# Métodos de estadística computacional y machine learning para ciencias de la vida, con una aplicación a COVID-19

Gonzalo E. Mena

May 20th, 2020

Data Science Initiative and Statistics Department, Harvard University

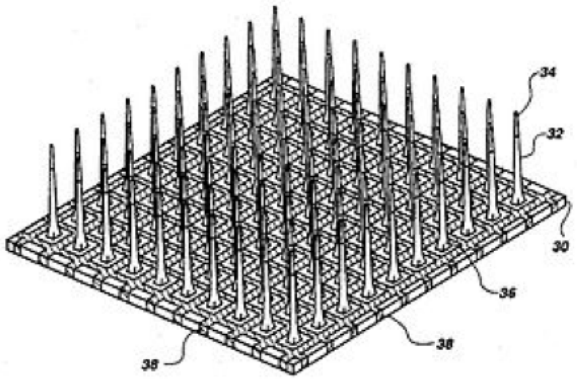
# Palabras iniciales (advertencias)

- Charla académica, resumen sobre trabajos aplicados en ciencias de la vida (neurociencia).
- Al final: algunos métodos para COVID-19. Más preguntas que respuestas. Mostraré algunos datos y análisis muy preliminares sin calibrar.
- Objetivo: incentivar discusión y motivar trabajo en el área y el uso de **modelos bayesianos.**
- Spanglish

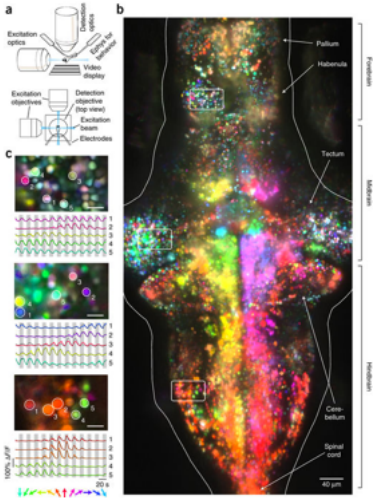
# World's current situation

- Several large-scale imaging and stimulation technologies. To read and write neural activity.

Electrophysiology




Calcium imaging



Freeman et. al, 2014

# World's current situation

- Several large-scale imaging and stimulation technologies. To read and write neural activity.
- Consensus on relevance.


Goal: to develop new experimental tools that will revolutionize our understanding of the brain.

# World's current situation

- Several large-scale imaging and stimulation technologies. To read and write neural activity.
- Consensus on relevance.

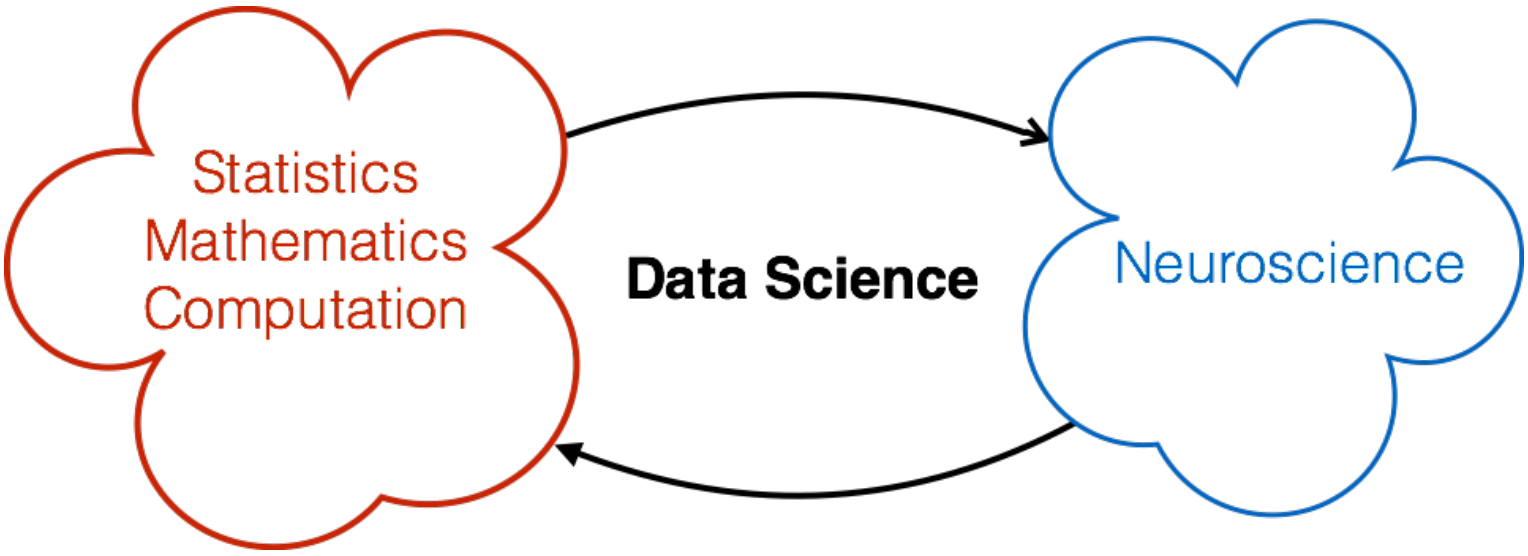

Goal: to develop new experimental tools that will revolutionize our understanding of the brain.

**Major bottleneck:**

data analysis capabilities are much below high-throughput data collection rates (TB's/hour).

Cannot fully exploit the potential of these technologies.

# This talk: (Neural) Data Science + COVID-19 (at the end)




**Claim**

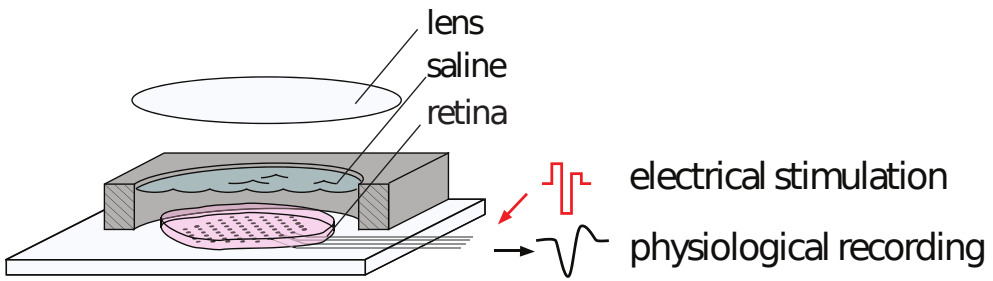
The dialog between life sciences (neuroscience) and Statistics/Mathematics/Computation is of mutual benefit.

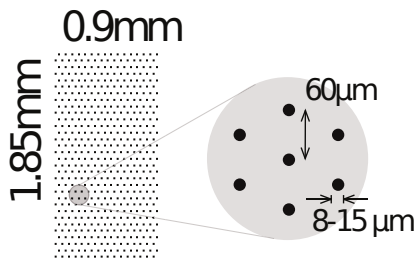
Here, **Bayesian Statistics**

# Large-scale Spike Sorting with Stimulation Artifacts

## Overarching goal

Stimulation and recording in large multi-electrode arrays (MEA) to read and write neural activity to achieve control.



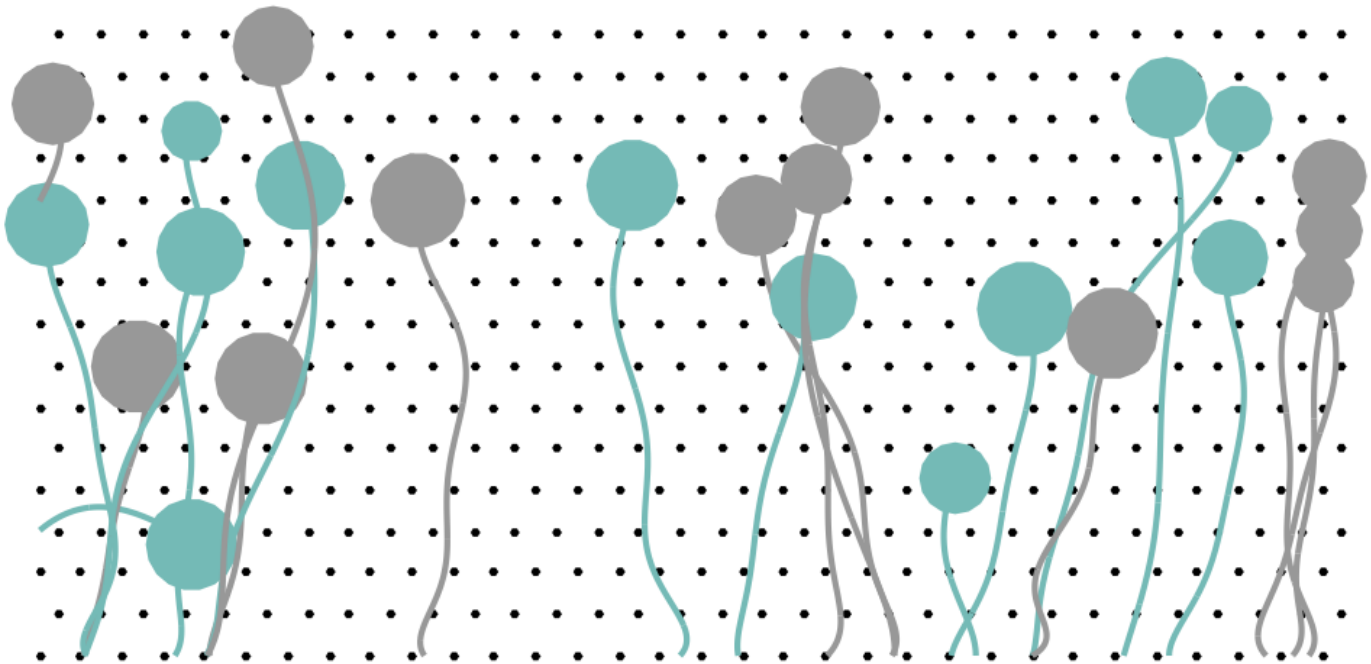




- For **control** need to know the stimulus→ response map **fast**.
- Large-scale, online data analysis. 512 electrodes, 20 Khz $\sim$ 50 GB/hour.
- Scientific and **Clinical** significance: development of high-resolution retinal prosthesis.

# Tailored activation

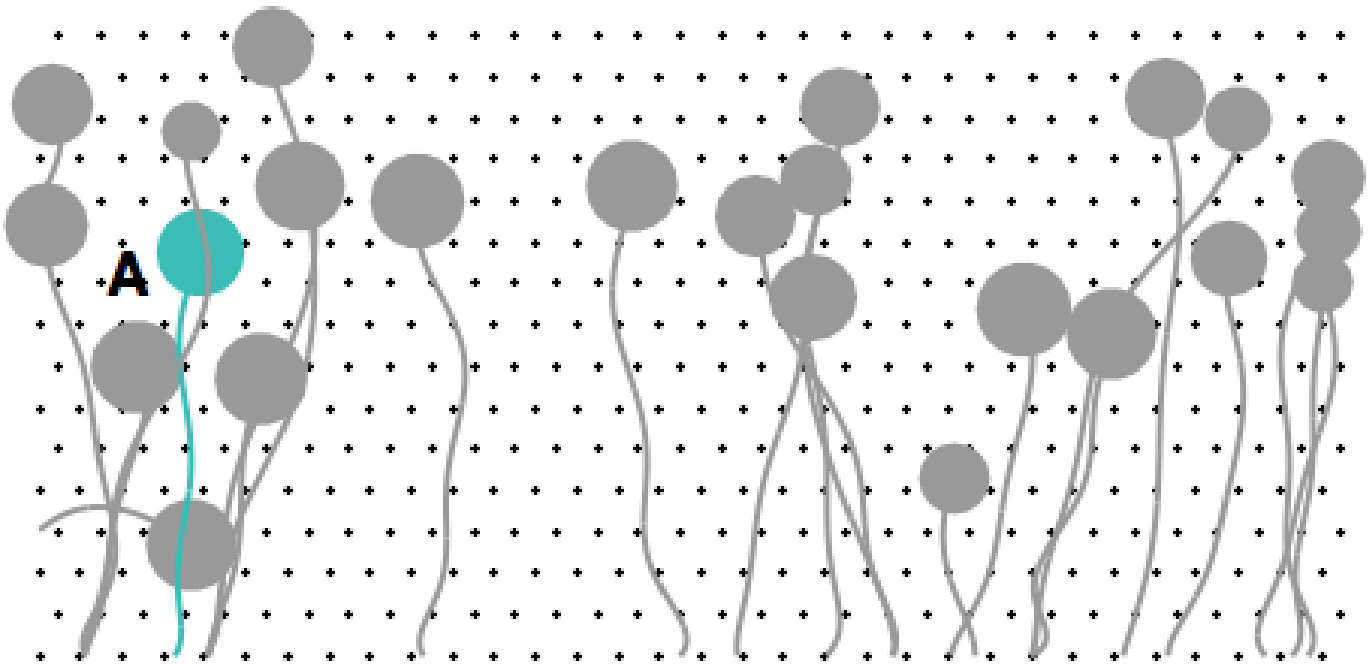
Goal: To generate artificial vision, elicit arbitrary patterns of neural activity with tailored stimuli.

# Tailored activation



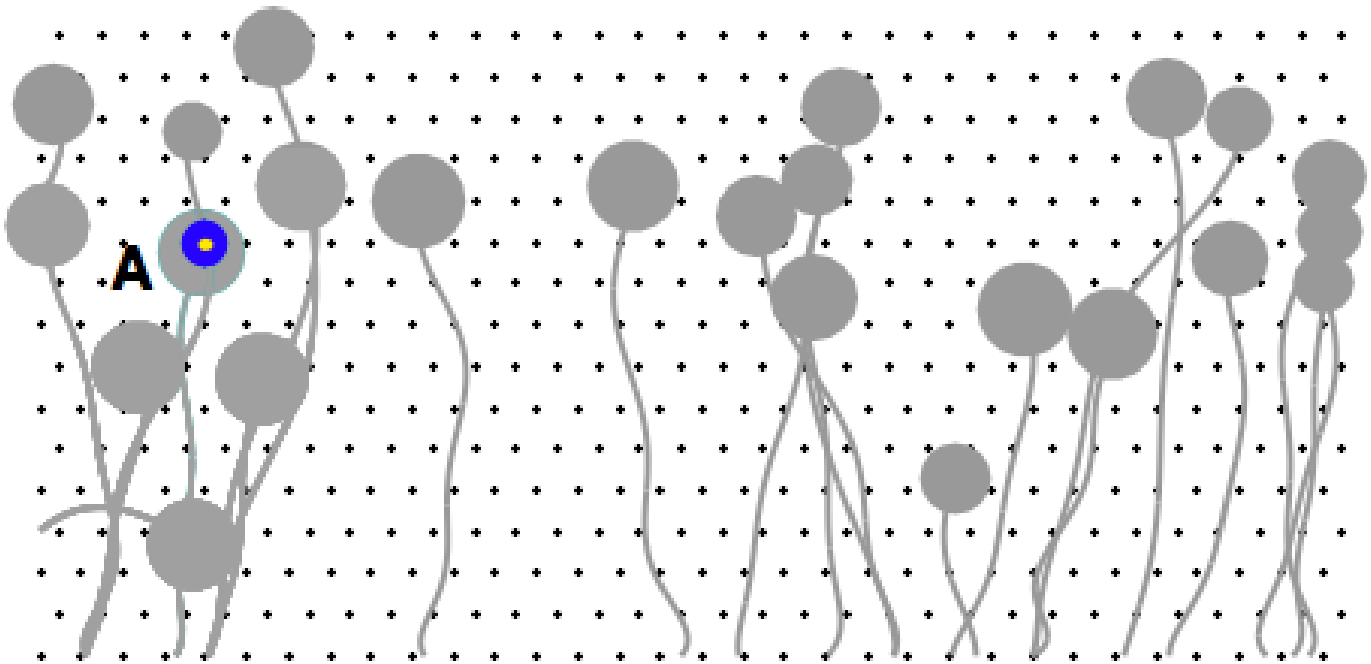
Question: Is it possible to activate *only* the colored neurons?

# Tailored activation



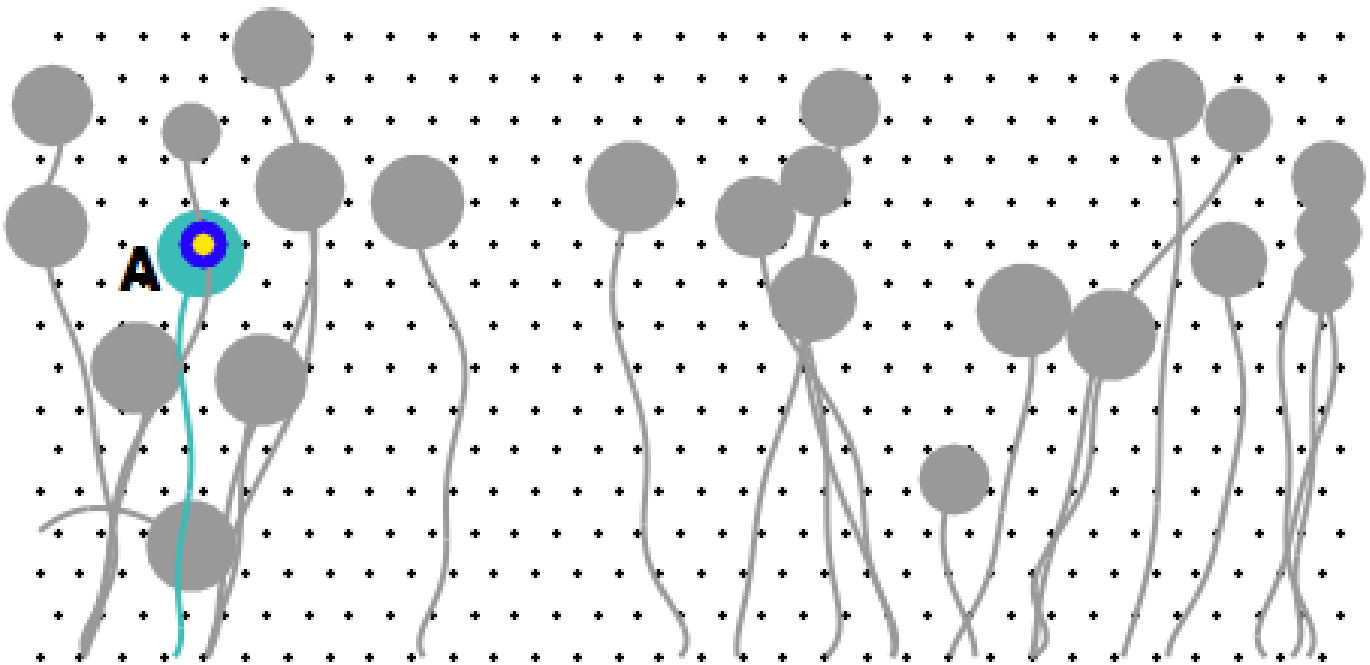

Easier question: is it possible to activate *only* neuron A?

# Tailored activation



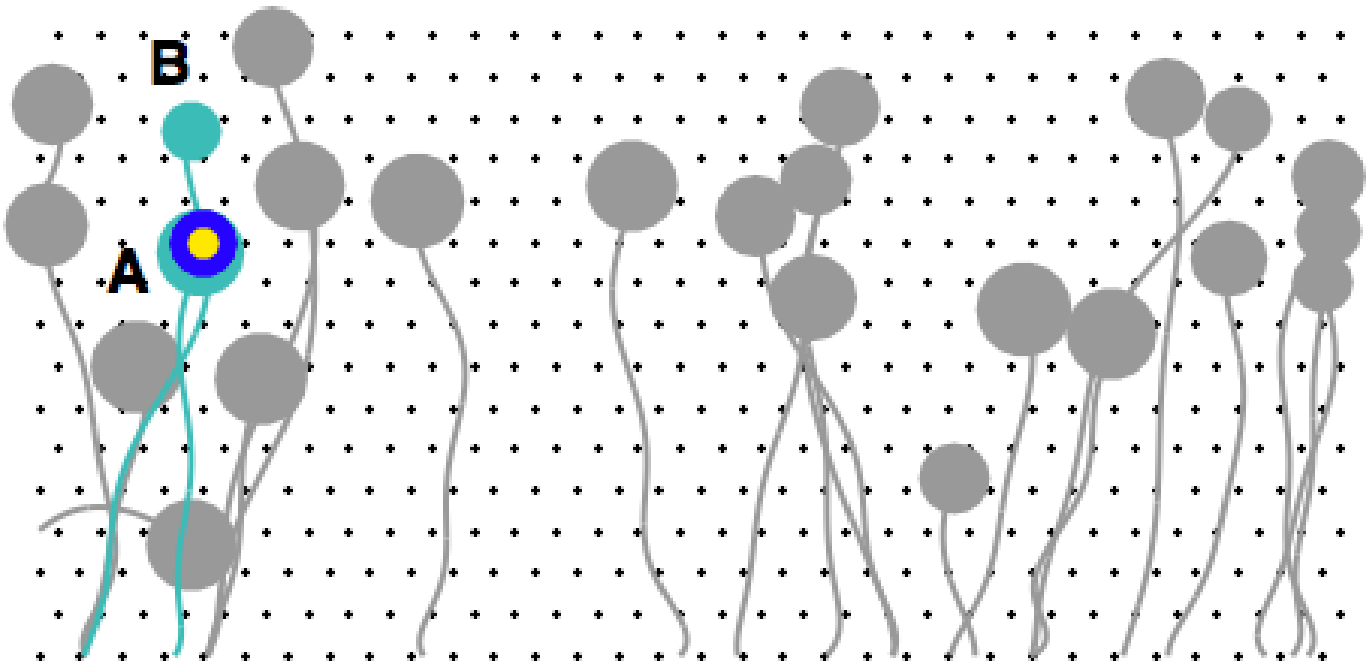

Stimulating with a pulse of $0.5\mu A$ on the electrode around the soma does not activate neuron A.

# Tailored activation



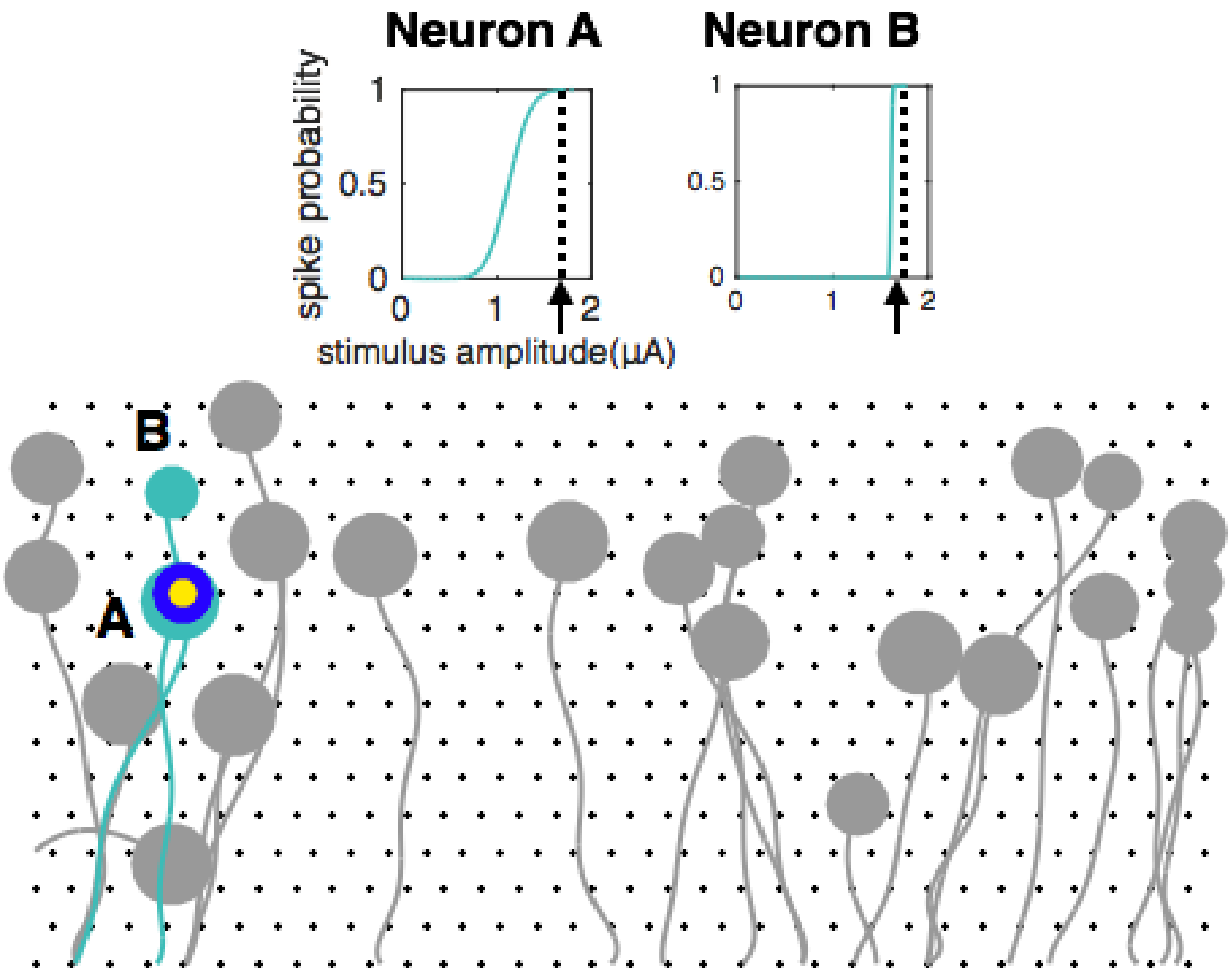

However, stimulating with $1.0\mu A$ does activate the neuron.

# Tailored activation




Further, stimulating with $1.5\mu A$ also activates nearby neuron B, through its axon.

# Tailored activation



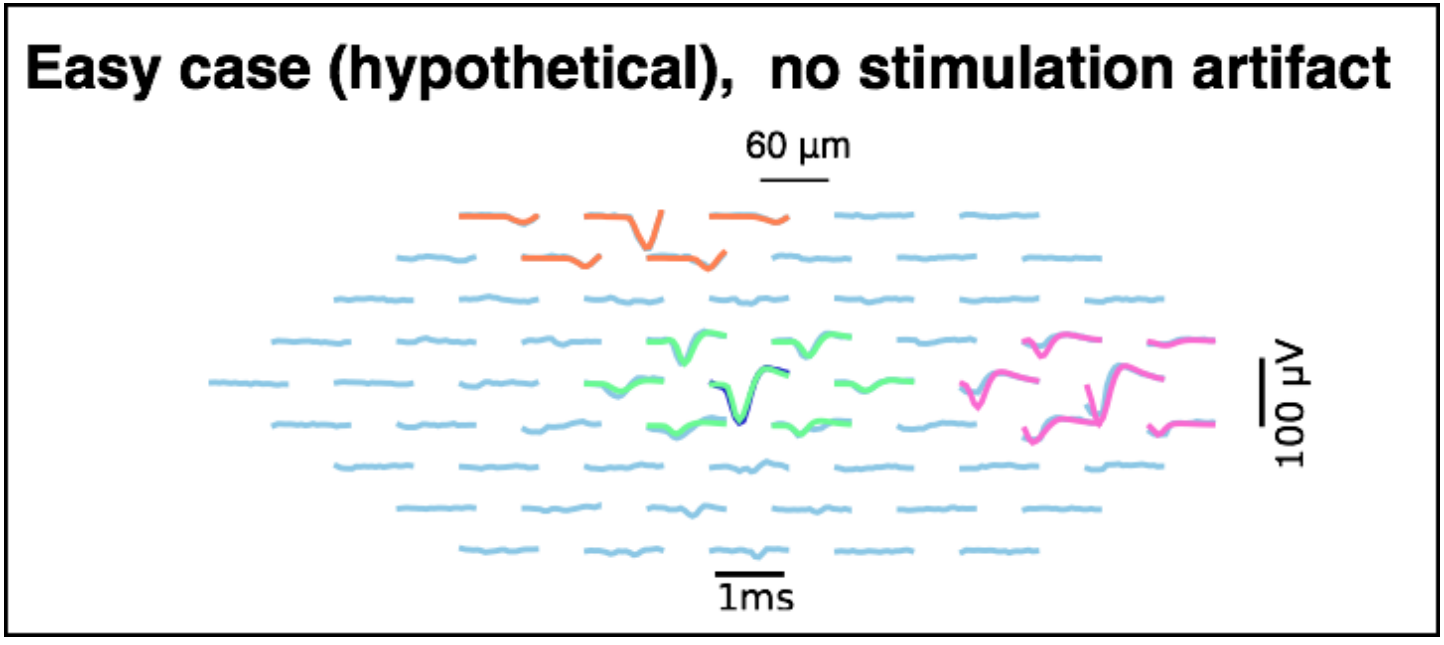

*Activation curves* summarize responsiveness of neurons. Inferred from many increasing stimuli.

# Stimulation artifacts

**Major hurdle:** electrical stimuli are sensed in electrodes as **artifacts**, stymying identification of neural activity.

# Stimulation artifacts

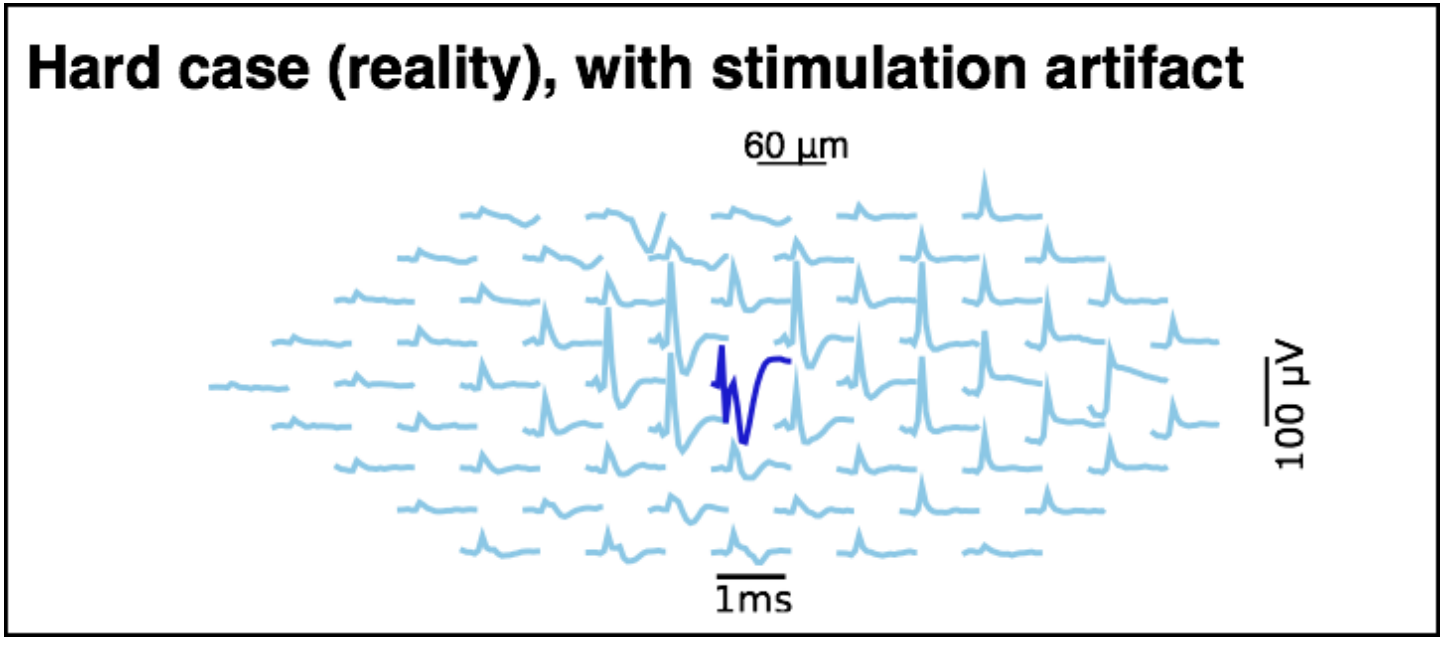
**Major hurdle:** electrical stimuli are sensed in electrodes as **artifacts**, stymying identification of neural activity.



# Stimulation artifacts

**Major hurdle:** electrical stimuli are sensed in electrodes as **artifacts**, stymying identification of neural activity.

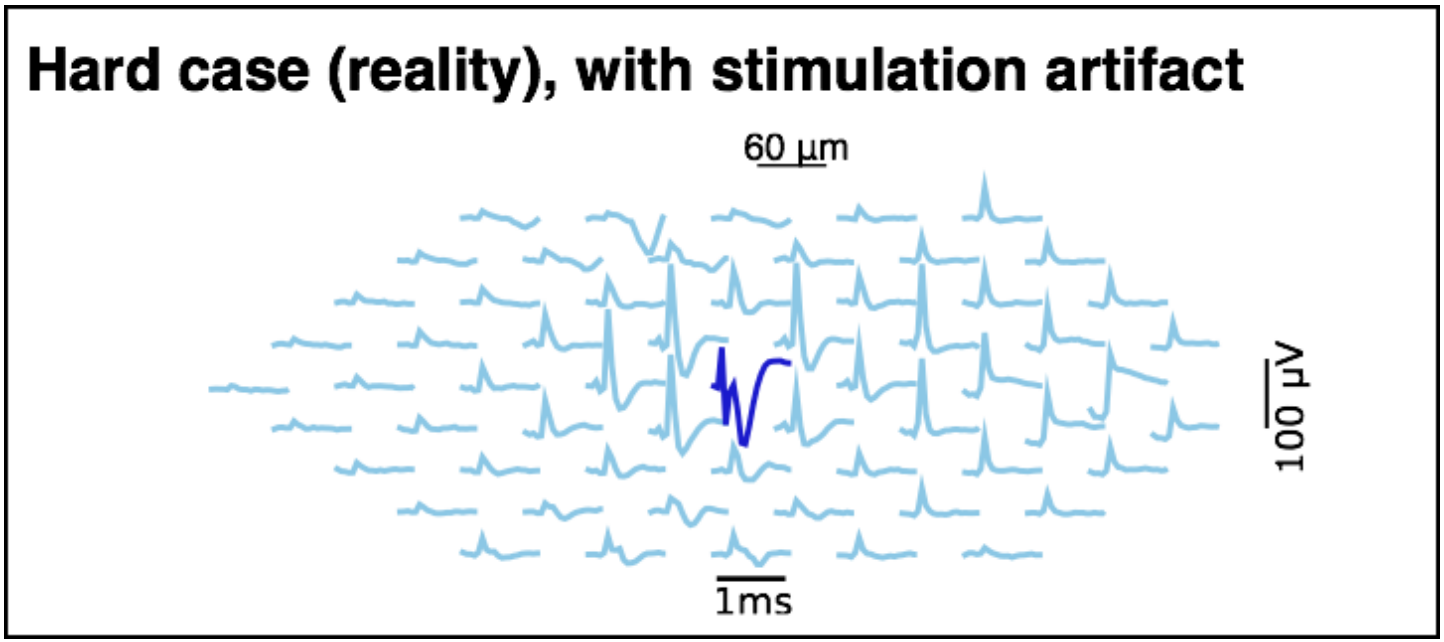
- Artifacts are much larger than spikes, overlap temporally with them.



# Stimulation artifacts

**Major hurdle:** electrical stimuli are sensed in electrodes as **artifacts**, stymying identification of neural activity.

- Artifacts are much larger than spikes, overlap temporally with them.




Current solutions break down.

Can take weeks to a human. Not online.

# Stimulation Artifacts

**Problem**

Data contains a *nuisance* parameter $A$,

$$Y = A + s + \epsilon,$$

Recorded traces $Y$, artifact $A$, neural activity $s$ and noise $\epsilon$.
To infer $s$ need to know $A$.

# Stimulation Artifacts

## Problem

Data contains a *nuisance* parameter $A$,

$$Y = A + s + \epsilon,$$

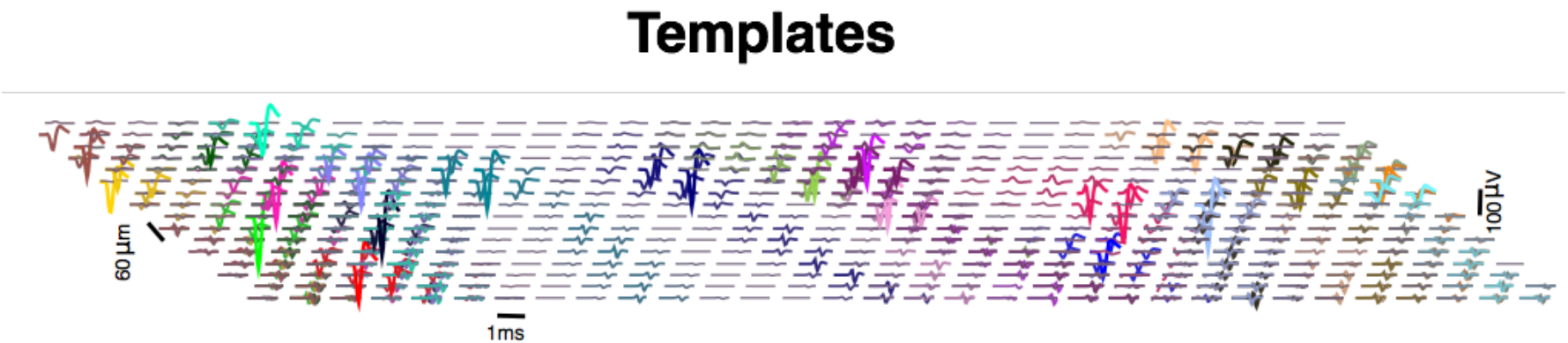
Recorded traces $Y$, artifact $A$, neural activity $s$ and noise $\epsilon$.
To infer $s$ need to know $A$.

## Solution

**Impose structure and prior knowledge in $A, s,$ and $\epsilon$ so $\hat{A}, \hat{s}$ can be resolved.**

# Neural activity structure

- Spike sorting of spontaneous activity to identify neurons.
- Provide us with templates (or spikes, or action potentials waveforms)

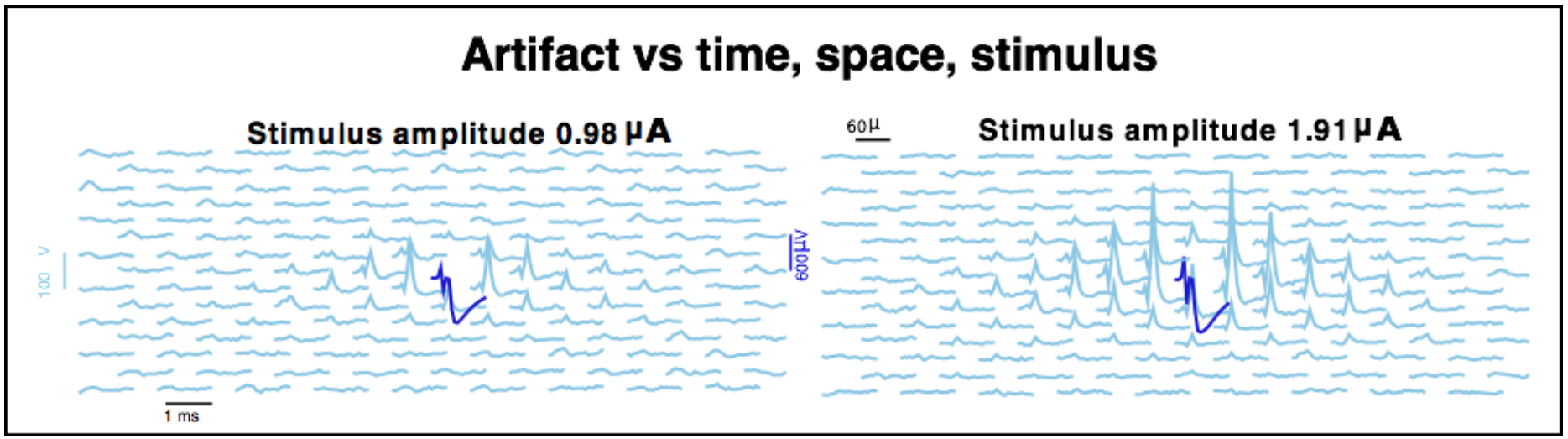


# The structure of stimulation artifacts

- Properties are revealed by silencing neural activity.

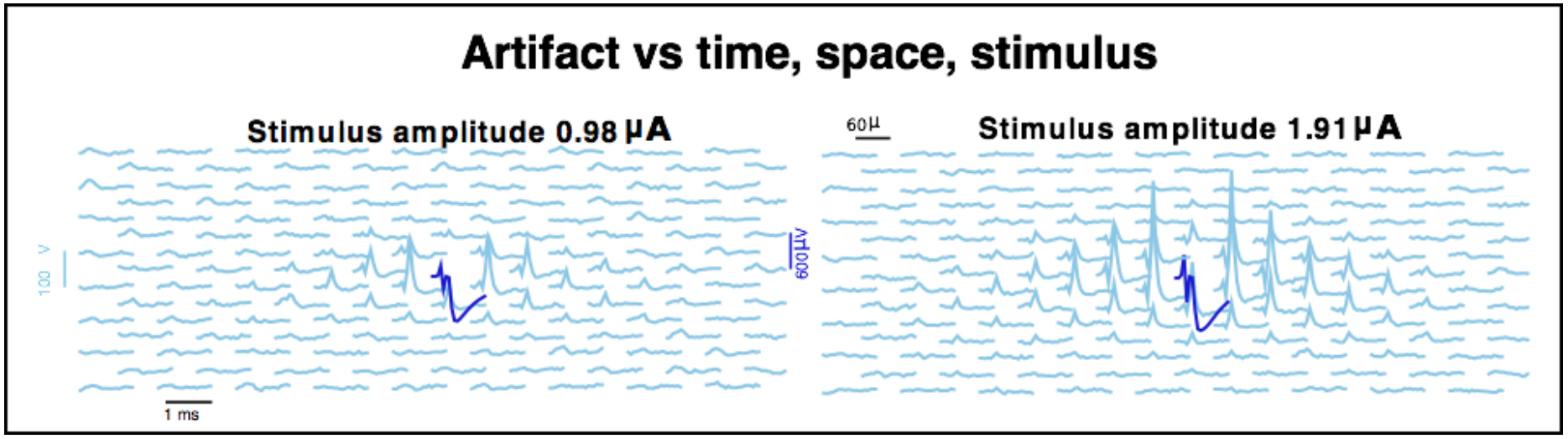
# The structure of stimulation artifacts

- Properties are revealed by silencing neural activity.
- Decays smoothly with distance from stimulating electrode and has a peak in time. Increases with strength of stimulus. Doesn't change if stimulus is the same.



# The structure of stimulation artifacts

- Properties are revealed by silencing neural activity.
- Decays smoothly with distance from stimulating electrode and has a peak in time. Increases with strength of stimulus. Doesn't change if stimulus is the same.




Non-linear and non-stationary, but smooth and structured.

# Crafting a principled solution

Consider the model

$$Y = A + s + \epsilon,$$

- Data $Y = Y_{t,e,j,i}$ over **time** ($1 \leq t \leq T$), **space** (electrode, $1 \leq e \leq E$), **strength** ($1 \leq j \leq J$) and **trial** ($1 \leq i \leq I$) dimensions.

# Crafting a principled solution

Consider the model

$$Y = A + s + \epsilon,$$

- Data $Y = Y_{t,e,j,i}$ over **time** ($1 \leq t \leq T$), **space** (electrode, $1 \leq e \leq E$), **strength** ($1 \leq j \leq J$) and **trial** ($1 \leq i \leq I$) dimensions.

**Imposing structure**

- Represent neural activity $s$ with Toeplitz matrices (shapes) and binary vectors (timing).

# Crafting a principled solution

Consider the model

$$Y = A + s + \epsilon,$$

- Data $Y = Y_{t,e,j,i}$ over **time** ($1 \leq t \leq T$), **space** (electrode, $1 \leq e \leq E$), **strength** ($1 \leq j \leq J$) and **trial** ($1 \leq i \leq I$) dimensions.

## Imposing structure

- Represent neural activity $s$ with Toeplitz matrices (shapes) and binary vectors (timing).
- Gaussian process (GP) to encode prior knowledge of artifact $A \sim GP(0, K^{\theta})$, and to borrow strength.

# Crafting a principled solution

Consider the model

$$Y = A + s + \epsilon,$$

- Data $Y = Y_{t,e,j,i}$ over **time** ($1 \leq t \leq T$), **space** (electrode, $1 \leq e \leq E$), **strength** ($1 \leq j \leq J$) and **trial** ($1 \leq i \leq I$) dimensions.

## Imposing structure

- Represent neural activity $s$ with Toeplitz matrices (shapes) and binary vectors (timing).
- Gaussian process (GP) to encode prior knowledge of artifact $A \sim GP(0, K^{\theta})$, and to borrow strength.
- Problem: $n \approx 10^6$ artifact variables, $O(n^3)$ does not scale.

# Crafting a principled solution

Consider the model

$$Y = A + s + \epsilon,$$

- Data $Y = Y_{t,e,j,i}$ over **time** ($1 \leq t \leq T$), **space** (electrode, $1 \leq e \leq E$), **strength** ($1 \leq j \leq J$) and **trial** ($1 \leq i \leq I$) dimensions.

## Imposing structure

- Represent neural activity $s$ with Toeplitz matrices (shapes) and binary vectors (timing).
- Gaussian process (GP) to encode prior knowledge of artifact $A \sim GP(0, K^{\theta})$, and to borrow strength.
- Problem: $n \approx 10^6$ artifact variables, $O(n^3)$ does not scale.
- Solution: Kronecker decomposition

$$K^{(\theta,\phi^2)} = \rho K_t \otimes K_e \otimes K_j + \phi^2 I.$$

- Each kernel must represent smoothness and non-stationarity.

# Algorithm

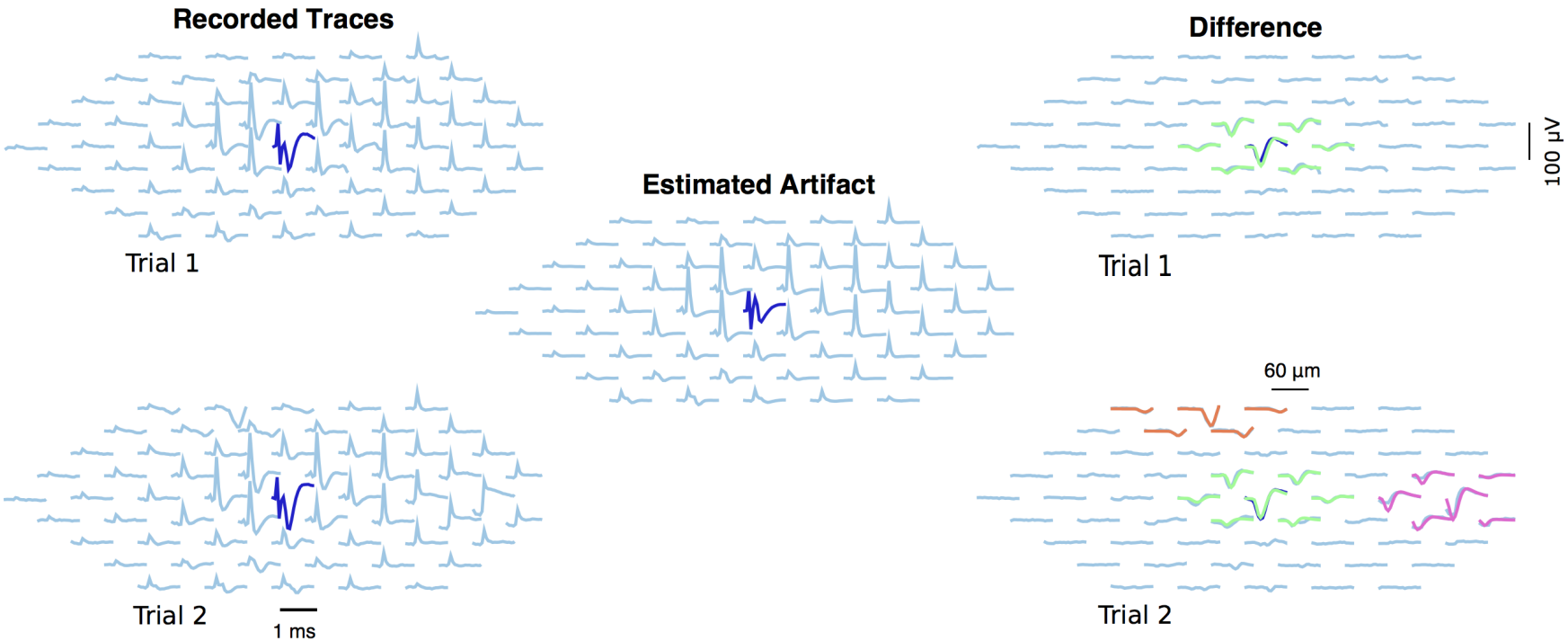
Goal: Obtain $\hat{A}, \hat{s}$ from the model $Y = A + s + \epsilon, A \sim GP(0, K^{\hat{\theta}})$

- Produce estimates increasingly in $j$ (strength).
  - Rationale: at lowest strengths $A$ is better behaved and easier to estimate.
  - Initial guess $\hat{A}^0_{j+1}$ is the extrapolation from $\hat{A}_{[1,j]}$.
- Given $j$, alternate between maximizing $p(s_j | Y_j, \hat{A}_j, \hat{\theta})$ for $\hat{s}_j$ and maximizing $p(A_j | Y_j, \hat{s}_j, \hat{\theta})$ for $\hat{A}_j$.
  - $\hat{s^n_{j,i}}$ given $\hat{A}_j$: $s^n_{j,i} = T^n b_{j,i}$ are binary vectors; do greedy template matching.

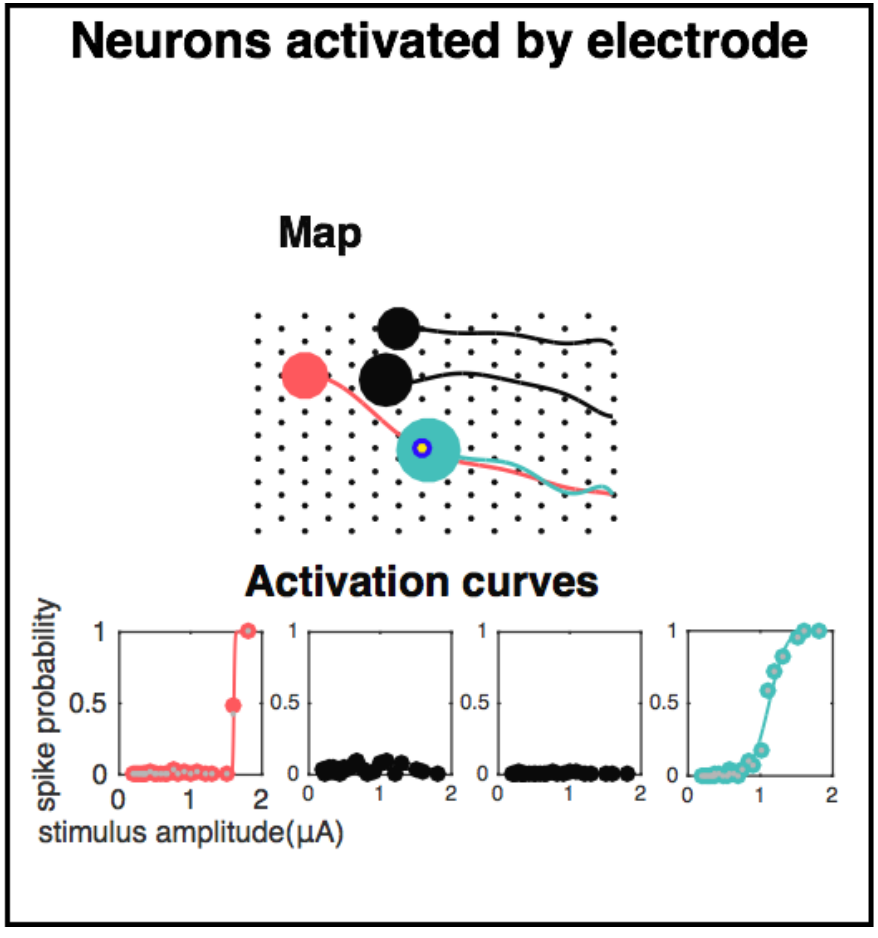
$$\min_{b^n_{j,i}} \left\| (Y_{j,i} - \hat{A}_j) - \sum_n T^n b_{j,i} \right\|^2 .$$

  - $\hat{A}_j$ given $\hat{s}_j$ via filtering (posterior mean) of spike-subtracted traces.

# Example of sorting

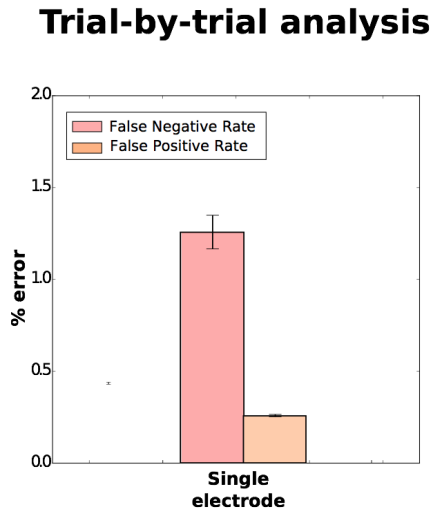


# Large-scale automatic analysis




Gray dots indicate human judgement.

# Population results



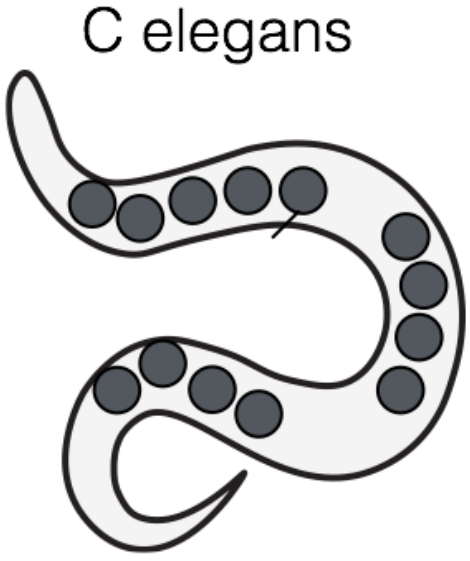

**1,713,233 trials.**

- **Accuracy greater than 99.5%, also agreement in latencies.**[1]
- **Past: weeks$\rightarrow$ Now: $\approx$15 minutes. Compatible with online control experiments.**
- **Enhanced capabilities of technology.**

[1] Mena et al., PLOS computational Biology, 2017.

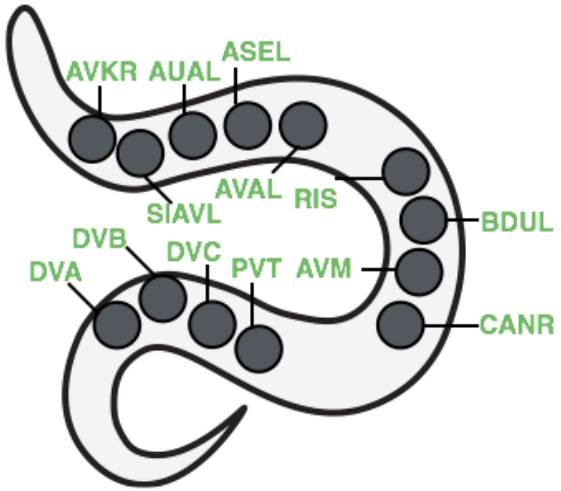
# Probabilistic neural identity inference in C.elegans

# The relevance of C.elegans

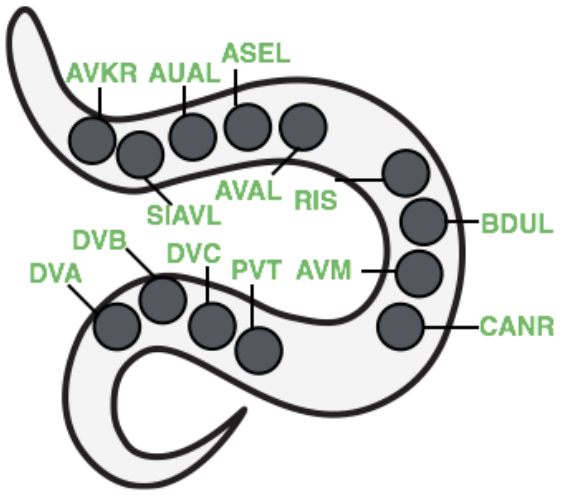


# The relevance of C.elegans

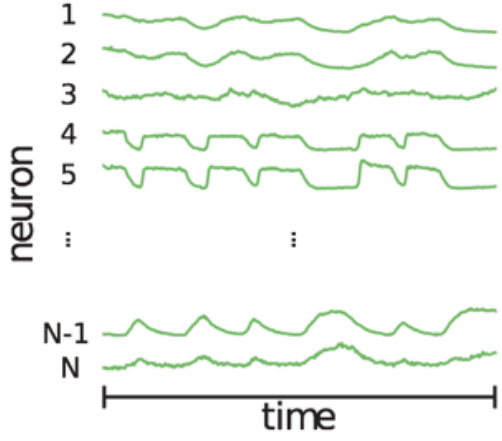
Always same neurons (302)



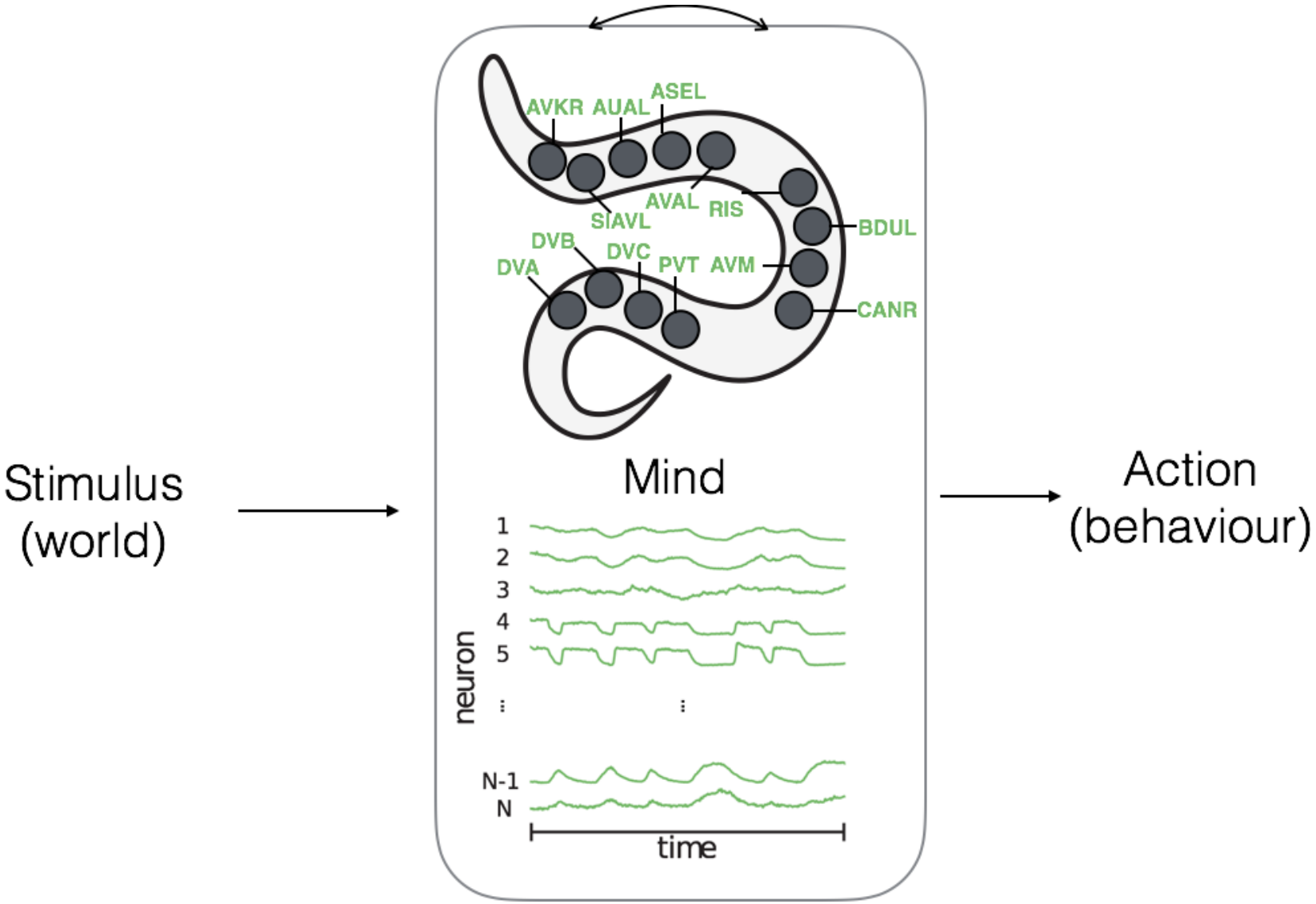
# The relevance of C.elegans




Activity is recorded as time series

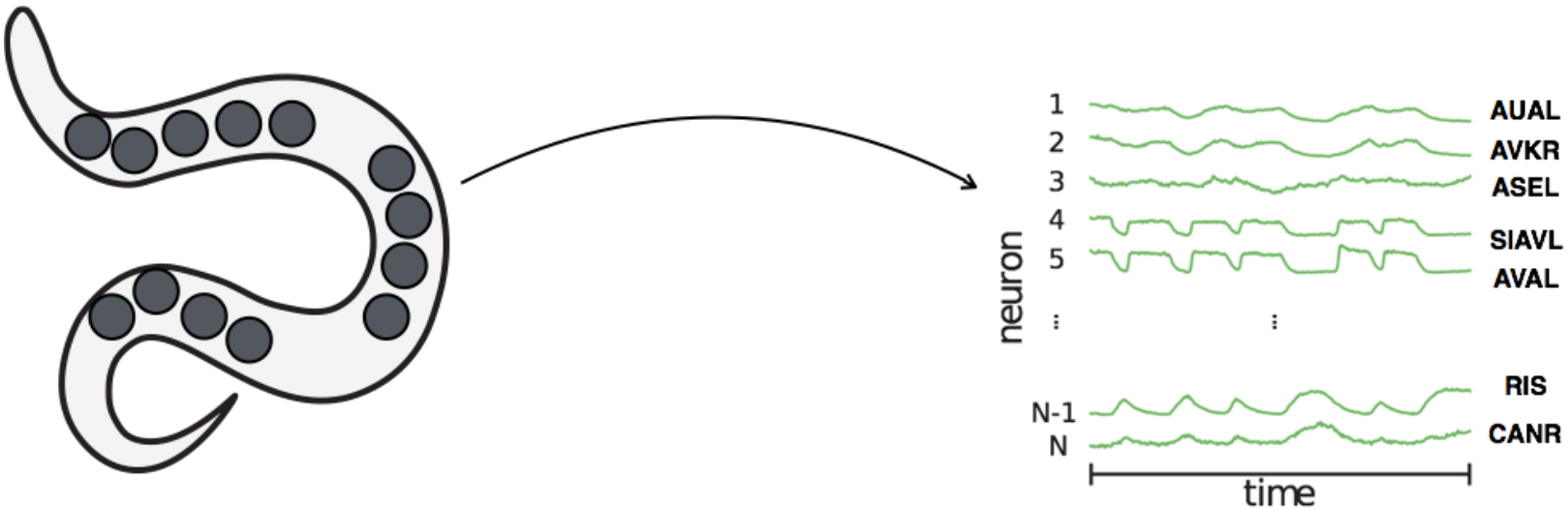


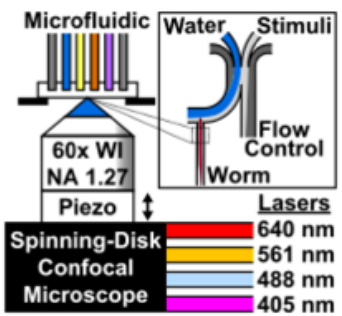
# The relevance of C.elegans



# A data processing pipeline

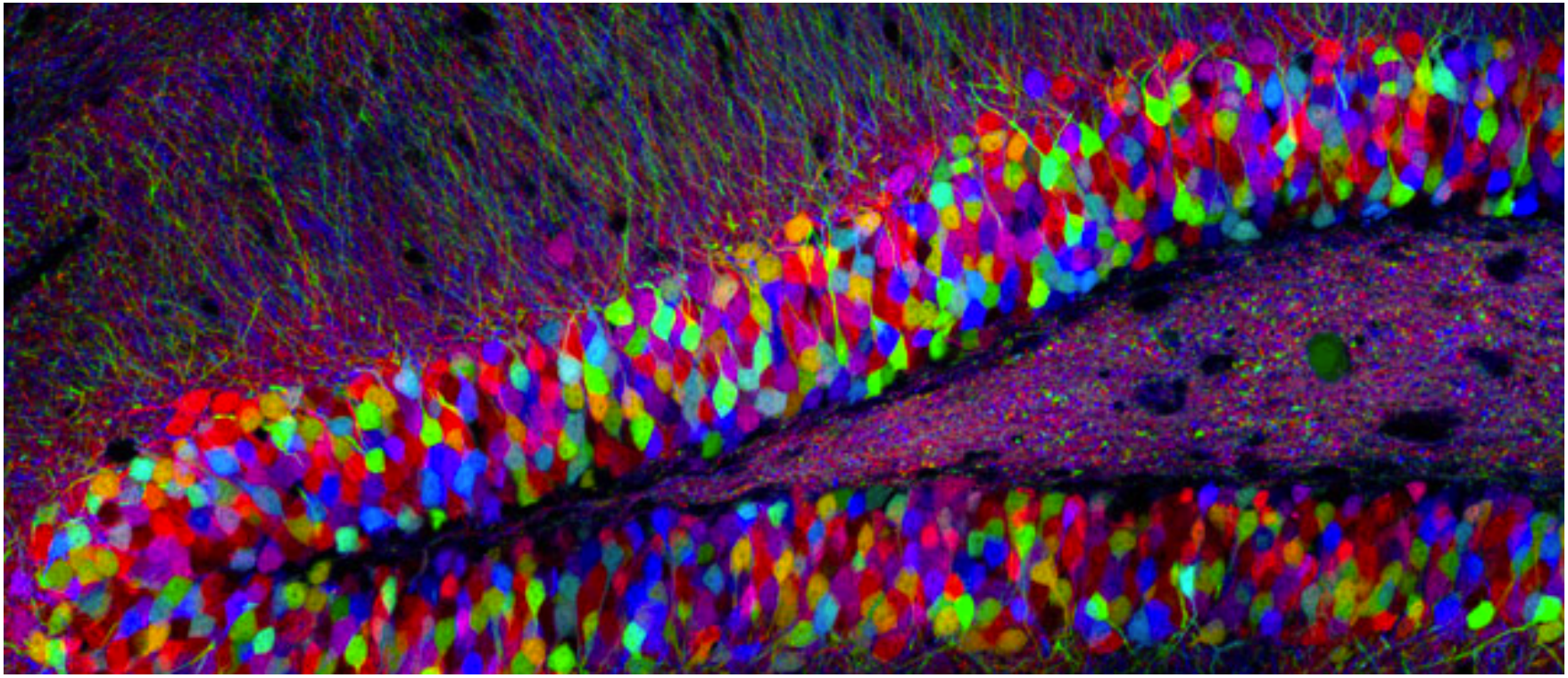
- Raw data: 5D point processes (space x time x color)
- First step: finding neurons.
- Second step: **identifying** neurons



# Find neurons with the help of color

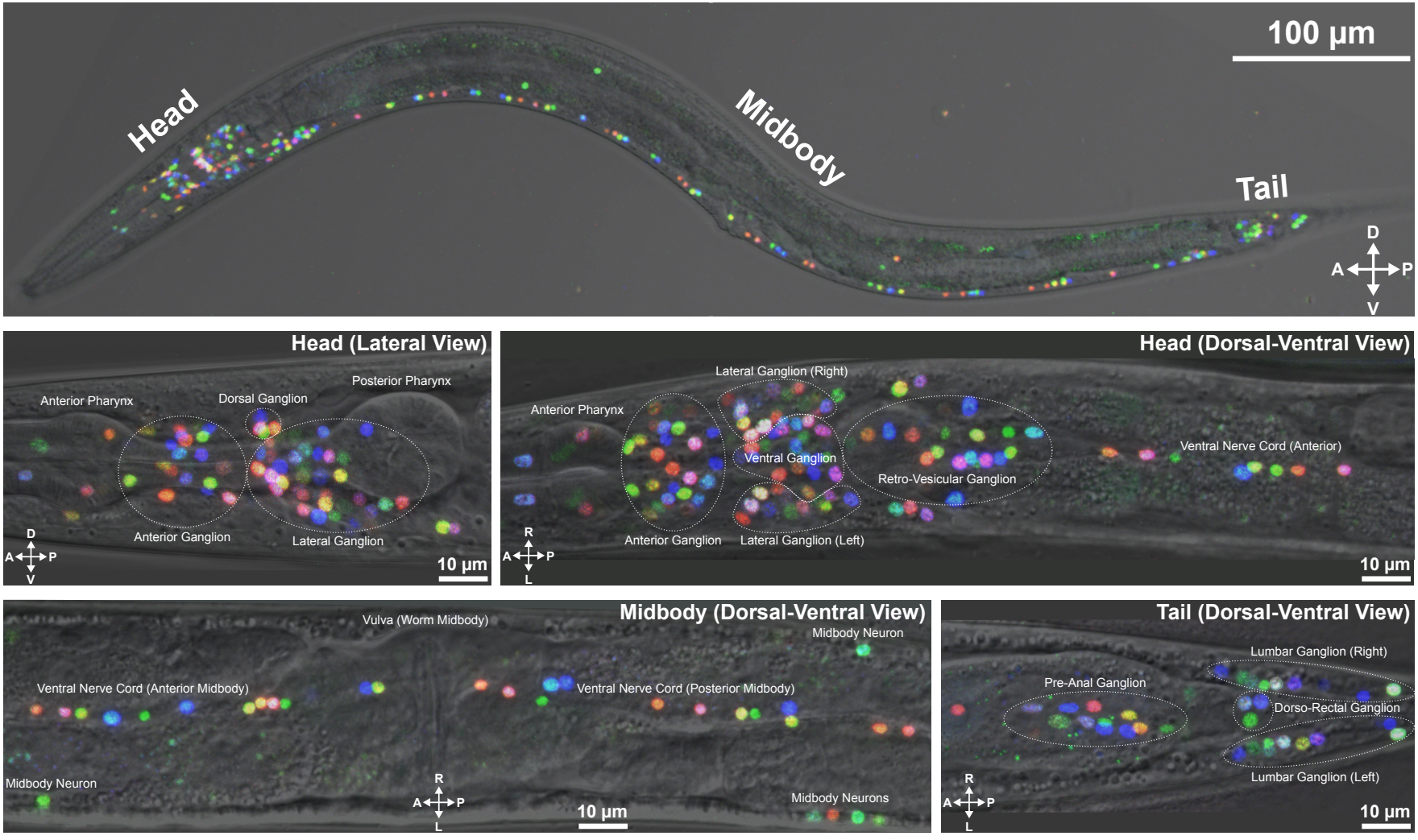
Brainbow (Lichtman and Sanes, 2008) **stochastic** coloring of neurons



Tamily Weissman, 2008 Photomicrography competition

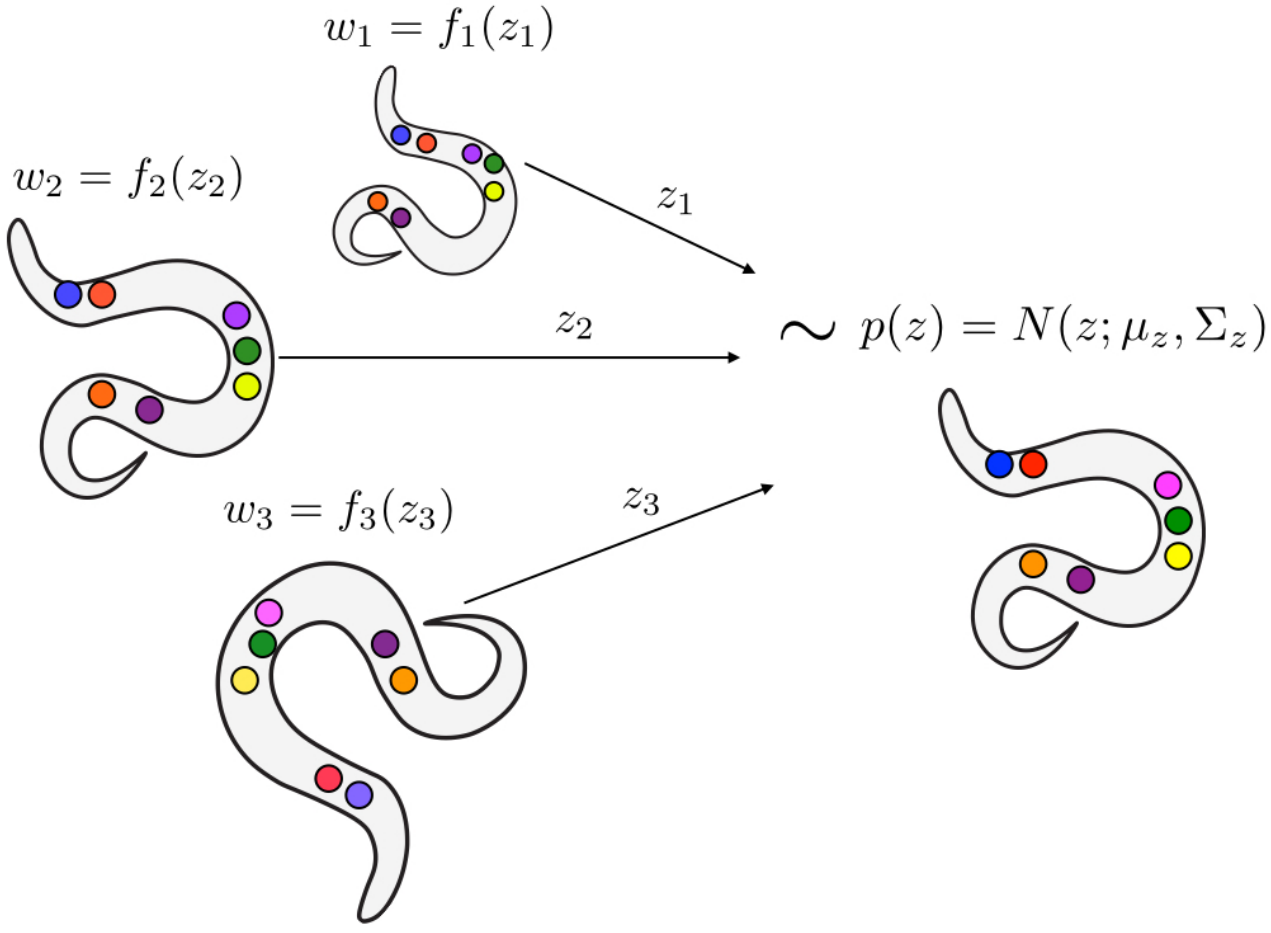
# Neural identification: NeuroPal

NeuroPal: A Neuronal Polychromatic Atlas of Landmarks for Whole-Brain Imaging in C. elegans. Now 'deterministic' colors.



# A canonical representation

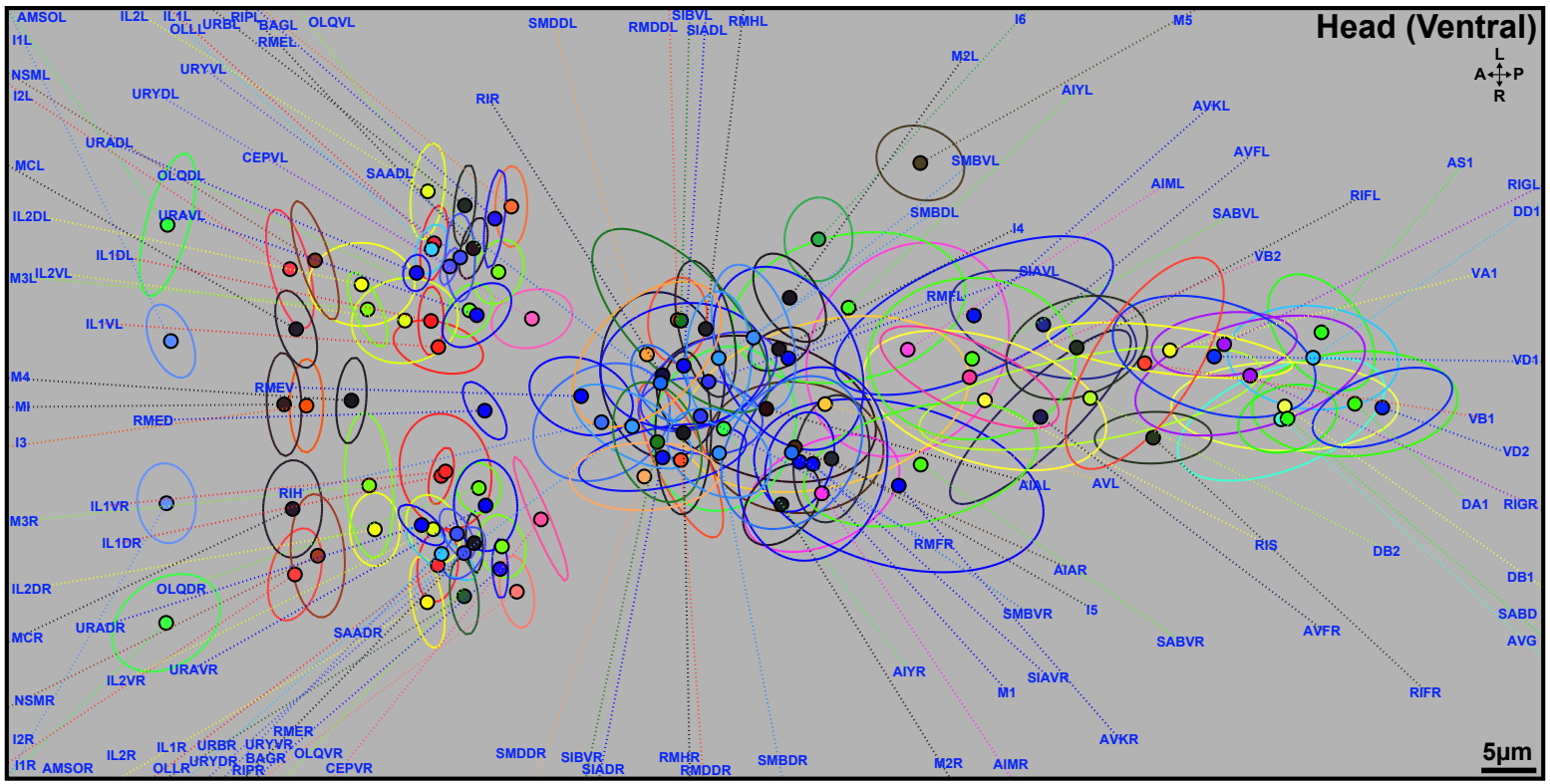
From training worms in $w_i$ space define a latent canonical $z$ space via affine transformations $f_i$.



# Probabilistic Atlas

On canonical space neurons are represented as $z = (z_n)$ with

$$p(z) = \prod_{n=1}^{N} p(z_n) = \prod_{n=1}^{N} \frac{1}{(2\pi\Sigma_n)^{d/2}} e^{-(y_n-\mu_n)^\top \Sigma_n^{-1}(y_n-\mu_n)}.$$



# Probabilistic neural identification

- There are always mistakes. How can we use this setup to model, uncertainty, i.e. a distribution over permutations $P$.

# Probabilistic neural identification

- There are always mistakes. How can we use this setup to model, uncertainty, i.e. a distribution over permutations $P$.
- In canonical space, observed data is $y = Pz$. Induces the posterior

$$p(P|y, \{\mu, \Sigma\}) \propto e^{\langle P, \log L \rangle},$$

with

$$\log L_{n,m} = -\frac{1}{2}(z_n - \mu_m)^\top \Sigma_n^{-1}(z_n - \mu_m).$$

# Probabilistic neural identification

- There are always mistakes. How can we use this setup to model, uncertainty, i.e. a distribution over permutations $P$.
- In canonical space, observed data is $y = Pz$. Induces the posterior

$$p(P|y, \{\mu, \Sigma\}) \propto e^{\langle P, \log L \rangle},$$

with

$$\log L_{n,m} = -\frac{1}{2}(z_n - \mu_m)^\top \Sigma_n^{-1}(z_n - \mu_m).$$

- Deterministic assignment amounts to finding maximum likelihood, i.e. solving

$$\max_P \langle P, \log L \rangle,$$

# Probabilistic neural identification

- There are always mistakes. How can we use this setup to model, uncertainty, i.e. a distribution over permutations $P$.
- In canonical space, observed data is $y = Pz$. Induces the posterior

$$p(P|y, \{\mu, \Sigma\}) \propto e^{\langle P, \log L \rangle},$$

with

$$\log L_{n,m} = -\frac{1}{2}(z_n - \mu_m)^\top \Sigma_n^{-1}(z_n - \mu_m).$$

- Deterministic assignment amounts to finding maximum likelihood, i.e. solving

$$\max_P \langle P, \log L \rangle,$$

- Probabilistically, with *marginal* inference, i.e. the matrix $\rho = E(P)$, the probability of each neuron having a label.

# Variational Inference with Sinkhorn Permanent

- $p(P|L) = \frac{1}{perm(L)} e^{\langle P, \log L \rangle}$, the normalizing constant is the permanent of $L$, $perm(L)$, a #P hard problem (Valiant, 1979)

# Variational Inference with Sinkhorn Permanent

- $p(P|L) = \frac{1}{perm(L)} e^{\langle P, \log L \rangle}$, the normalizing constant is the permanent of $L$, $perm(L)$, a #P hard problem (Valiant, 1979)
- Inference of $\rho = E(P) \iff$ computation of $perm(L)$,

# Variational Inference with Sinkhorn Permanent

- $p(P|L) = \frac{1}{perm(L)} e^{\langle P, \log L \rangle}$, the normalizing constant is the permanent of $L$, $perm(L)$, a #P hard problem (Valiant, 1979)
- Inference of $\rho = E(P) \iff$ computation of $perm(L)$,

$$\log perm(L) = \sup_{\mu \in \mathcal{B}} \langle \log L, \mu \rangle - A^*(\mu)$$

with $\rho = \mu$ achieving the supremum over the Birkhoff polytope $\mathcal{B}$, and $A^*(\mu)$ the entropy dual function (Wainwright and Jordan, 2008)

# Variational Inference with Sinkhorn Permanent

- $p(P|L) = \frac{1}{perm(L)} e^{\langle P, \log L \rangle}$, the normalizing constant is the permanent of $L$, $perm(L)$, a #P hard problem (Valiant, 1979)
- Inference of $\rho = E(P) \iff$ computation of $perm(L)$,

$$\log perm(L) = \sup_{\mu \in \mathcal{B}} \langle \log L, \mu \rangle - A^*(\mu)$$

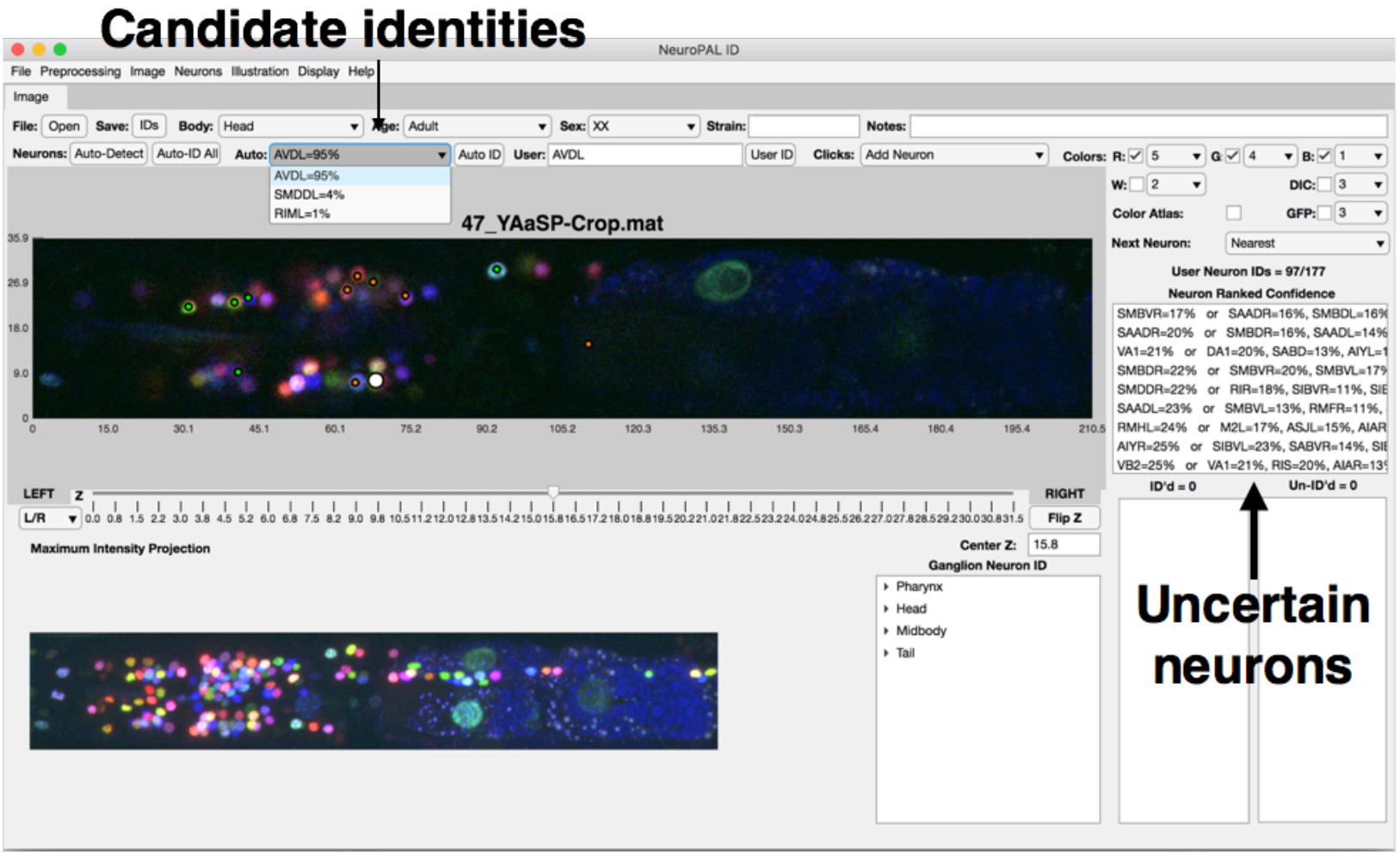
with $\rho = \mu$ achieving the supremum over the Birkhoff polytope $\mathcal{B}$, and $A^*(\mu)$ the entropy dual function (Wainwright and Jordan, 2008)

- Variational Inference: replace $A^*(\mu) \approx \langle \mu, \log \mu \rangle$, then

$$\log perm_S(L) = \sup_{\mu \in \mathcal{B}} \langle \log L, \mu \rangle - \langle \mu, \log \mu \rangle. \quad (1)$$

and $\rho = \mu$ above is obtained by the Sinkhorn algorithm

# The GUI


Candidate identities
NeuroPAL ID
File Preprocessing Image Neurons Illustration Display Help
Image
File: Open Save: IDs Body: Head Age: Adult Sex: XX Strain: Notes:
Neurons: Auto-Detect Auto-ID All Auto: AVDL=95% Auto ID User: AVDL User ID Clicks: Add Neuron
Colors: R: 5 G: 4 B: 1
W: 2 DIC: 3
Color Atlas: GFP: 3
Next Neuron: Nearest
AVDL=95%
SMDDL=4%
RIML=1%
47_YAaSP-Crop.mat
User Neuron IDs = 97/177
Neuron Ranked Confidence
SMBVR=17% or SAADR=16%, SMBDL=16%
SAADR=20% or SMBDR=16%, SAADL=14%
VA1=21% or DA1=20%, SABD=13%, AIYL=1
SMBDR=22% or SMBVR=20%, SMBVL=17%
SMDDR=22% or RIR=18%, SIBVR=11%, SIB
SAADL=23% or SMBVL=13%, RMFR=11%,
RMHL=24% or M2L=17%, ASJL=15%, AIAR
AIYR=25% or SIBVL=23%, SABVR=14%, SIB
VB2=25% or VA1=21%, RIS=20%, AIAR=13%
ID'd = 0
Un-ID'd = 0
LEFT Z RIGHT
L/R
Flip Z
Maximum Intensity Projection
Center Z: 15.8
Ganglion Neuron ID
Pharynx
Head
Midbody
Tail
Uncertain neurons

# Some machine learning methods arising from the above

# Sinkhorn Networks

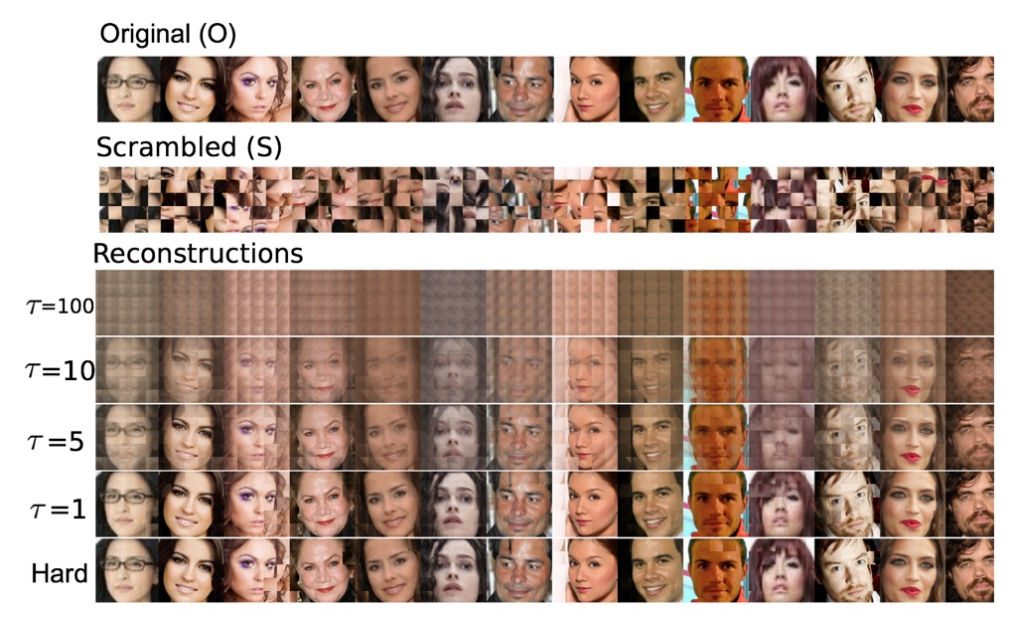
Artificial neural networks are useful for predicting discrete data, a label. What about permutations?

Problem Statement: How to decode scrambled objects $\tilde{X}$ into non-scrambled $X$ (e.g. jigsaw puzzle).

- Data are pairs $(X_i, \tilde{X}_i)$ where $\tilde{X}_i$ are constructed by permuting pieces of $X_i$:

$$X_i = P^{-1}_{\theta,\tilde{X}_i} \tilde{X}_i, \quad P_{\theta,\tilde{X}} \approx M(g(\tilde{X}, \theta)).$$

- To train, replace $S(g(\tilde{X}, \theta)/\tau) \approx M(g(\tilde{X}, \theta))$.



# Deep Generative Models

- One of the most exciting applications of Deep Learning is generating unseen data from a training dataset. e.g. GANS, VAE.

# Deep Generative Models

- One of the most exciting applications of Deep Learning is generating unseen data from a training dataset. e.g. GANS, VAE.

- In any case, vector $z$ is sampled from a noise distribution and passed through a neural network $g_\theta(z)$

# Deep Generative Models

- One of the most exciting applications of Deep Learning is generating unseen data from a training dataset. e.g. GANS, VAE.

- In any case, vector $z$ is sampled from a noise distribution and passed through a neural network $g_\theta(z)$
- Instead: to generate an object first generate a set of pieces $p$ and then a permutation $\pi$ of those pieces that will assemble the object $o$.

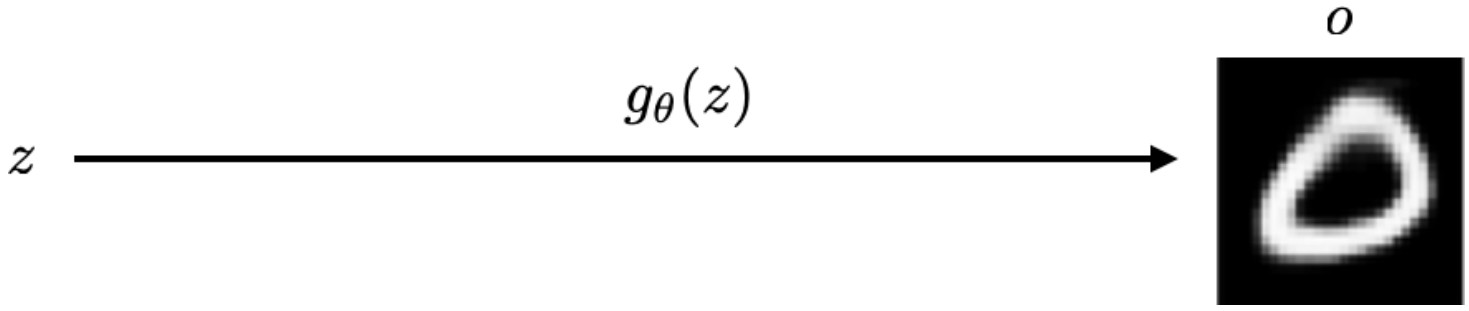
# Deep Generative Models

- One of the most exciting applications of Deep Learning is generating unseen data from a training dataset. e.g. GANS, VAE.

- In any case, vector $z$ is sampled from a noise distribution and passed through a neural network $g_\theta(z)$
- Instead: to generate an object first generate a set of pieces $p$ and then a permutation $\pi$ of those pieces that will assemble the object $o$.
- Like going to Home Center Sodimac. Get the pieces and the instructions manual.

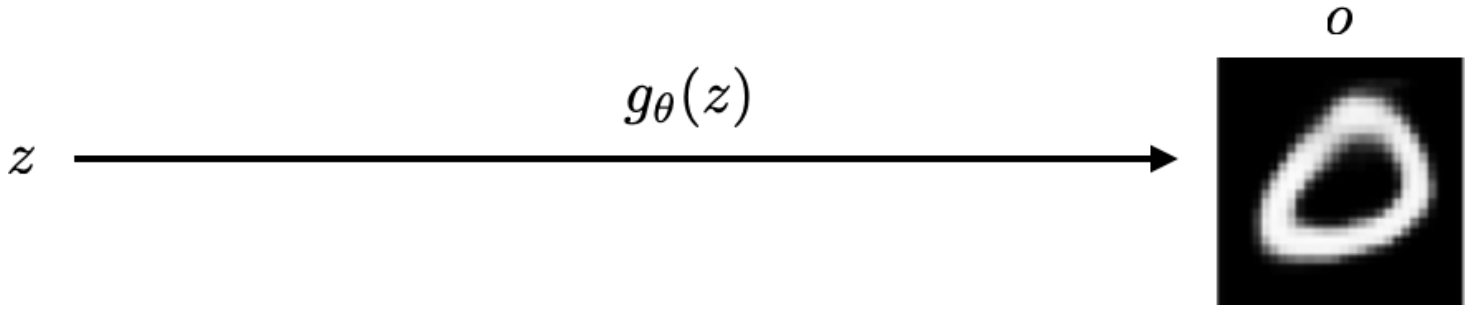
# New Deep Generative Methodology

## Usual generative modeling perspective

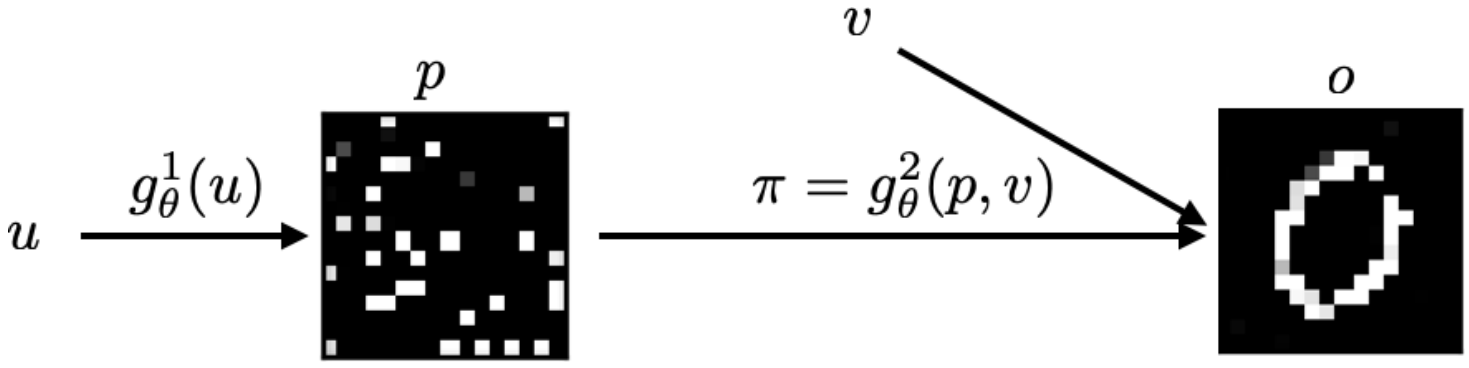


# New Deep Generative Methodology

## Usual generative modeling perspective

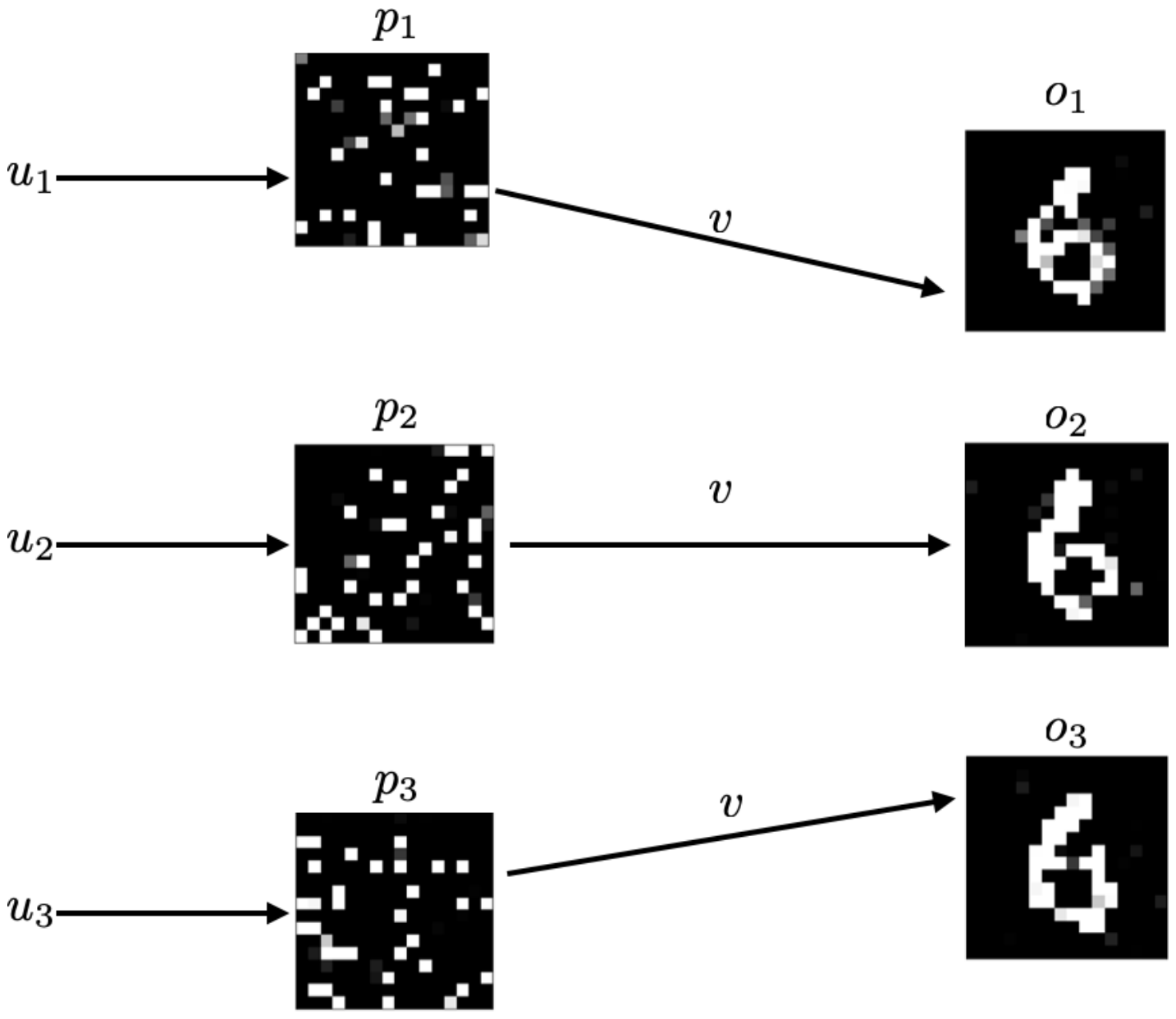



## New generative modeling perspective

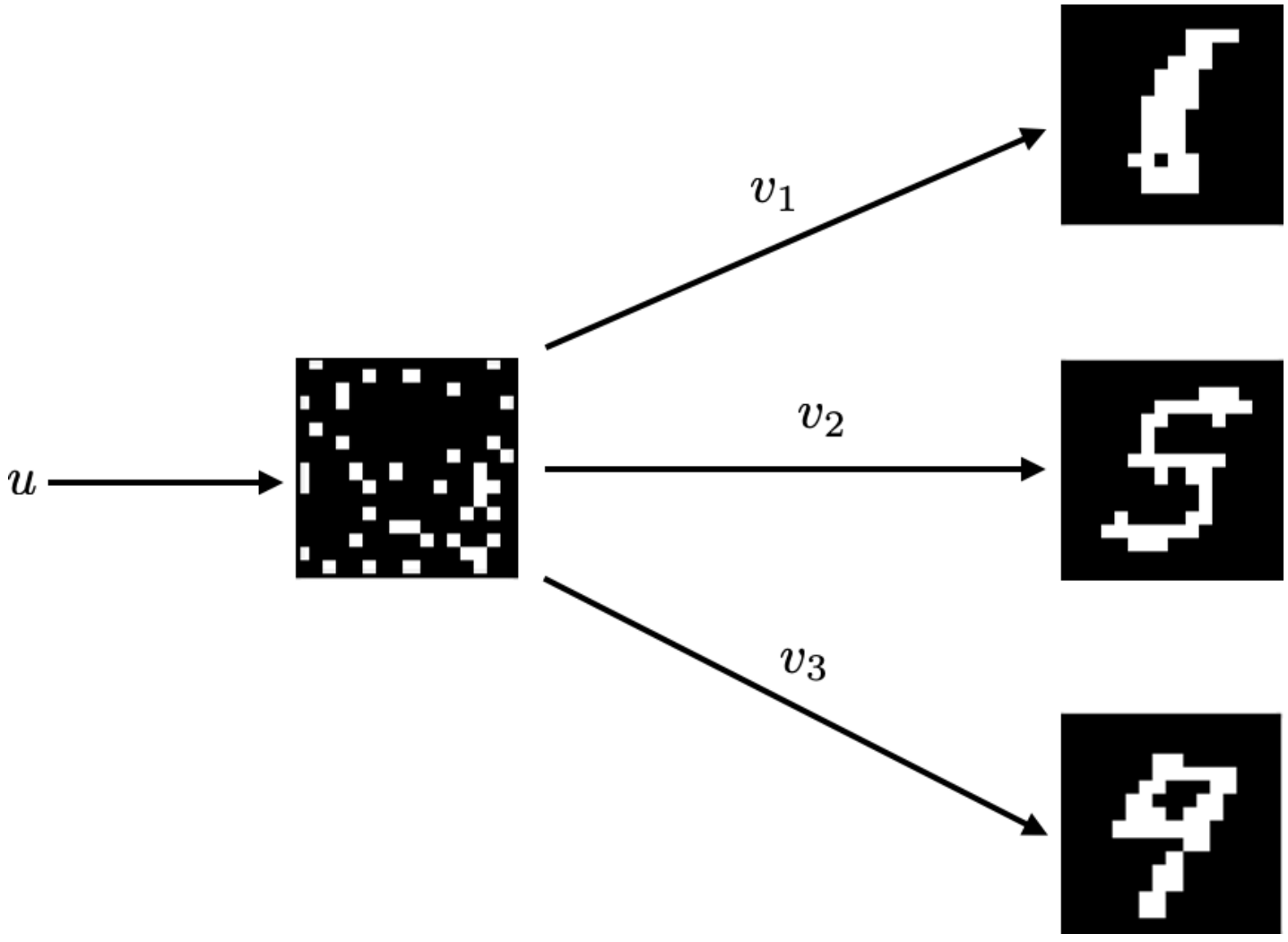



$\pi$ is a permutation (approximated with Sinkhorn algorithm).

# Same number from different pieces



# Different numbers from same pieces

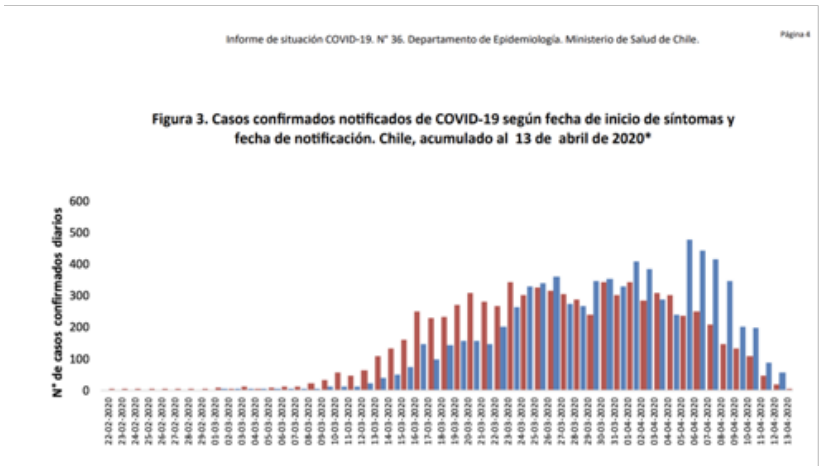


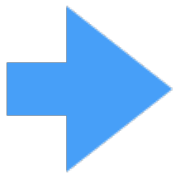
# Modelamiento estadístico de COVID-19

- Incluso si todos los datos se hacen públicos no nos dan la imagen completa. Los registros son 'ingenuos'.
- **Se puede reconstruir la verdadera situación en escalas espacio-temporales finas, dados los registros?**
- Además de incompletos, los registros están sesgados y retrasados. Los sesgos y retrasos pueden depender del tiempo y el espacio.
- Las predicciones y la toma de decisiones son sensibles a estos números
- Los tests pueden convertirse en un recurso limitado. Se puede "extrapolar" a partir de situaciones donde hay mejor testeo?
- **Cómo se integran distintos tipos de datos (casos, UCI, muertes, historia, encuesta MOVID-19 de vigilancia sindrómica, tests de otras enfermedades, movilidad, búsquedas de internet)** para reconstruir la verdadera situación?
- Acá: **Modelos bayesianos jerárquicos**

# Los datos de @perez y la fecha de inicio de síntomas

Sofisticados métodos de inteligencia artificial permiten convertir los histogramas de los informes de situación epidemiológica en una base de datos.



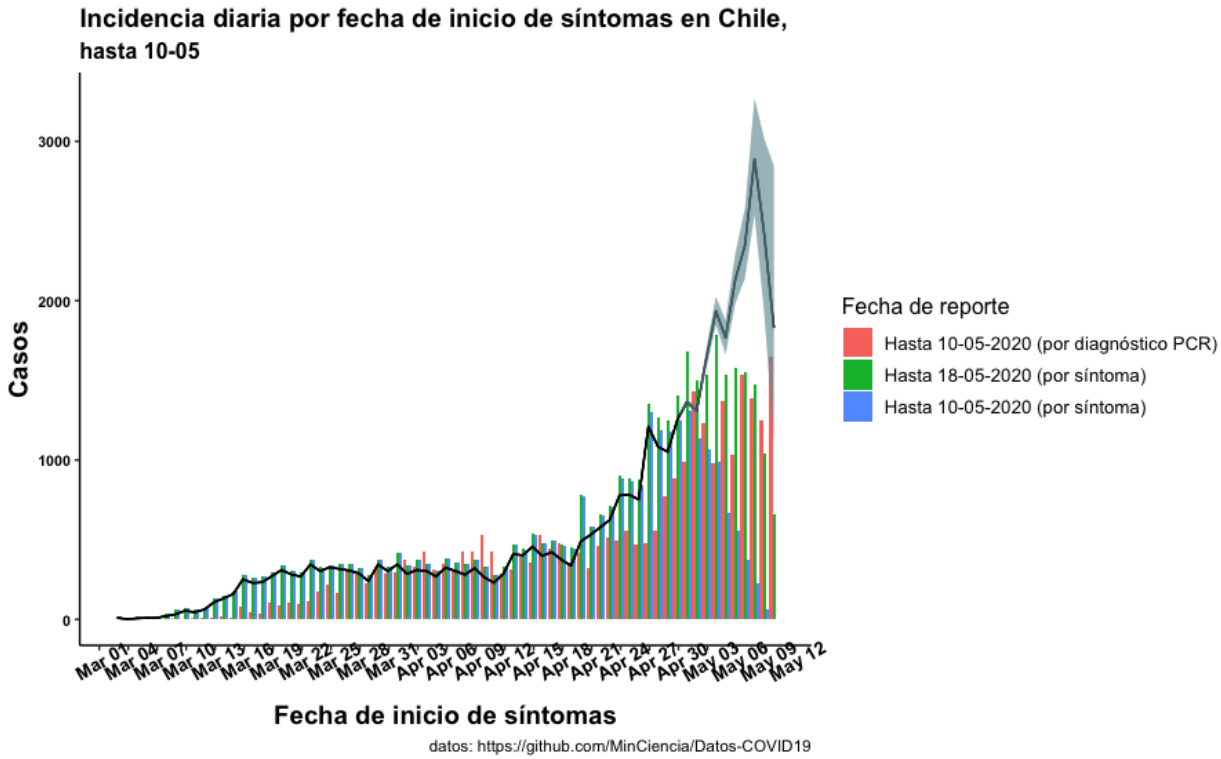



| | Reportado el 13/04 FIG. 2 | | Reportado el 14/04 FIG. 2 | | Reportado el 15/04 FIG. 2 | |
|---|---|---|---|---|---|---|
| Fecha | inicio síntomas | notificacion | inicio síntomas | notificacion | inicio síntomas | notificacion |
| 3/15/2020 | 159 | 46 | 159 | 51 | 158 | 53 |
| 3/16/2020 | 251 | 71 | 252 | 76 | 255 | 78 |
| 3/17/2020 | 229 | 144 | 231 | 145 | 232 | 147 |
| 3/18/2020 | 233 | 95 | 235 | 100 | 235 | 102 |
| 3/19/2020 | 271 | 140 | 272 | 142 | 274 | 143 |
| 3/20/2020 | 310 | 155 | 318 | 156 | 315 | 157 |
| 3/21/2020 | 282 | 155 | 284 | 158 | 280 | 157 |
| 3/22/2020 | 269 | 144 | 272 | 145 | 274 | 147 |
| 3/23/2020 | 345 | 200 | 349 | 205 | 357 | 206 |
| 3/24/2020 | 303 | 263 | 304 | 267 | 304 | 271 |
| 3/25/2020 | 327 | 329 | 324 | 333 | 329 | 334 |
| 3/26/2020 | 316 | 339 | 318 | 347 | 315 | 343 |
| 3/27/2020 | 307 | 361 | 314 | 365 | 318 | 368 |
| 3/28/2020 | 289 | 274 | 290 | 281 | 294 | 274 |
| 3/29/2020 | 240 | 267 | 248 | 267 | 253 | 268 |
| 3/30/2020 | 345 | 346 | 356 | 355 | 364 | 354 |
| 3/31/2020 | 303 | 354 | 311 | 355 | 311 | 347 |
| 4/1/2020 | 345 | 329 | 353 | 337 | 364 | 334 |
| 4/2/2020 | 285 | 408 | 297 | 423 | 301 | 426 |
| 4/3/2020 | 310 | 385 | 314 | 392 | 329 | 392 |
| 4/4/2020 | 303 | 287 | 304 | 298 | 318 | 298 |
| 4/5/2020 | 237 | 238 | 259 | 236 | 269 | 240 |
| 4/6/2020 | 251 | 479 | 287 | 480 | 318 | 482 |
| 4/7/2020 | 208 | 444 | 238 | 462 | 262 | 468 |
| 4/8/2020 | 146 | 417 | 186 | 437 | 225 | 447 |

**Muy importante: cuándo empezaron los síntomas de cada caso es más importante que los casos tabulados por su fecha de reporte.**

# Ahorasticar (nowcasting)

Para entender la transmisión se debe saber cuánta gente se está enfermando en cada momento. Los casos aparecen sólo después del diagnóstico. NobBS: Nowcasting by bayesian smoothing (McGough et al, 2020)




Línea y bandas de confianza basadas en NobBS. Advertencia: este modelo no ha sido evaluado/calibrado para este contexto.

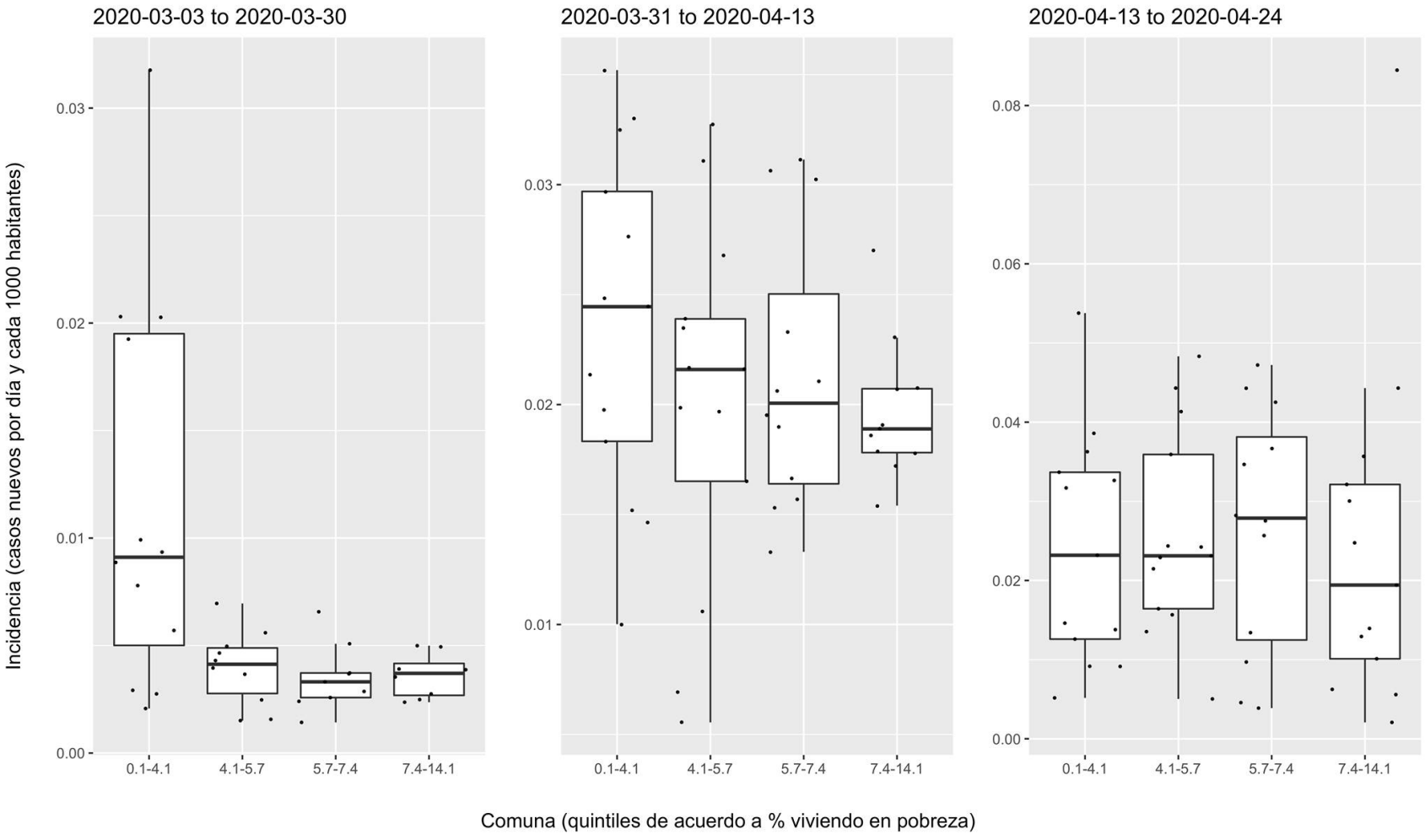
# Algunas ecuaciones

- $Y_t = \sum_d Y_{t,d}$, $Y_{t,d}$= casos nuevos el día $t$ reportados con un retraso de $d$ días.
- Modelo bayesiano

$$Y_{t,d} = Poisson(\lambda_{t,d}), \log(\lambda_{t,d}) = \alpha_t + \log(\beta_d)$$

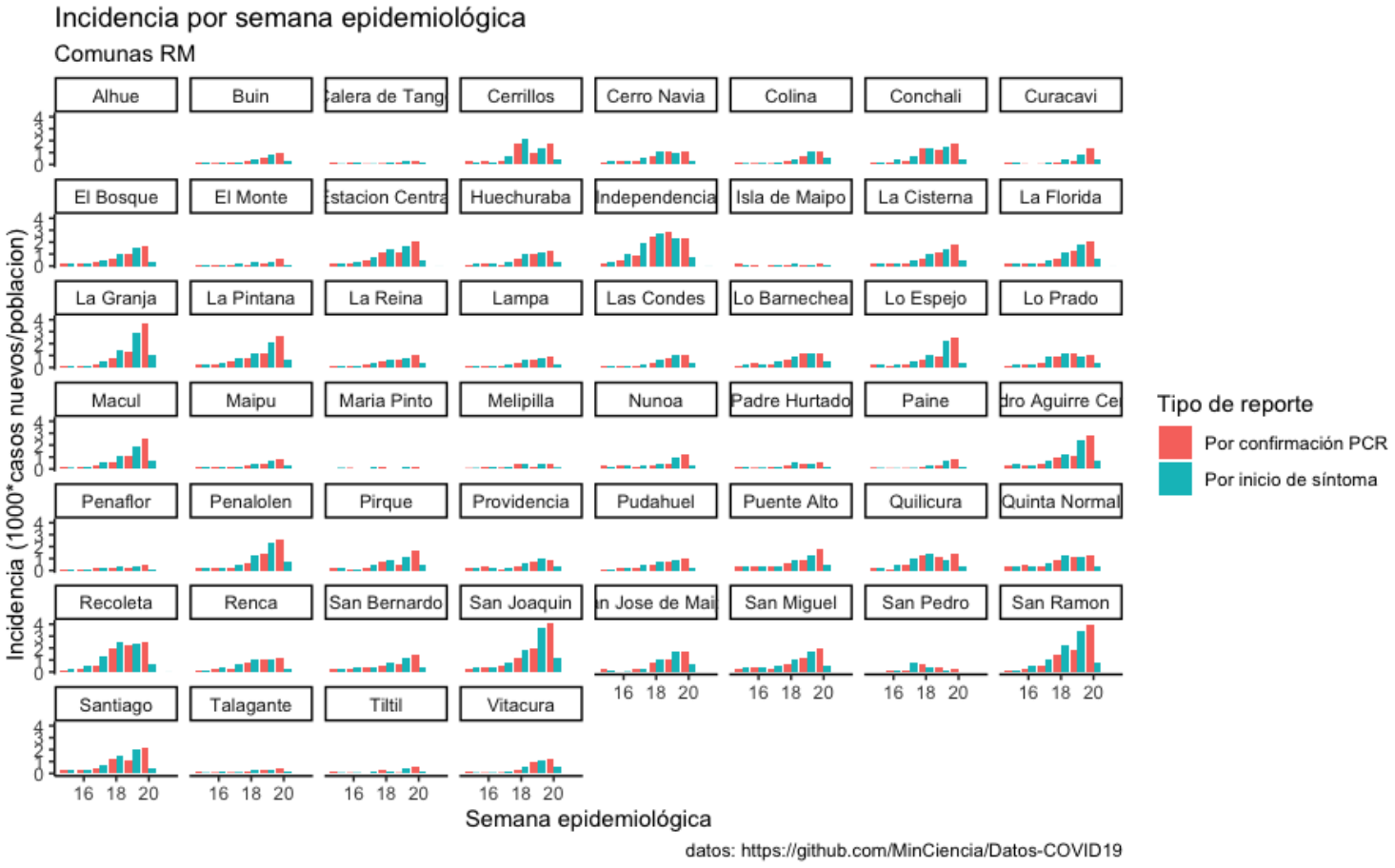
- Para ahorasticar se requiere inferir $\alpha_t$, el que se asume seguir un paseo aleatorio o movimiento browniano como *prior*, $\alpha_t \sim \mathcal{N}(\alpha_{t-1}, \tau^2)$.
- $\beta_d$ es la distribución de retraso en reportar, con Dirichlet prior.

# Los procesos espaciales pueden ser clave



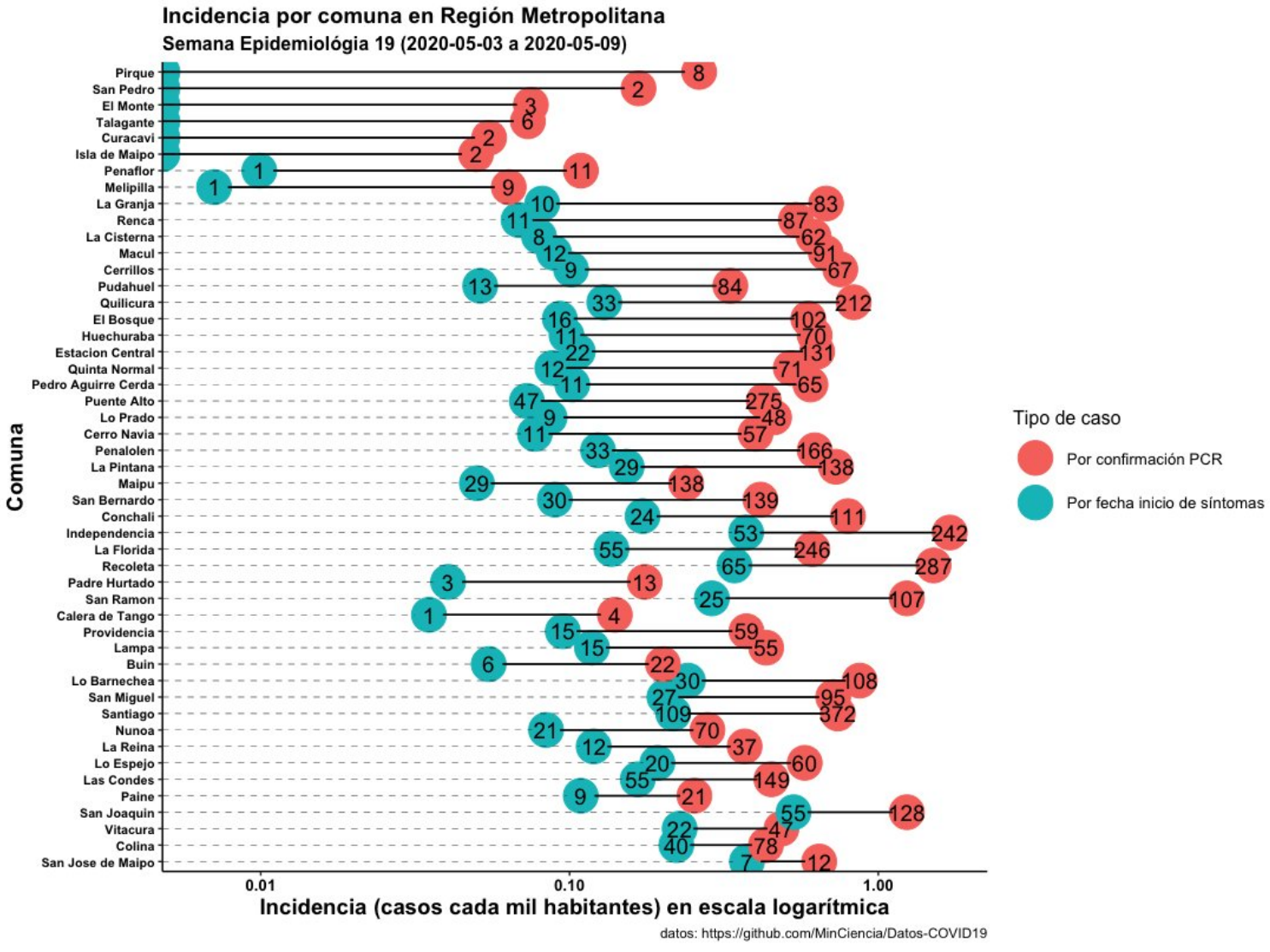

El control de la epidemia puede tener que ver con entender cambios en dinámicas más finas

# Se puede hacer a escala espacial más fina?



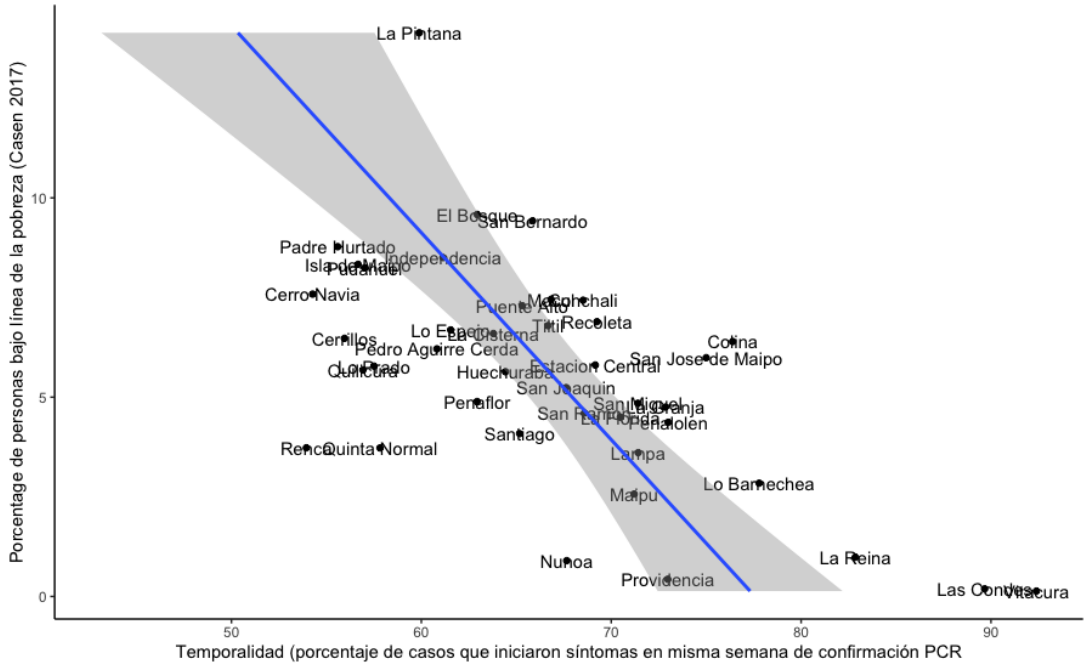

Advertencia: Santiago no es Chile

# Se puede hacer a escala espacial más fina?




En algunas comunas los casos parecen reportarse más tarde

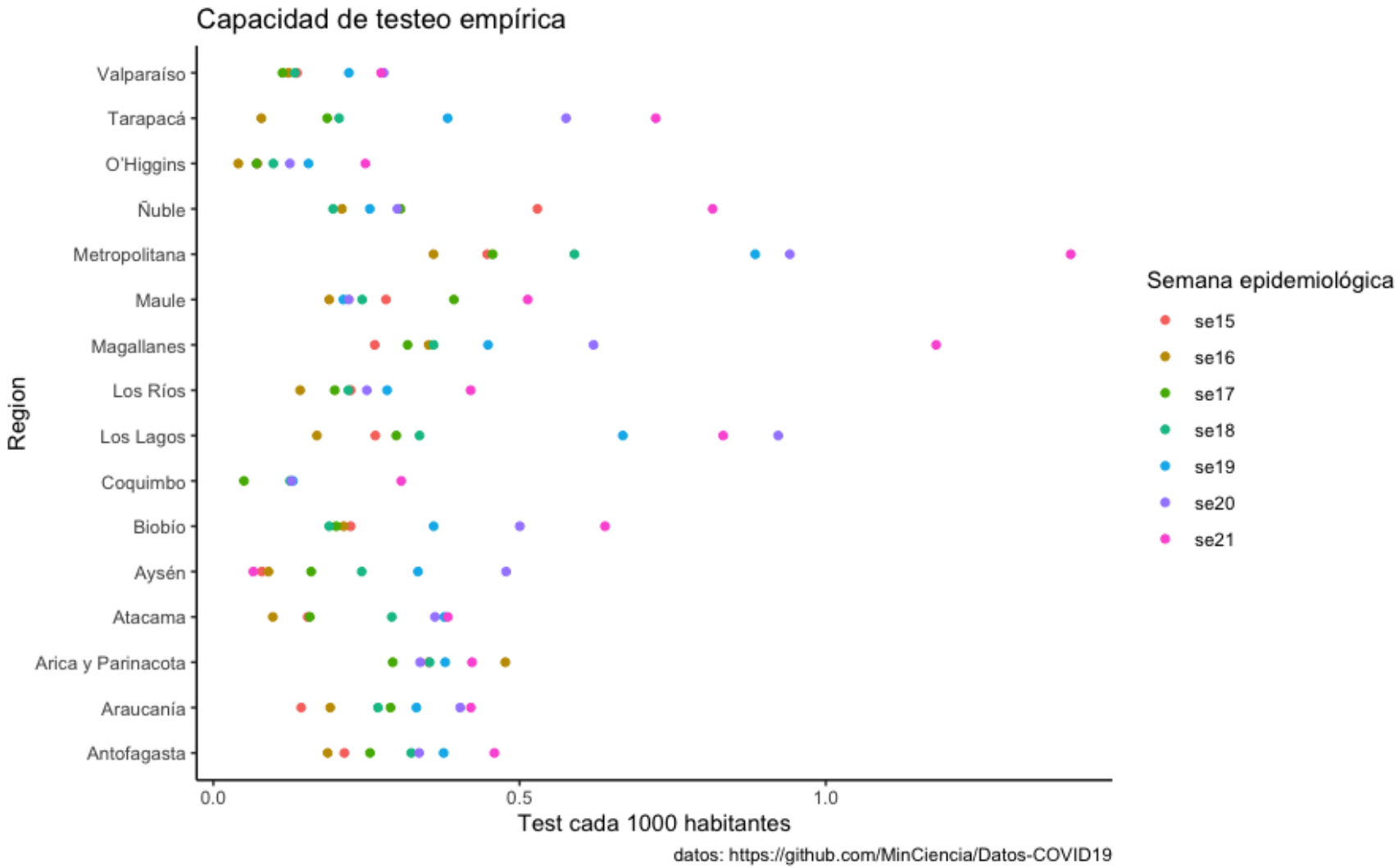
# La dependencia de los retrasos en reportar en otras variables




$$R \approx -0.45$$

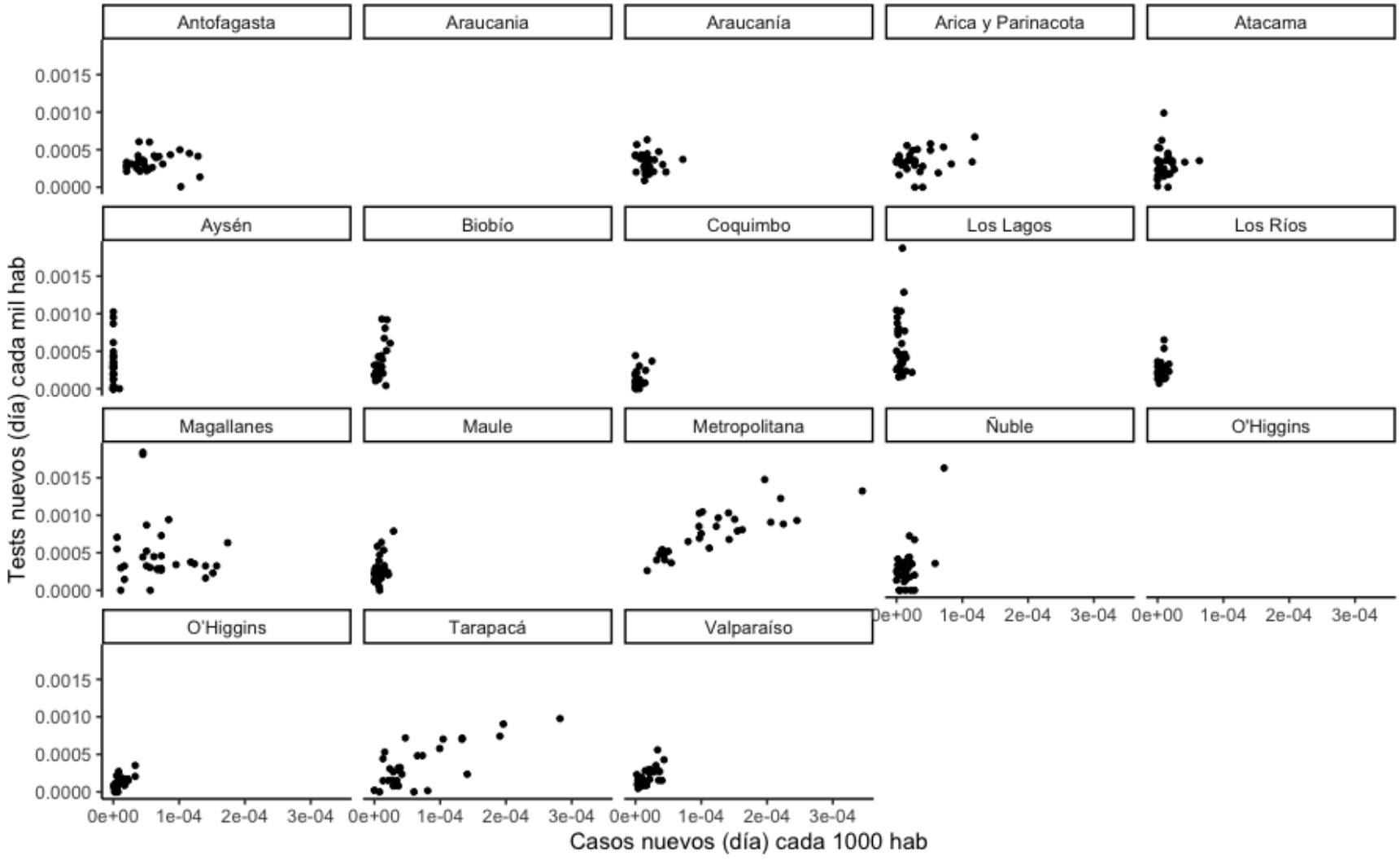
- Una correlación negativa (quizás espuria) entre pobreza y confirmación de casos oportuna.
- **hipótesis**: Retrasos diferenciales en responder podrían evidenciar (ser un proxy) de sub-reporte (Stoner et al, JASA 2019); si los resultados tardan mucho puede significar que otra gente no está accediendo a tests.

# La realidad del testeo I




La disponibilidad de tests también es una función del espacio

# La realidad del testeo II




Tasas de positividad podrían también indicar problemas

# Algunas ecuaciones II

Cómo incluir subreporte en espacio-tiempo?

- $Y_{t,s} = \sum_d Y_{t,s,d}$, $Y_{t,s,d}$= casos nuevos el día $t$ reportados con un retraso de $d$ días. En locacion $s$ (comuna, región, etc). En realidad hay $Z_{t,s,d}$ casos que son reportados con probabilidad $\pi_{s,t}$.
- Modelo bayesiano

$$Y_{t,d,s}|Z_{t,d,s} \sim Binomial(Z_{t,s,d}, \pi_{s,t}),$$

$$Z_{t,s,d} \sim Poisson(\lambda_{t,s,d}), \log(\lambda_{t,s,d}) = \alpha_{t,s} + \log(\beta_{s,d})$$

- $\pi_{s,t} \sim f(x_{s,t})$ representa la probabilidad reportar y depende de otros regresores (retraso en responder, tasa de positividad, etc). "Pedir prestado" la fuerza de testeo donde se testea mejor.
- Para ahorasticar se refiere inferir $\alpha_t = (\alpha_{t,s})_s$, el que se asume seguir un paseo aleatorio o movimiento browniano como *prior*, $\alpha_t \sim \mathcal{N}(\alpha_{t-1}, \Sigma_s)$ donde $\Sigma_s$ representa la estructura de correlaciones espaciales.

# Más allá de este modelo

- Cómo agregar otros tipos de información: muertes, muertes por exceso, vigilancia sindrómica, MOVID-19, vigilancia epidemiológica de otros viruses, búsquedas de internet, etc.
- Modelos más complejo o **meta-análisis de varios modelos pequeños?** (boosting)

# Palabras finales

- Gracias a Liam Paninski, EJ Chichilnisky, Sasi Madugula, Jasper Snoek, Jonathan Niles-Weed, Nishal Shah, Amin Eviatar Yemini, Erdem Varol, Amin Nejatbakhsh.
- Gracias a Pamela Martínez, Joaquín Fontbona, Alejandro Maass, Mauricio Santillana, Fred Lu, Oliver Stoner, Pablo Martínez, Orlando Rivera, Phyllis Ju, Jorge Pérez, Cristóbal Cuadrado, Gonzalo Contador, Equipo Github Ministerio de Ciencia, etc
- Es posible ayudar!