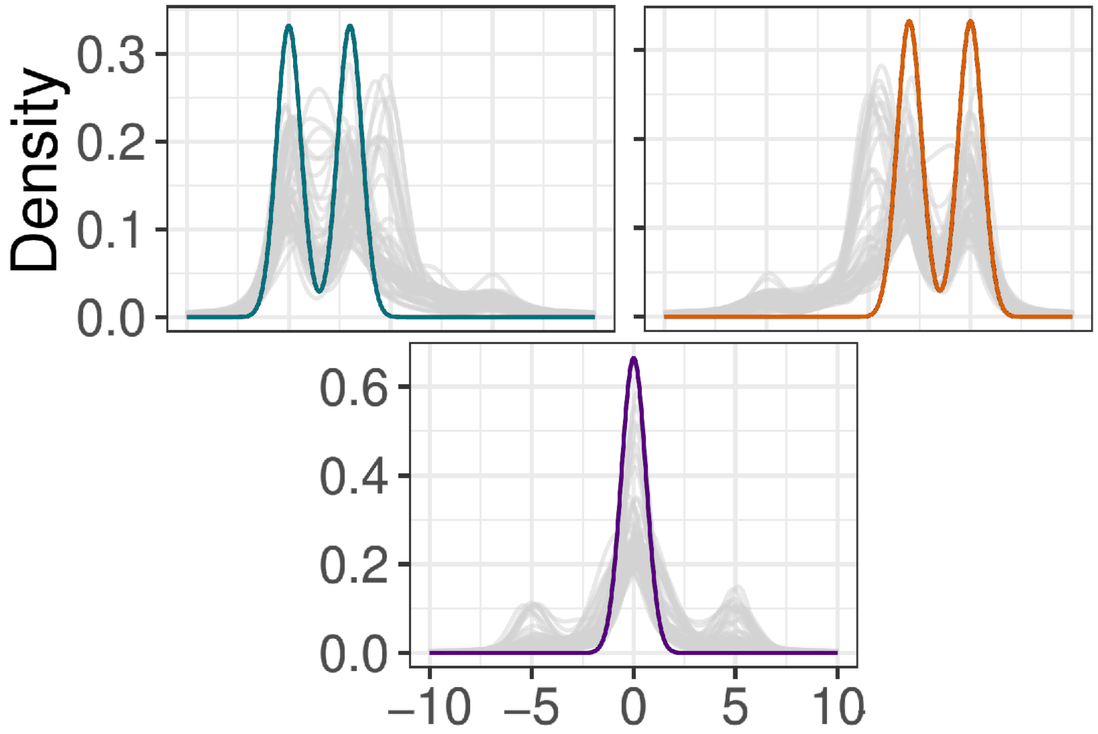
- Turns out that, for CAM, this prior correlation is pretty high

$$\rho_{j,j'}^{(CAM)} = 1 - \frac{1}{1+\alpha}\frac{\beta}{1+2\beta} \in (0.5, 1)$$

CAM and correlation - Toy example

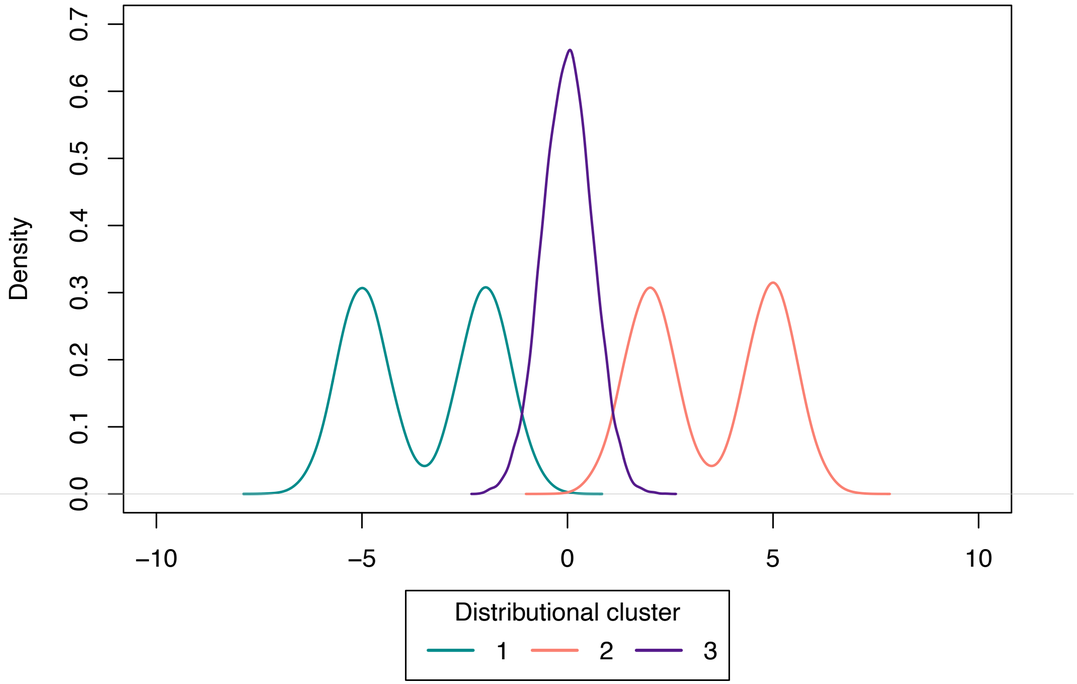
- Such high correlation a priori also has an **impact a posteriori**, especially with density estimation
- Consider the following toy example, where the distributions of different subpopulations have **no shared atoms** and a few observations per group (e.g., $n_j = 10$)

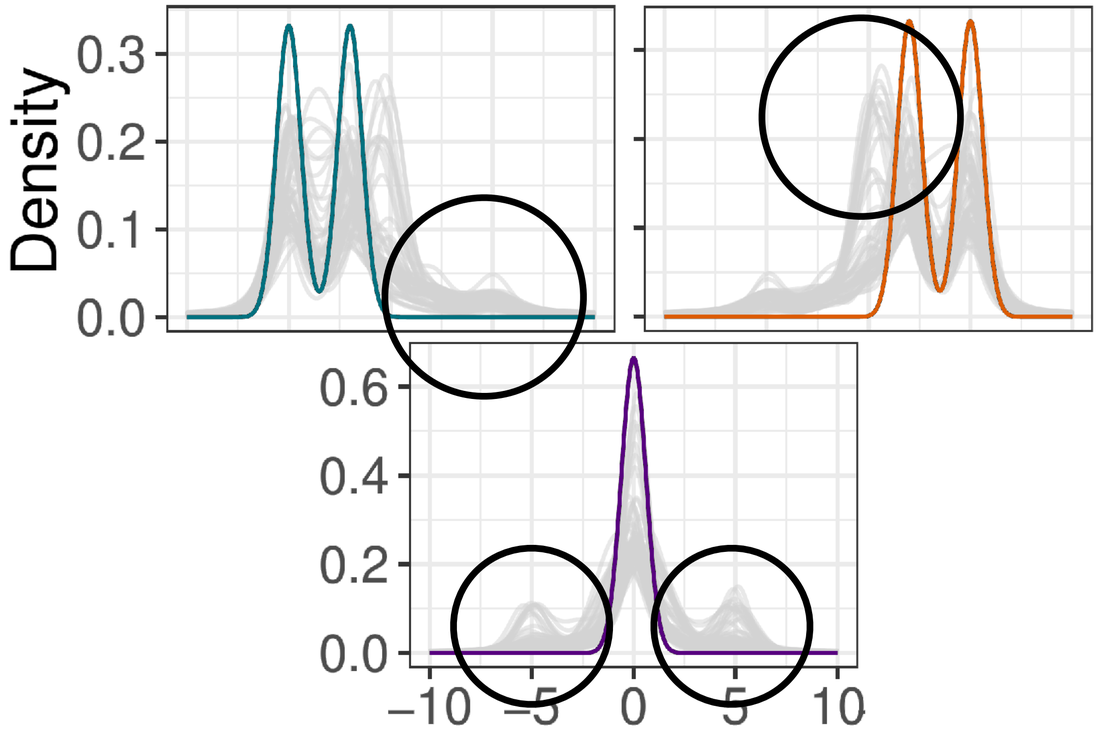




CAM and correlation - Toy example

- Such high correlation a priori also has an **impact a posteriori**, especially with density estimation
- Consider the following toy example, where the distributions of different subpopulations have **no shared atoms** and a few observations per group (e.g., $n_j = 10$)
- The posterior density estimates contain spurious modes





An Orwellian problem

- Weights arising from a SB construction have an implicit **decreasing stochastic ordering**

$$\mathbb{E}\left[\omega_{1,k}\right] \geq \mathbb{E}\left[\omega_{2,k}\right] \geq \mathbb{E}\left[\omega_{3,k}\right] \geq \mathbb{E}\left[\omega_{4,k}\right] \geq \ldots \qquad \forall k$$

- Therefore, the first atoms in the sequence have, in expectation, greater importance:

$$G_1^* = \sum_{l \geq 1} \omega_{l,1} \delta_{\theta_l^*} = \omega_{1,1}\delta_{\theta_1^*} + \omega_{2,1}\delta_{\theta_2^*} + \omega_{3,1}\delta_{\theta_3^*} + \ldots$$

$$G_2^* = \sum_{l \geq 1} \omega_{l,2} \delta_{\theta_l^*} = \omega_{1,2}\delta_{\theta_1^*} + \omega_{2,2}\delta_{\theta_2^*} + \omega_{3,2}\delta_{\theta_3^*} + \ldots$$

$$G_3^* = \sum_{l \geq 1} \omega_{l,3} \delta_{\theta_l^*} = \omega_{1,3}\delta_{\theta_1^*} + \omega_{2,3}\delta_{\theta_2^*} + \omega_{3,3}\delta_{\theta_3^*} + \ldots$$

- And this holds across all the distributional atoms G_k^*'s

- This could imply forced similarities in the G_j's, even when not motivated by the data

An Orwellian problem

- Weights arising from a SB construction have an implicit **decreasing stochastic ordering**

$$\mathbb{E}\left[\omega_{1,k}\right] \geq \mathbb{E}\left[\omega_{2,k}\right] \geq \mathbb{E}\left[\omega_{3,k}\right] \geq \mathbb{E}\left[\omega_{4,k}\right] \geq \ldots \qquad \forall k$$

- Therefore, the first atoms in the sequence have, in expectation, greater importance:

$$G_1^* = \sum_{l \geq 1} \omega_{l,1} \delta_{\theta_l^*} = \omega_{1,1} \delta_{\theta_1^*} + \omega_{2,1} \delta_{\theta_2^*} + \omega_{3,1} \delta_{\theta_3^*} + \ldots$$

$$G_2^* = \sum_{l \geq 1} \omega_{l,2} \delta_{\theta_l^*} = \omega_{1,2} \delta_{\theta_1^*} + \omega_{2,2} \delta_{\theta_2^*} + \omega_{3,2} \delta_{\theta_3^*} + \ldots$$

$$G_3^* = \sum_{l \geq 1} \omega_{l,3} \delta_{\theta_l^*} = \omega_{1,3} \delta_{\theta_1^*} + \omega_{2,3} \delta_{\theta_2^*} + \omega_{3,3} \delta_{\theta_3^*} + \ldots$$

- And this holds across all the distributional atoms G_k^*'s

- This could imply forced similarities in the G_j's, even when not motivated by the data

Nonparametric vs parametric solutions

Proposition 2.1 *Let* $G_j, G_{j'} \mid Q \sim Q$ *and* $Q \sim \text{geCAM}(\mathcal{L}(\boldsymbol{\pi}), \mathcal{L}(\boldsymbol{\omega}_k), H)$, *with* $j \neq j'$. *Then, the correlation between* G_j *and* $G_{j'}$ *is given by*

Generic SB laws

$$\begin{aligned}\rho_{j,j'} &:= Corr(G_j, G_{j'}) \\ &= 1 - \left(1 - \mathbb{P}\left[G_j = G_{j'}\right]\right) \left(\frac{\mathbb{P}\left[y_{i,j} = y_{i',j'} \mid G_j \neq G_{j'}\right]}{\mathbb{P}\left[y_{i,j} = y_{i',j'} \mid G_j = G_{j'}\right]}\right) \\ &= 1 - (1 - q_1)(1 - q_2)\end{aligned} \tag{3}$$

where $q_1 = \mathbb{P}\left[G_j = G_{j'}\right] = \sum_{k \geq 1} \mathbb{E}\left[\pi_k^2\right]$ *and*

$$q_2 = 1 - \frac{\mathbb{P}\left[y_{i,j} = y_{i',j'} \mid G_j \neq G_{j'}\right]}{\mathbb{P}\left[y_{i,j} = y_{i',j'} \mid G_j = G_{j'}\right]} = \frac{\sum_{l \geq 1} \mathbb{E}\left[\omega_{l,k}\right]^2}{\sum_{l \geq 1} \mathbb{E}\left[\omega_{l,k}^2\right]}.$$

Moreover, the correlation is always non-negative.

$(1 - q_2)$ is the **problematic term,** preventing the correlation from going below 0.5 - it stems only from the **observational weights**

- We studied the evolution of the correlations for different specifications:
  - PY, 2PB, atom-skipping process (PAM) by Bi and Ji, 2023 - we call it the **skip-breaking process**, mostly for marketing reasons :)
- But… what if, instead, we **go finite**?

Finite-infinite Shared Atoms Nested model

- Let $(y_{1,j}, \ldots, y_{n_j,j})$ indicate the scores of the songs of the j-th artist

$$y_{1,j}, \ldots, y_{n_j,j} \mid p_j \overset{ind.}{\sim} p_j, \qquad p_j(\,\cdot\,) = \int_{\Theta} f(\,\cdot \mid \theta) G_j(d\theta)$$

$$G_1, \ldots, G_J \mid Q \sim Q \qquad Q = \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}$$

$$G_k^* = \sum_{l=1}^{L} \omega_{l,k} \delta_{\theta_l^*} \qquad \theta_1^*, \theta_2^*, \ldots \sim G_0$$

$$\{\pi_k\}_{k \geq 1} \sim GEM(\alpha) \qquad \{\omega_{l,k}\}_{l=1}^{L} \sim Dirichlet_L(\beta)$$

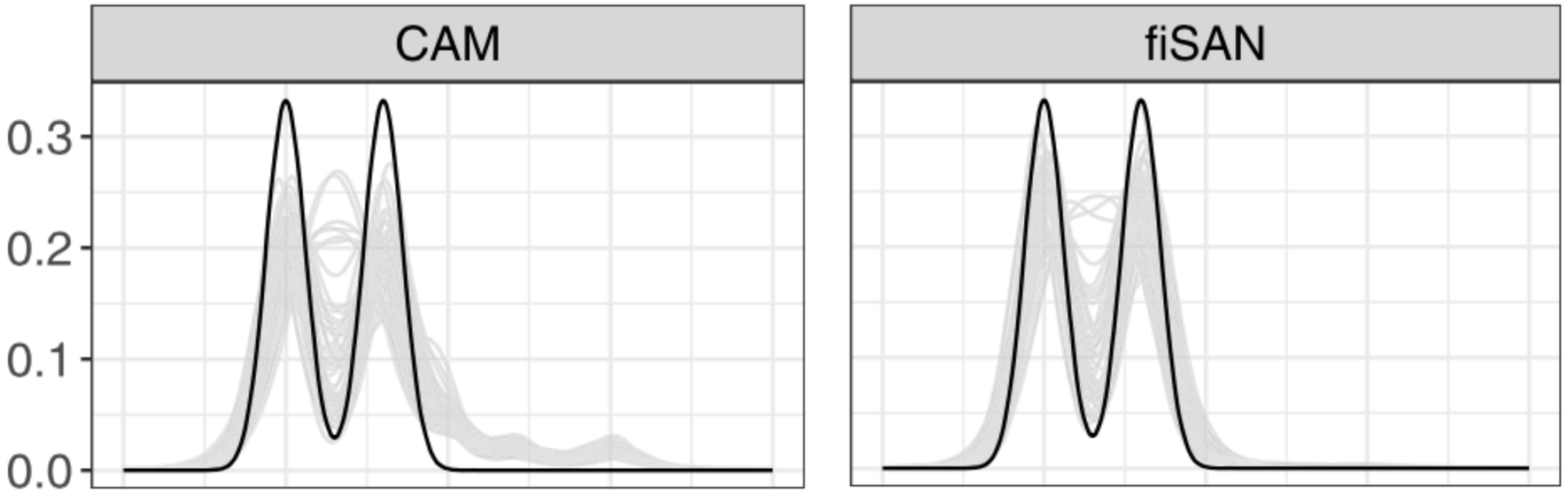
- We studied the behavior of this specification and adopted sparse finite mixtures
- Albeit it seems a simplification, it has an important impact on the model

- One can show that

$$\rho_{j,j'}^{(fiSAN)} = 1 - \frac{1}{1+\alpha} \frac{L-1}{(b+1)L} \in (0,1)$$

- We called the **symmetric atoms**, “freed by the stochastic ordering” **shared** atoms, thus **SAN: Share Atoms Nested** model

- This “simplification” does the trick, keeping the model tractable and interpretable



- One can show that

$$\rho_{j,j'}^{(fiSAN)} = 1 - \frac{1}{1+\alpha} \frac{L-1}{(b+1)L} \in (0,1)$$

- We called the **symmetric atoms**, “freed by the stochastic ordering” **shared** atoms, thus **SAN: Share Atoms Nested** model

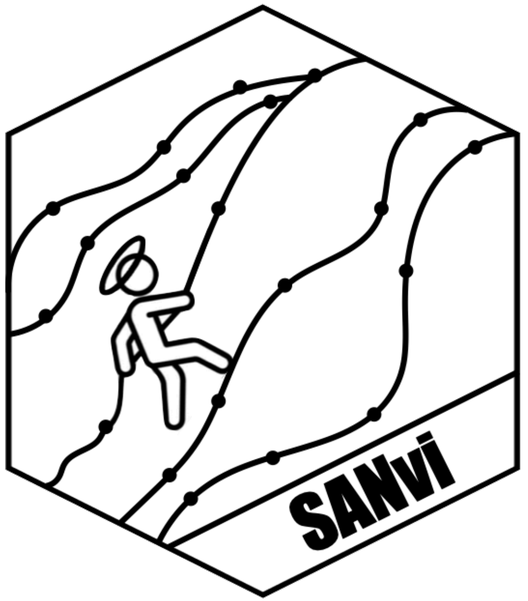
- This “simplification” does the trick, keeping the model tractable and interpretable

- Plus: this model specification lends itself to the derivation of **a mean-field variational inference algorithm** that

    - Allows us to scale the application to large datasets

    - Provides point estimates of parameters and partitions, avoiding the label-switching problem

- R package**s** available on CRAN! Currently working on **SANBA**



Outline

- **Pose and Poseidon**: Bayesian model for biclustering **large** **imaging data**
  - MALDI-Mass Spectrometry Imaging (MSI) mouse brain data
    - **Biclustering and grouped data**: separate exchangeability
    - **Image Segmentation**: accounting for spatial information with **HMRF**
    - **Large dataset**: fast estimation via mean-field variational inference
- **SCDC**: a model for **simultaneous clustering** and **deconvolution** of calcium traces
  - Calcium imaging data
    - **Spatially clustering** neurons via **PSB**
    - Deconvolution and **detection of spike trains** via latent **GPs** for temporal dependence