Machine Learning Methods for Neural Data Analysis

Bayesian decoders for neural spike trains

Scott Linderman

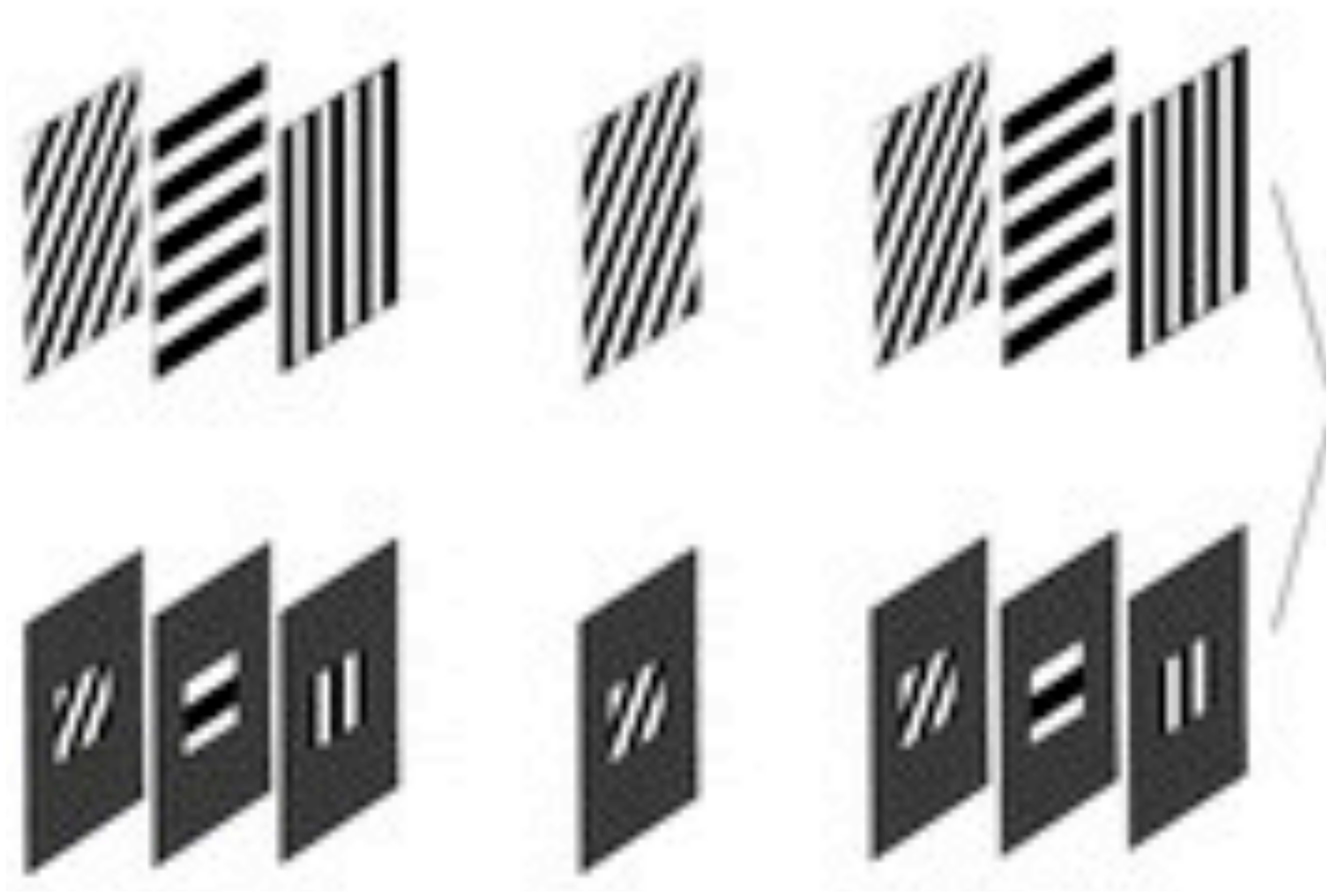
STATS 220/320 (NBIO220, CS339N).

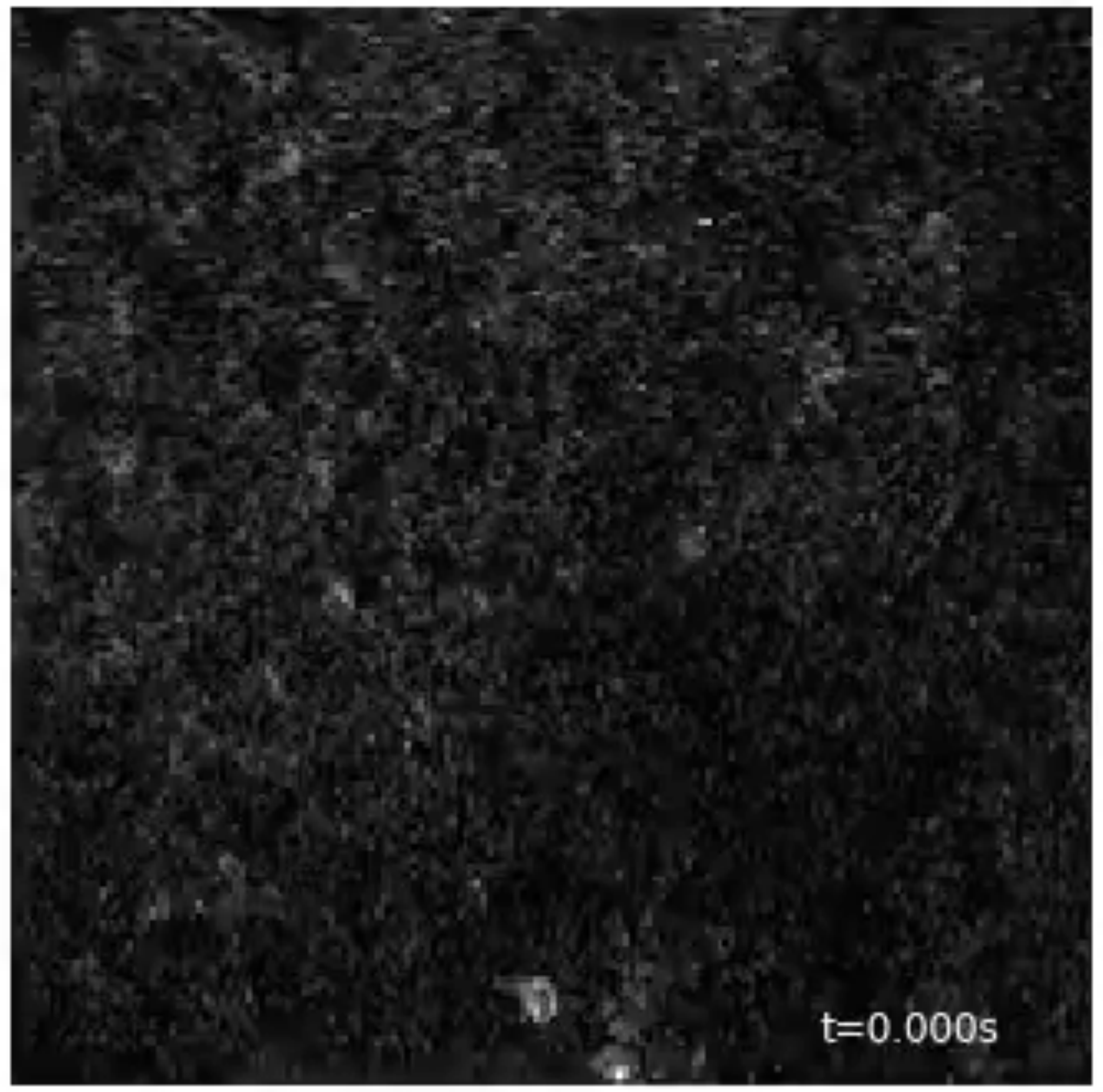
Agenda
Decoding neural spike trains

- Bayesian decoders
  - A straw man model, just for illustration
  - An aside on the multivariate Gaussian distribution
  - Improving upon the basic model

Big picture

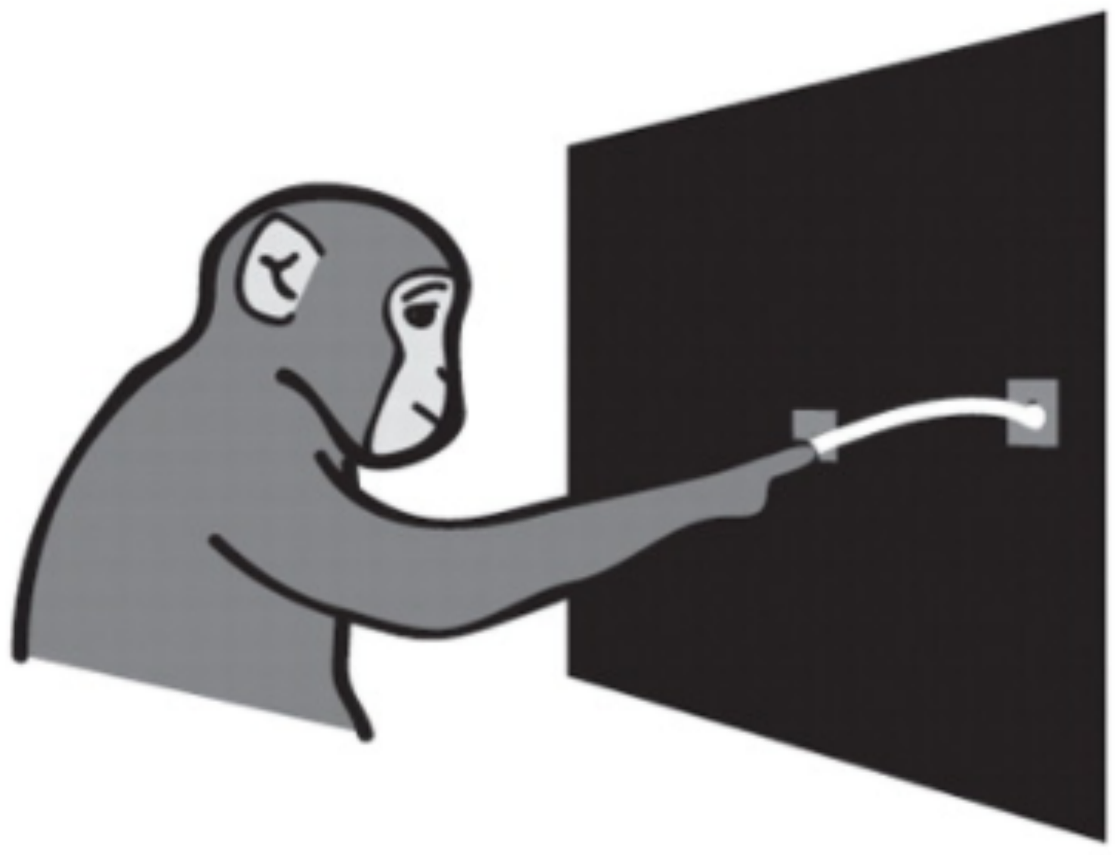
Stimulus —*encoding*→ Neural Recording —*decoding*→ Behavioral Video



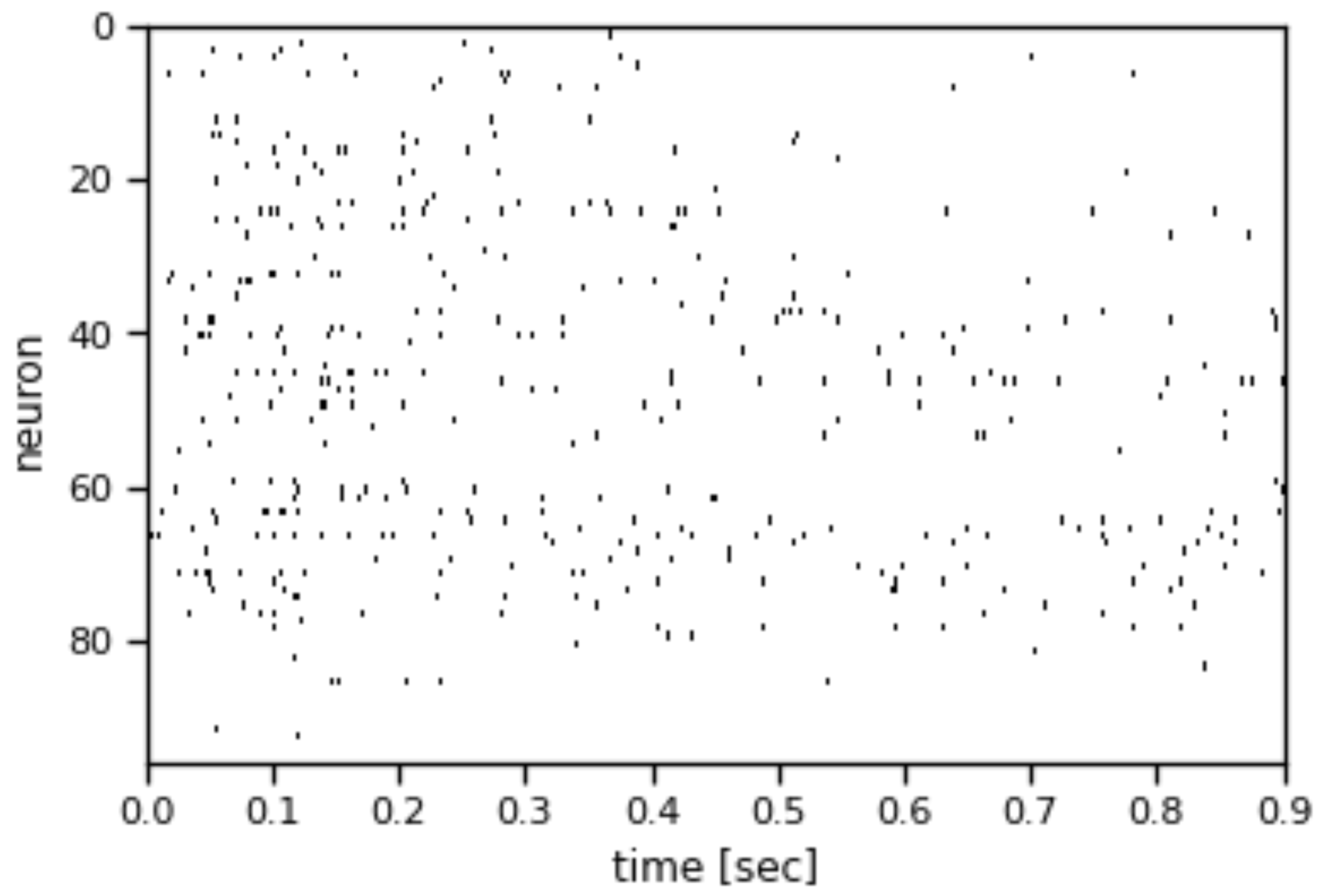


To a statistician, it's all regression!

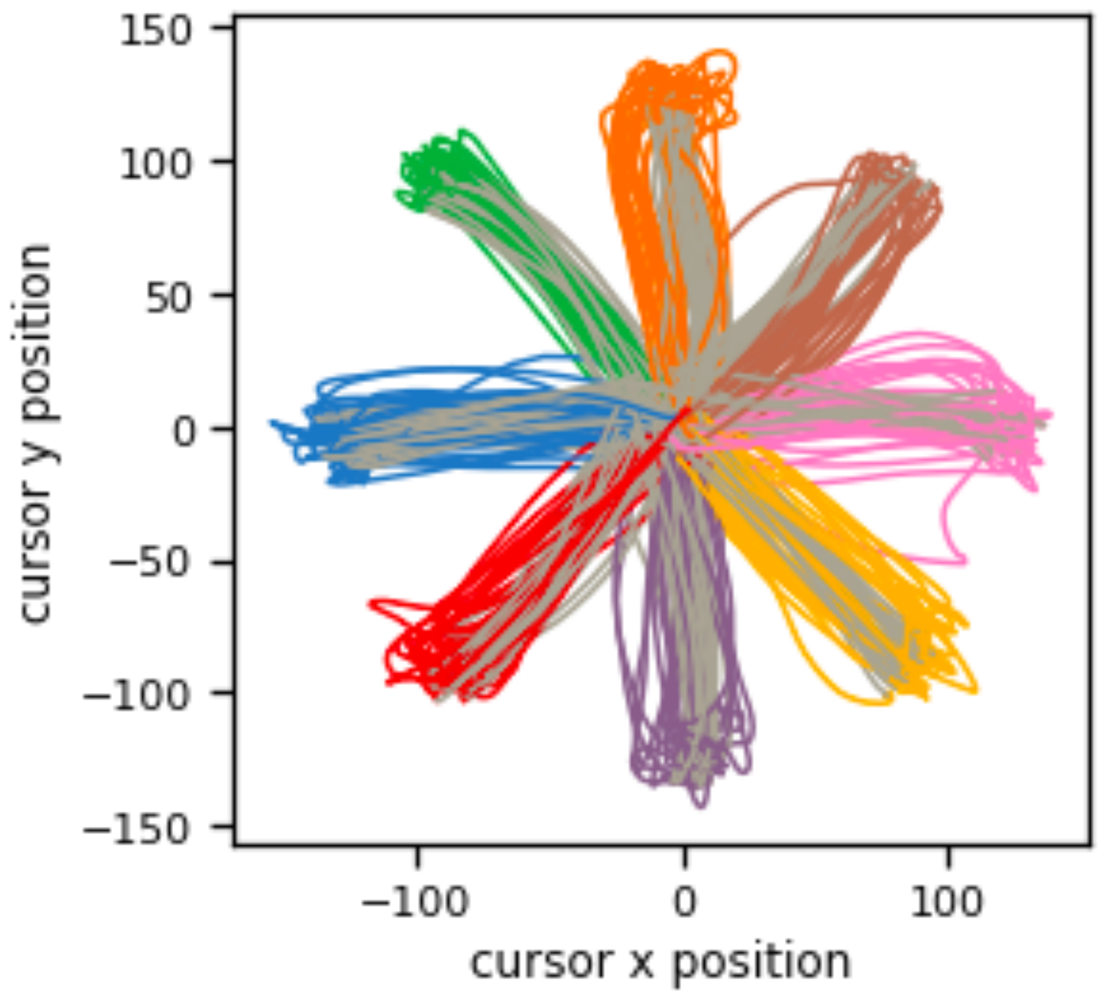
Decoding movement from recordings in motor cortex



Y



X



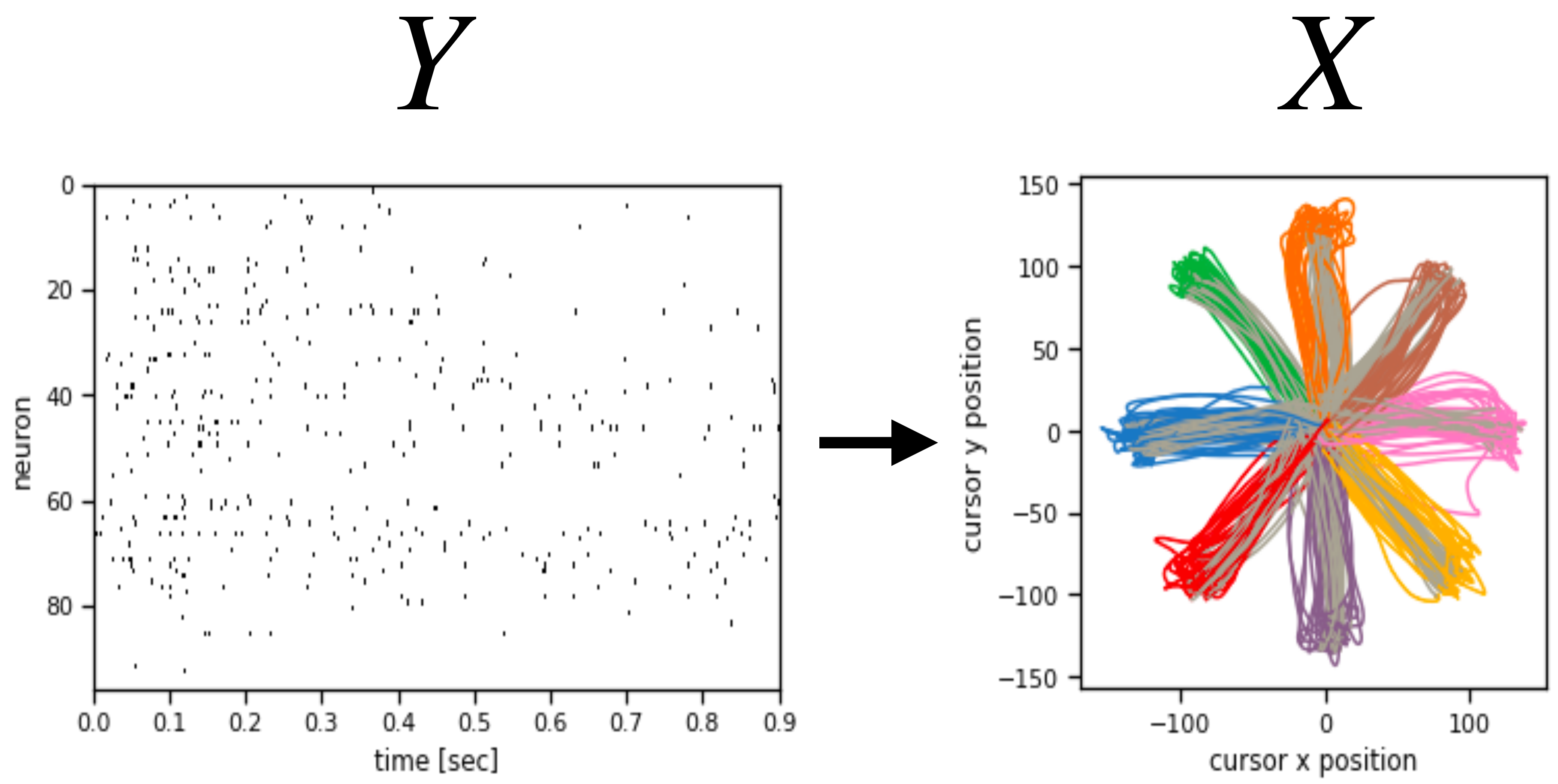
GOAL: estimate $p(X \mid Y)$

Shenoy Lab (Stanford)

Decoding movement from neural spike trains

Brainstorming

- How would you approach this problem?



Decoding movement from neural spike trains

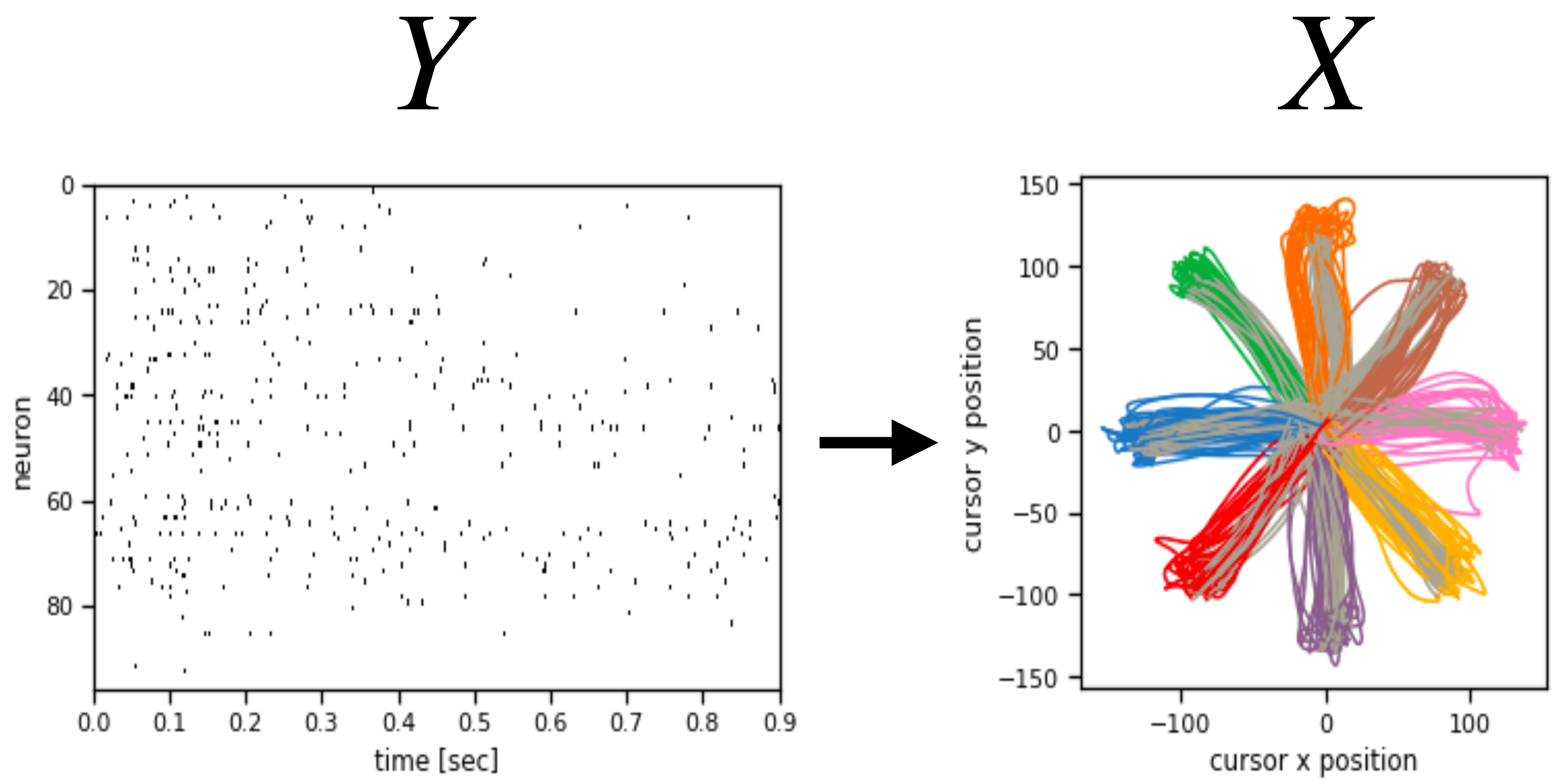
Brainstorming

- It’s just a regression problem… let’s use the same techniques (GLMs, CNNs, etc) that we used for encoders. I’d call these “direct” decoders, and it’s a perfectly fine approach if you have the data.

- Alternatively, suppose we know something about the prior distribution of movement, $p(X)$. E.g. current position and velocity determine next position.

- Moreover, suppose we know something about what the neurons encode. E.g. suppose the neurons encode current velocity.

- Can we use that knowledge to inform our decoder?



Decoding movement from neural spike trains

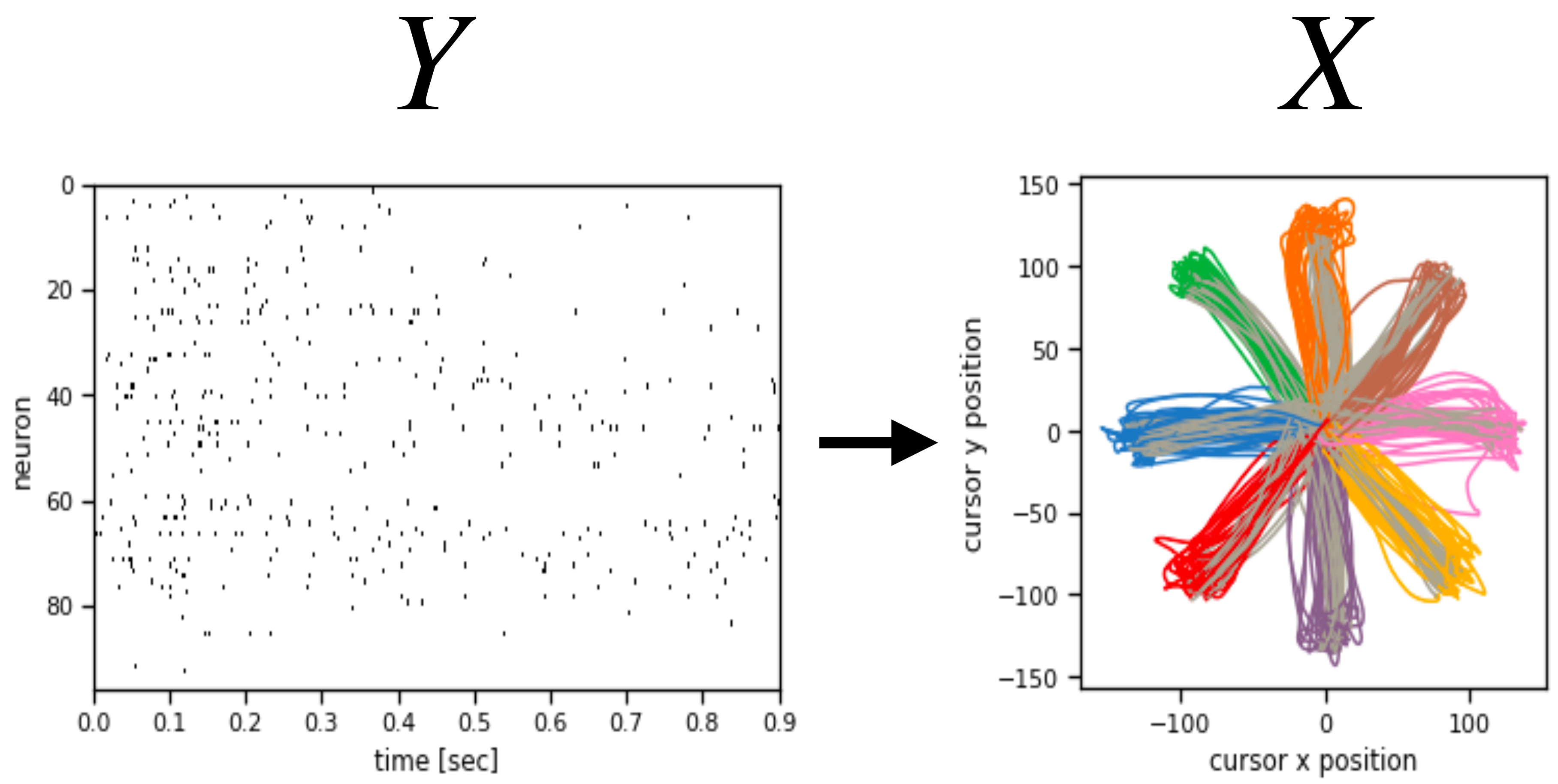
Bayesian decoders

- Bayes' Rule tells us how to combine a **prior** $p(X)$and a **likelihood** $p(Y \mid X)$ to obtain a **posterior**,

$$p(X \mid Y) = \frac{p(Y \mid X)p(X)}{p(Y)}$$

$$\propto p(Y \mid X)p(X)$$

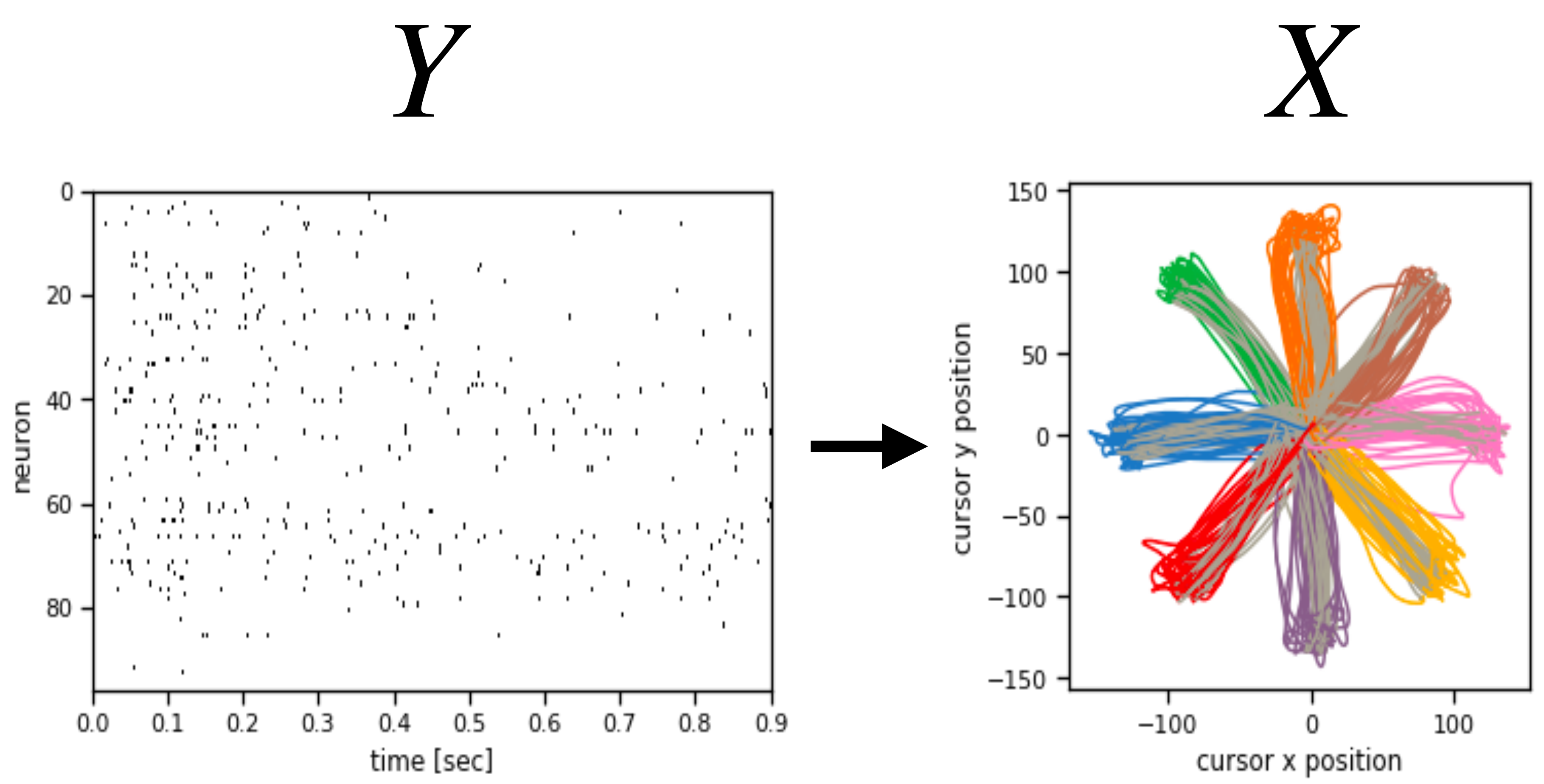
- Here, the likelihood is the **encoder** and the posterior is the **decoder**.



Decoding movement from neural spike trains

A very simple model

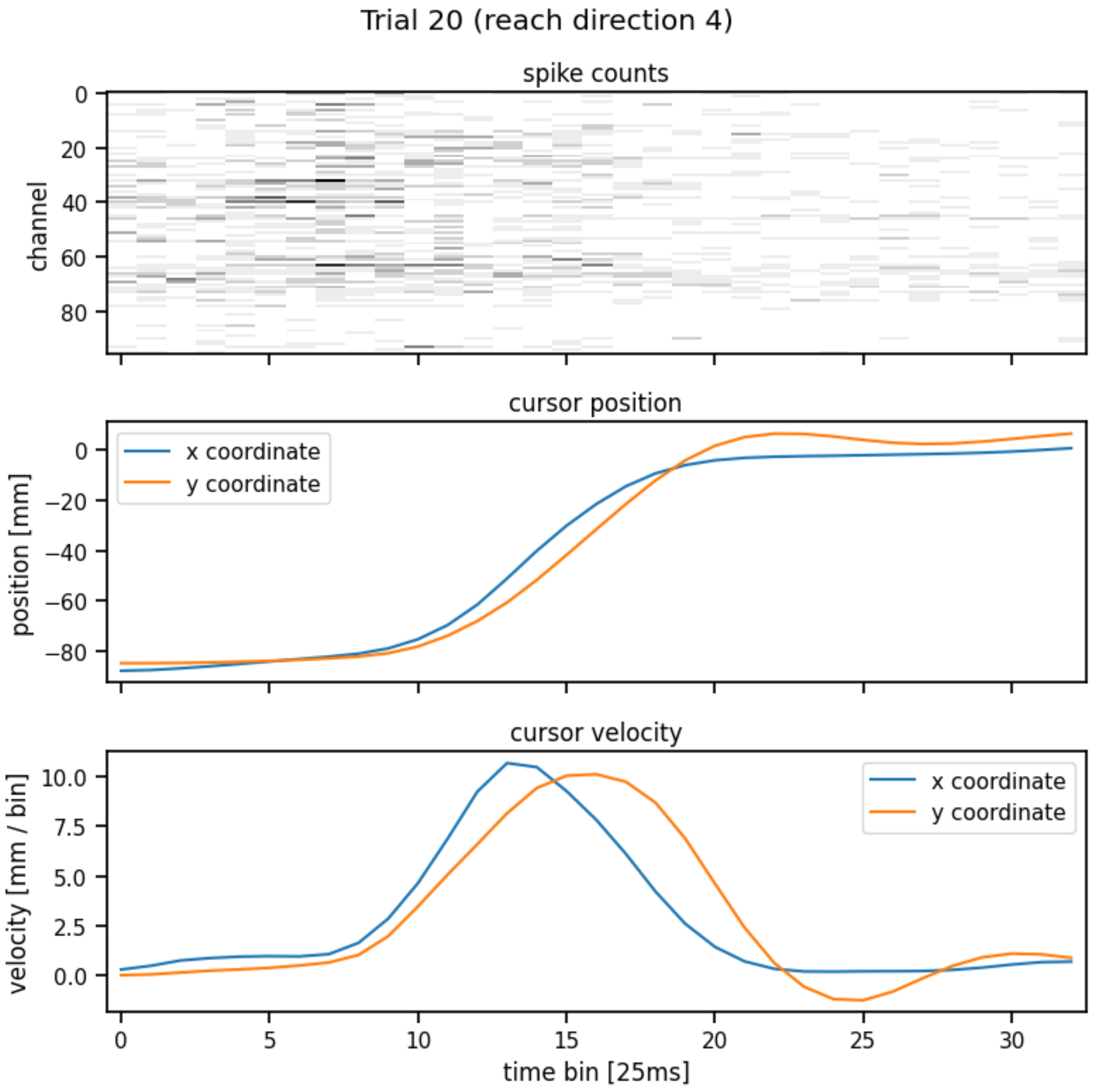
- Let $y_t \in \mathbb{N}^N$ denote the **spike counts** of N neurons at time t.
- Let $x_t \in \mathbb{R}^2$ denote the **cursor velocity** at time t.



Note: we will model the velocity, but here we show the position (integrated velocity) for illustration.

Decoding movement from neural spike trains

A more accurate depiction of the data

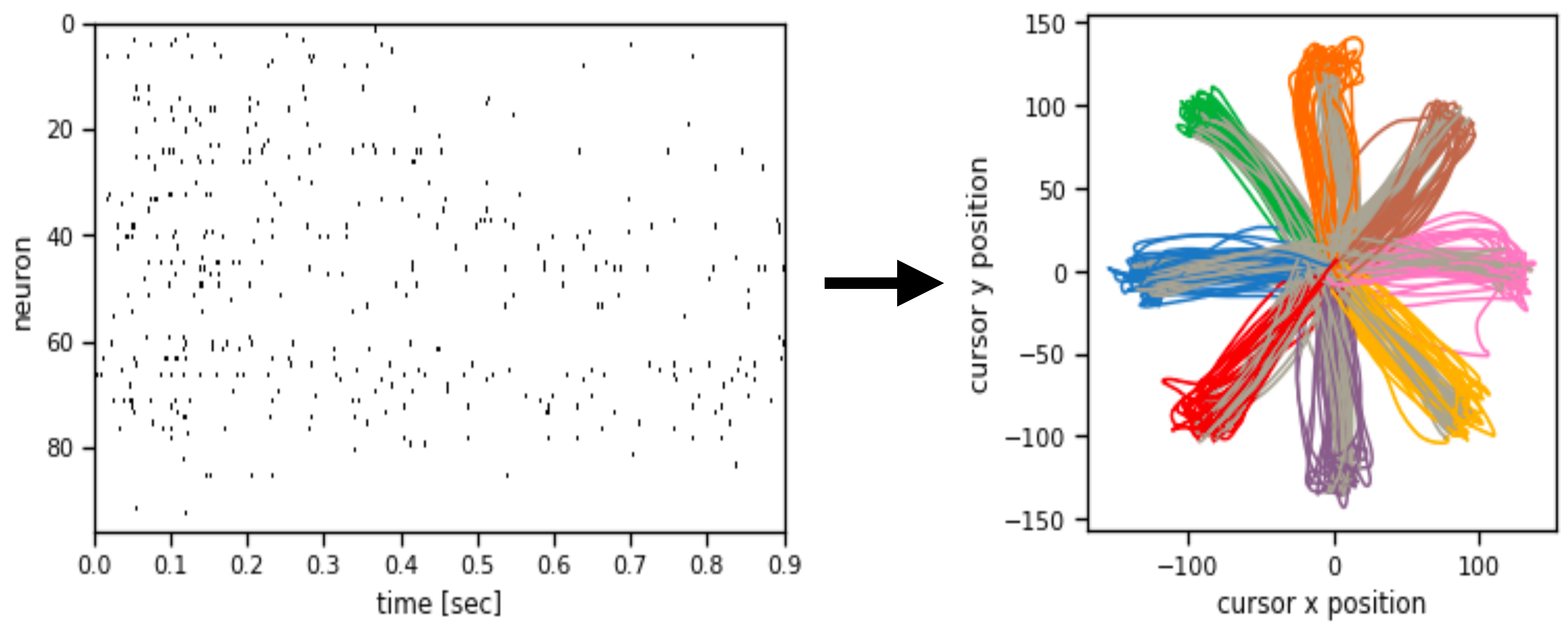


Decoding movement from neural spike trains

A simple example

Consider the following likelihood (i.e. encoder)…

Y X

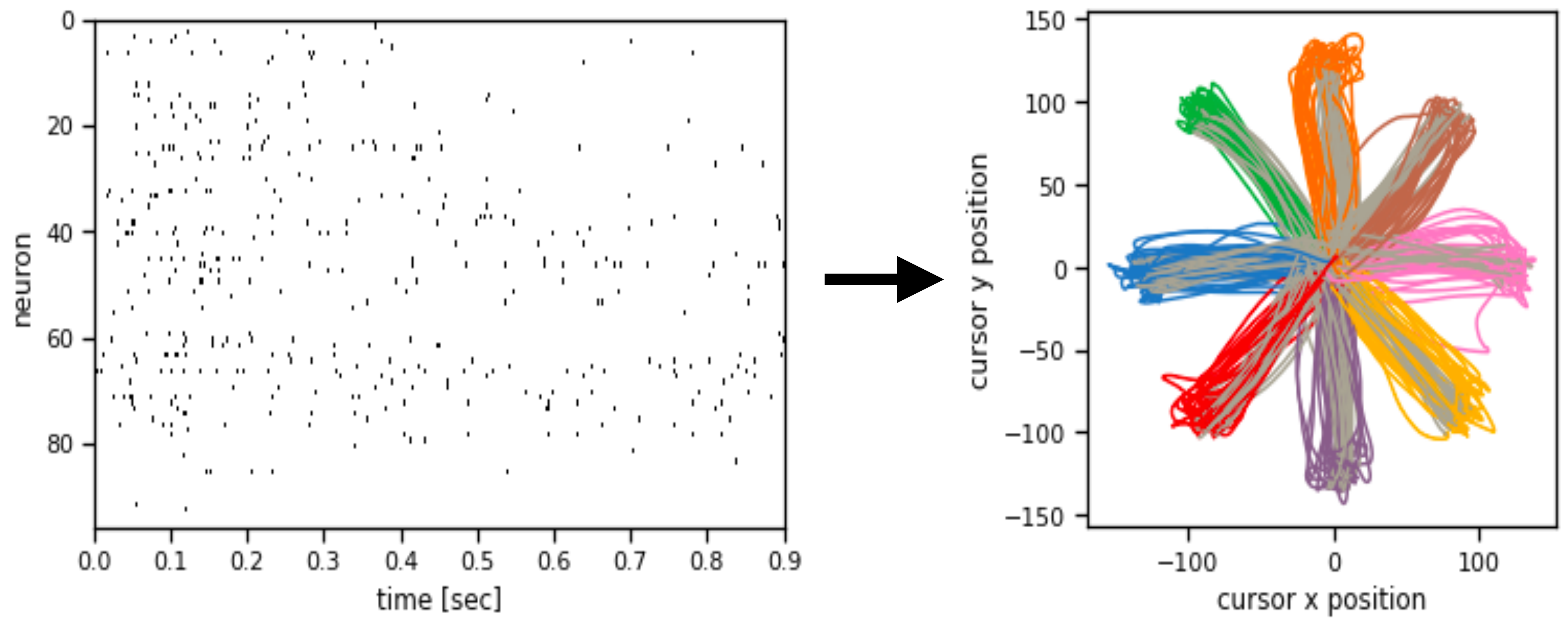


Decoding movement from neural spike trains

A simple example

- Consider the following prior…

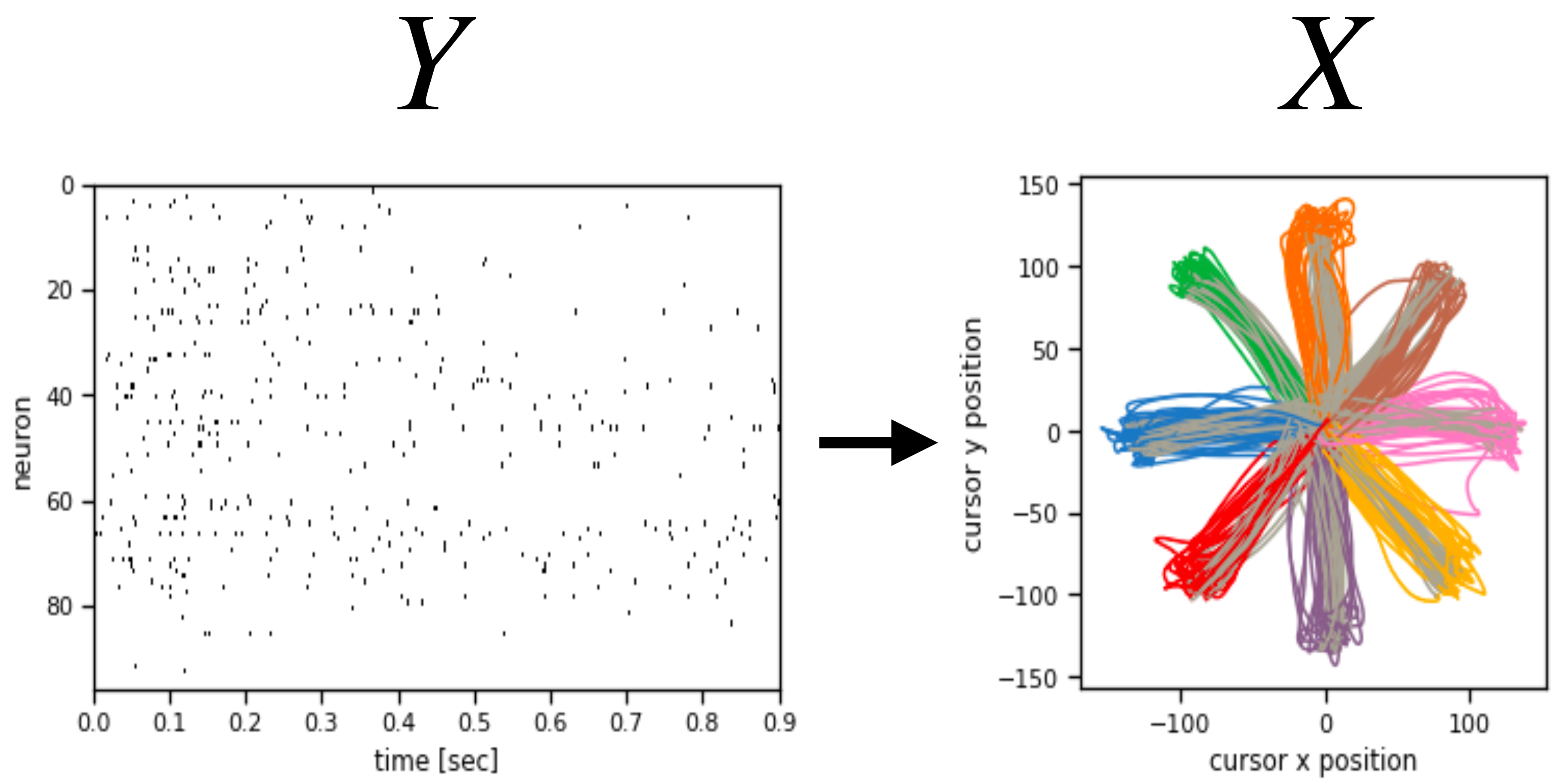
Y X



Decoding movement from neural spike trains

A simple example

Question: What are some limitations of this model?

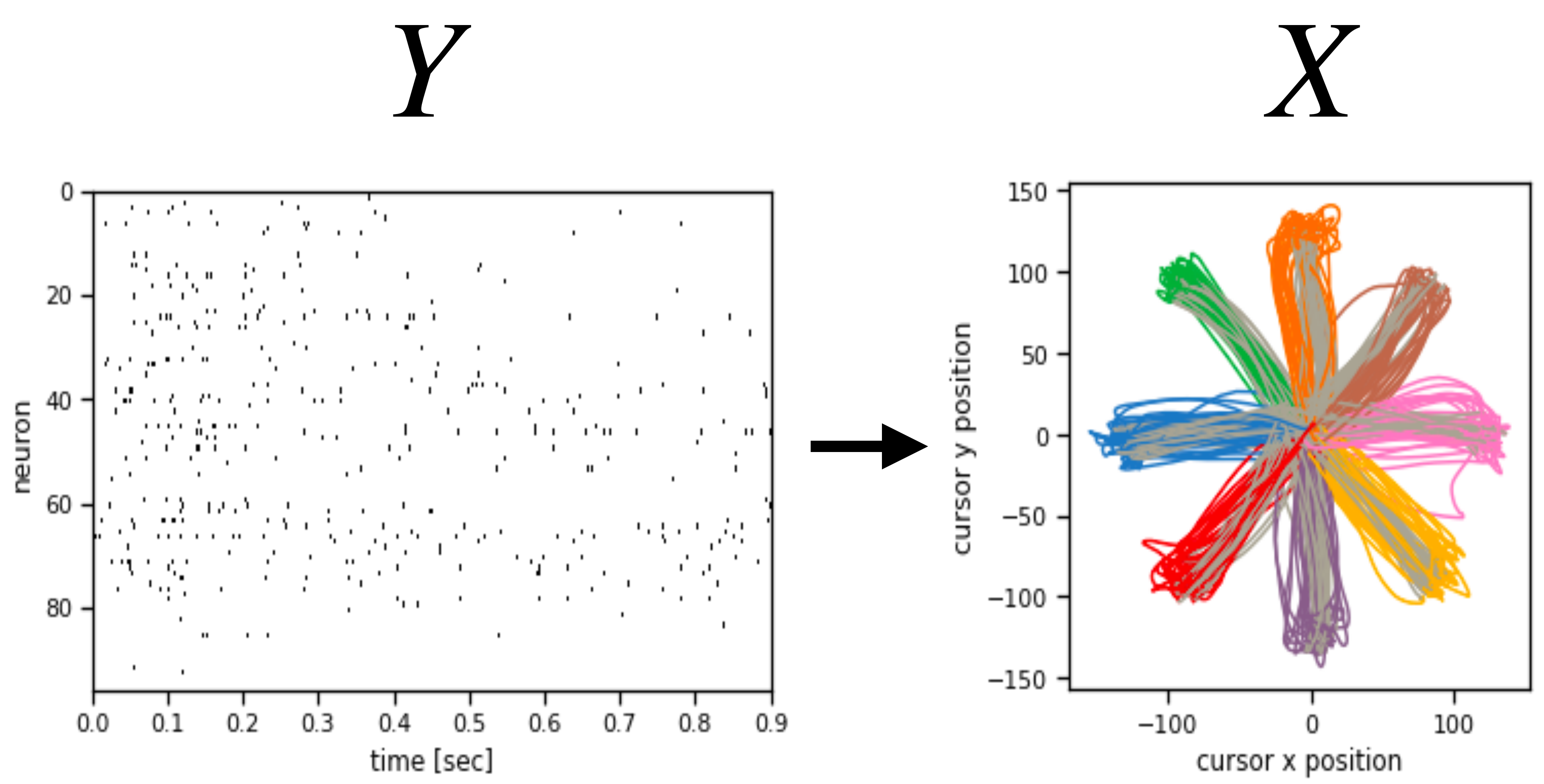


Decoding movement from neural spike trains

Deriving the posterior (decoder)

One good thing about this model is it's easy to work with!

Derive the posterior…



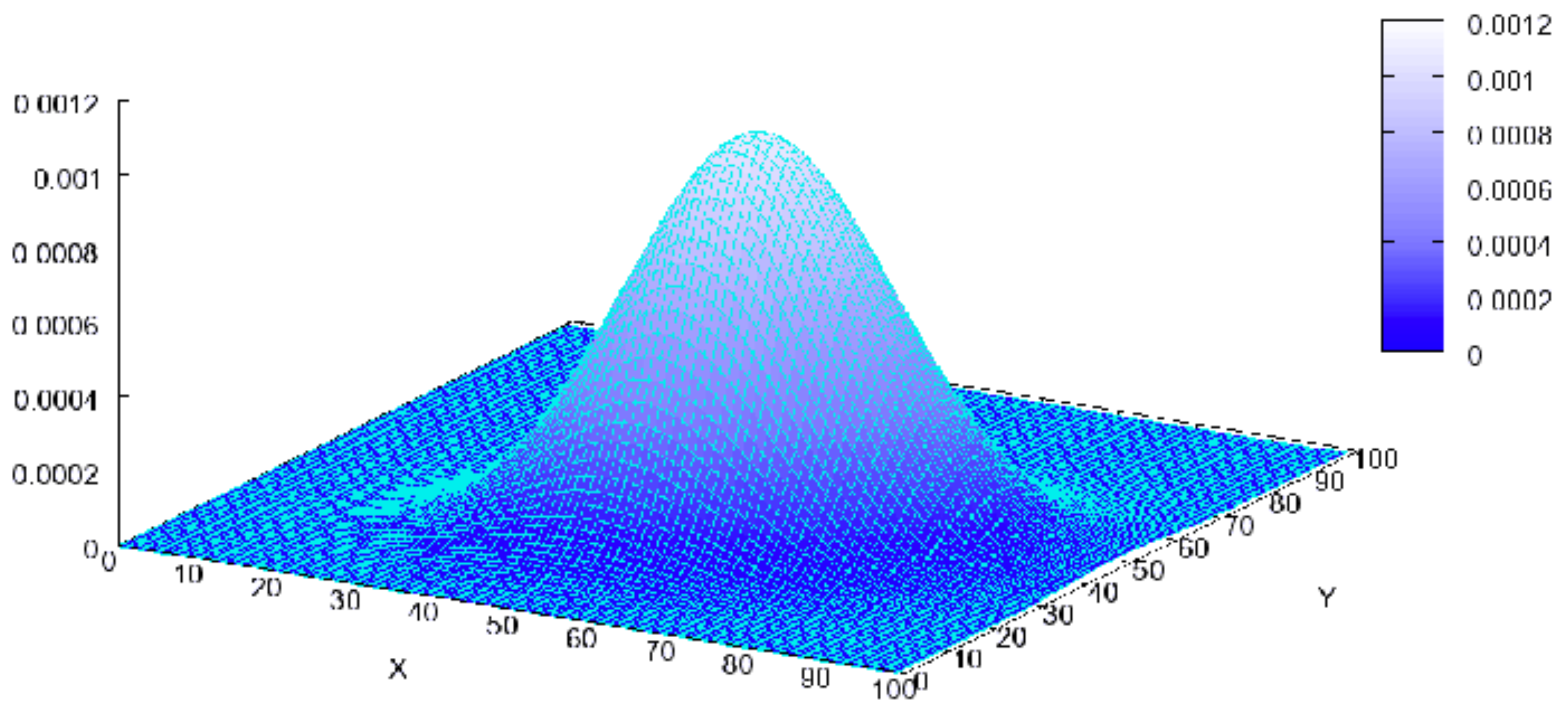
Aside: the multivariate Gaussian distribution

The multivariate Gaussian distribution

- Start with the standard normal distribution,
  - $z_d \sim \mathcal{N}(0,1) \iff p(z_d) = (2\pi)^{-1/2} \exp\left\{-\frac{z_d^2}{2}\right\}$
- Let $z = (z_1, \ldots, z_D)$ denote a vector of iid standard normal r.v.'s. Then,

$$\begin{aligned} p(z) &= \prod_{d=1}^{D} p(z_d) \\ &= \prod_{d=1}^{D} (2\pi)^{-1/2} \exp\left\{-\frac{z_d^2}{2}\right\} \\ &= (2\pi)^{-D/2} \exp\left\{-\frac{1}{2} z^\top z\right\} \end{aligned}$$

- We say $z \sim \mathcal{N}(0,I)$, a **multivariate normal distribution** with mean 0 and covariance I.

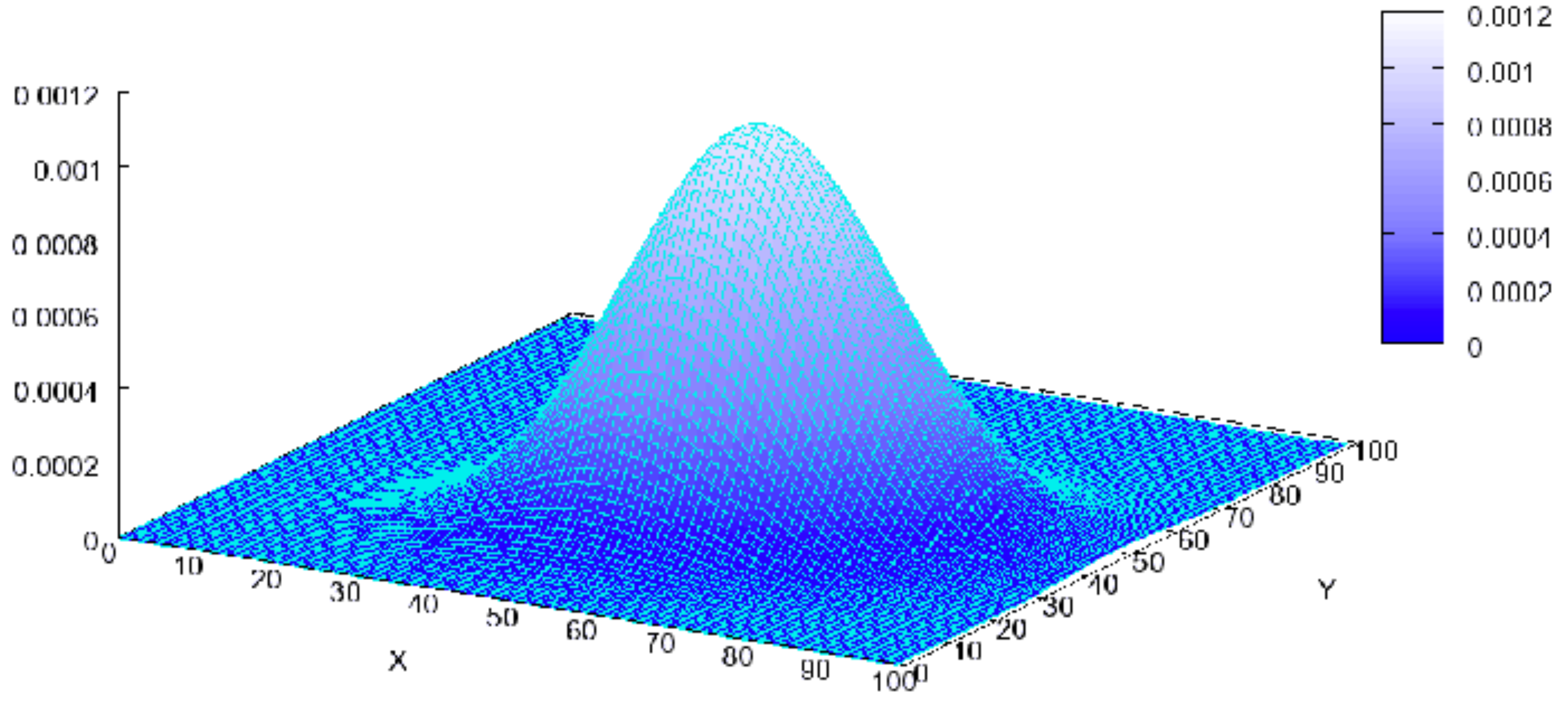


https://en.wikipedia.org/wiki/Multivariate_normal_distribution

Aside: the multivariate Gaussian distribution

- Now let $x = \mu + \Sigma^{1/2} z$ for $\mu \in \mathbb{R}^D$ and (invertible) $\Sigma^{1/2} \in \mathbb{R}^{D \times D}$.
- Then $z = \Sigma^{-1/2}(x - \mu)$.
- Change of variables formula:

$$
\begin{aligned}
p(x) &= \left| \frac{dz}{dx} \right| p(z(x)) \\
&= |\Sigma^{-1/2}| \mathcal{N}(\Sigma^{-1/2}(x - \mu), I) \\
&= (2\pi)^{-D/2} |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2}(x - \mu)^\top \Sigma^{-1}(x - \mu) \right\} \\
&\triangleq \mathcal{N}(x \mid \mu, \Sigma)
\end{aligned}
$$

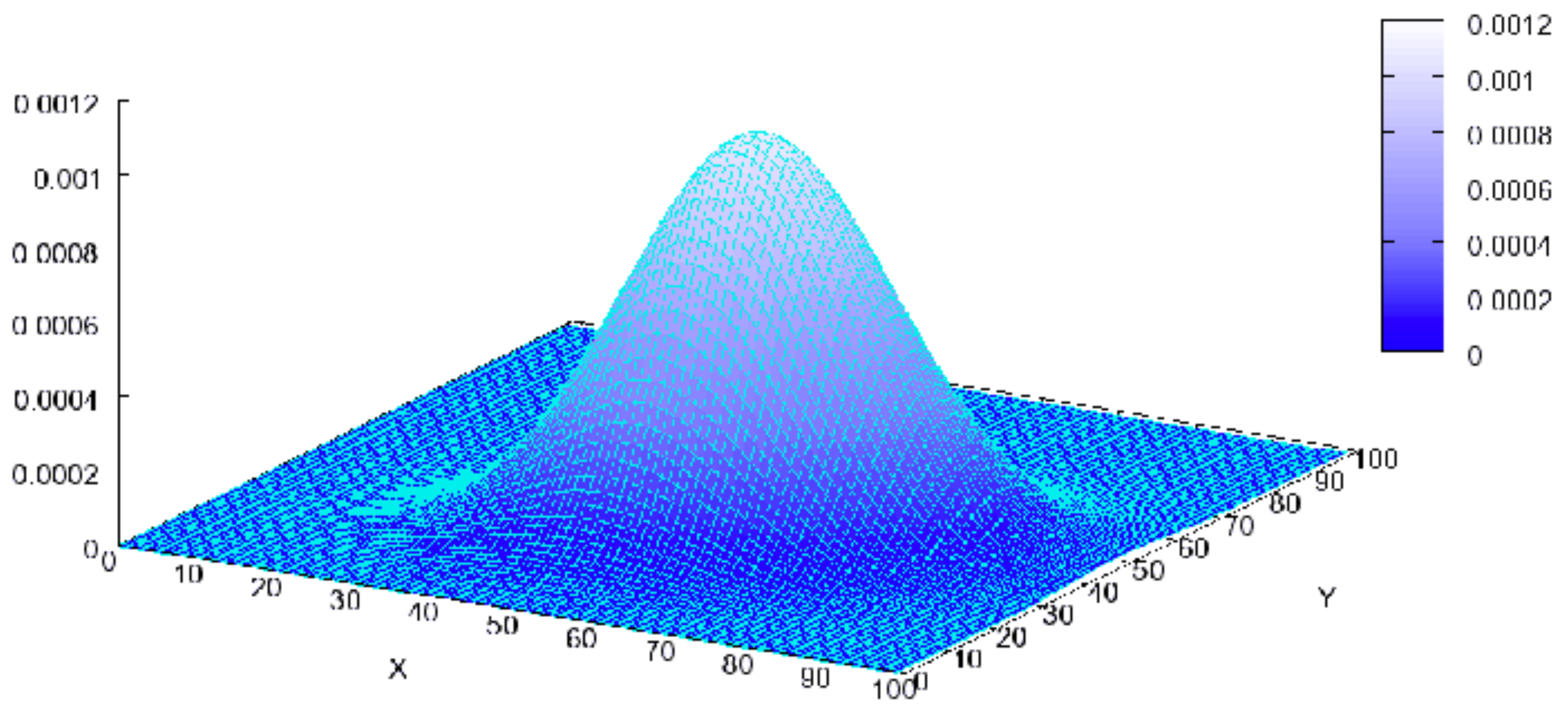


https://en.wikipedia.org/wiki/Multivariate_normal_distribution

Aside: the multivariate Gaussian distribution

“Information” form / natural parameters

$$p(x) = (2\pi)^{-D/2} \exp\left\{ -\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu) \right\}$$



https://en.wikipedia.org/wiki/Multivariate_normal_distribution

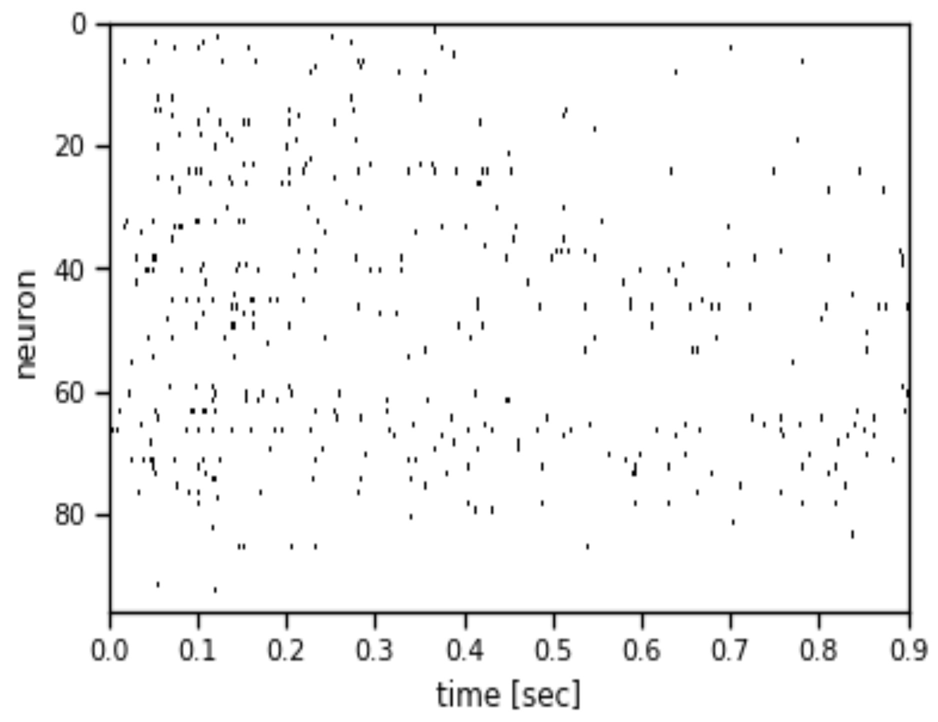
Decoding movement from neural spike trains

Deriving the posterior (decoder)

$$p(X \mid Y) \propto \prod_{t=1}^{T} \left[p(y_t \mid x_t)\, p(x_t) \right]$$

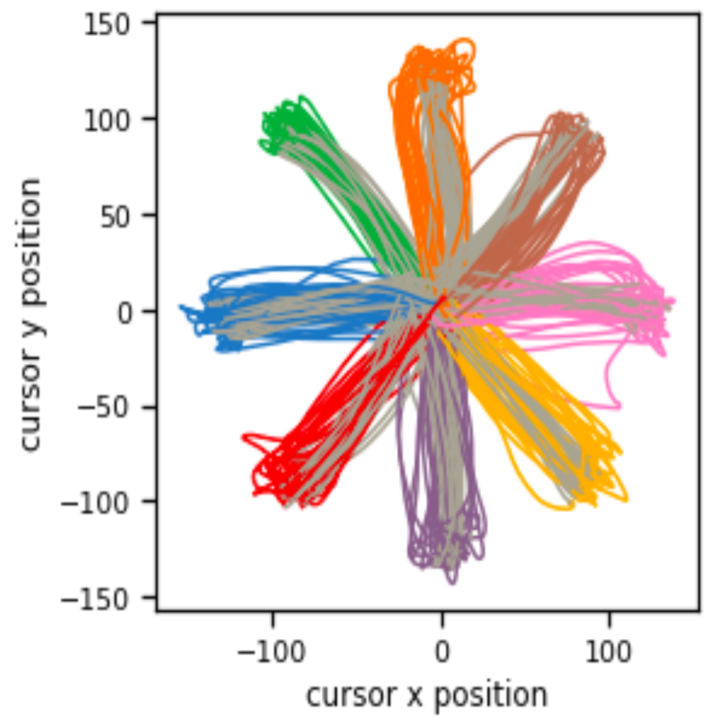
$$= \prod_{t=1}^{T} \mathcal{N}(y_t \mid Cx_t + d, R)\, \mathcal{N}(x_t \mid 0, Q)$$

Y



$\rightarrow$

X



Improving upon the basic model

Decoding movement from neural spike trains

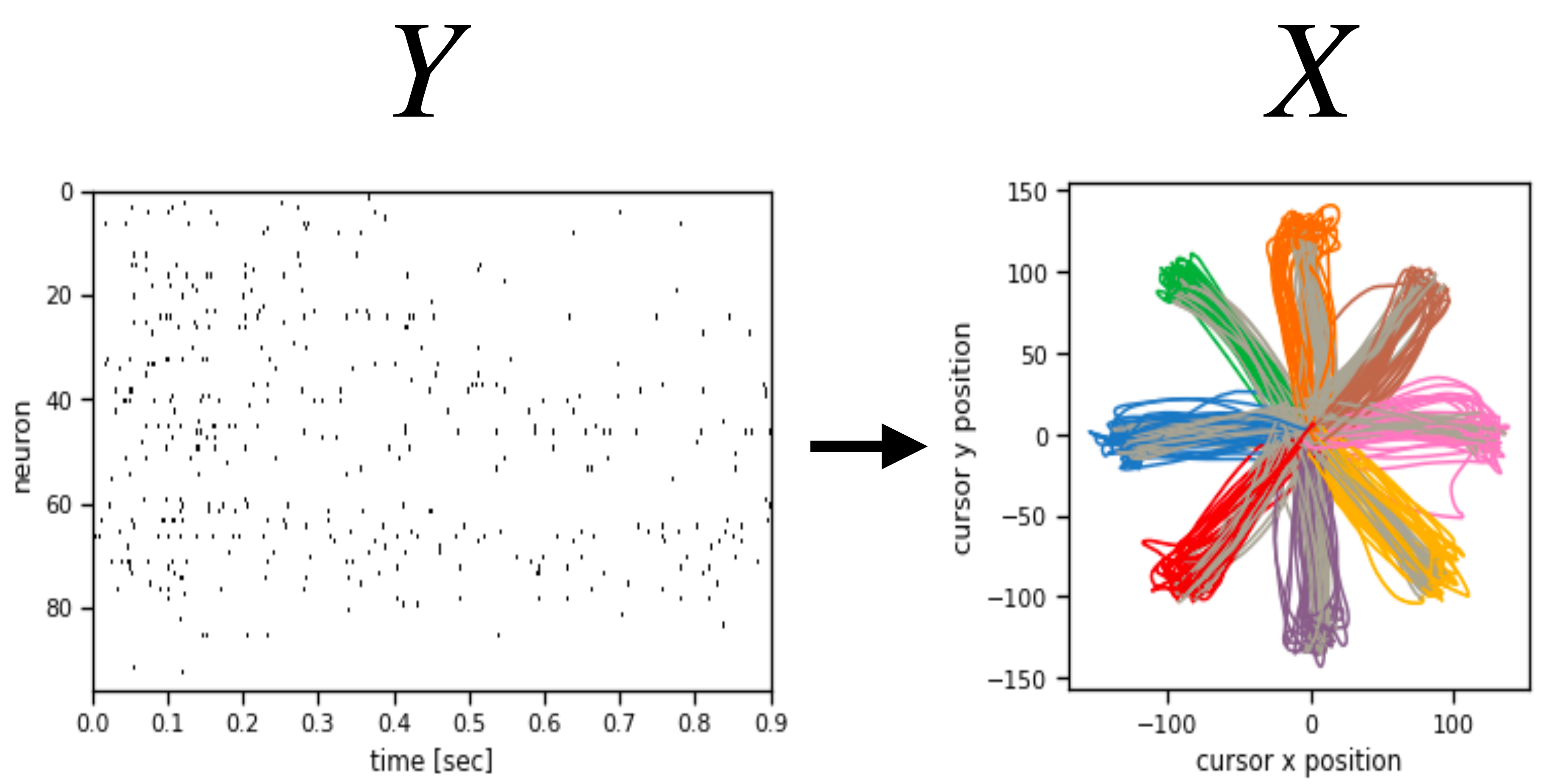
Prior on the sequence of velocities

- One of the problems with the basic model is that it treated each time bin as independent.

- Instead, consider the following prior

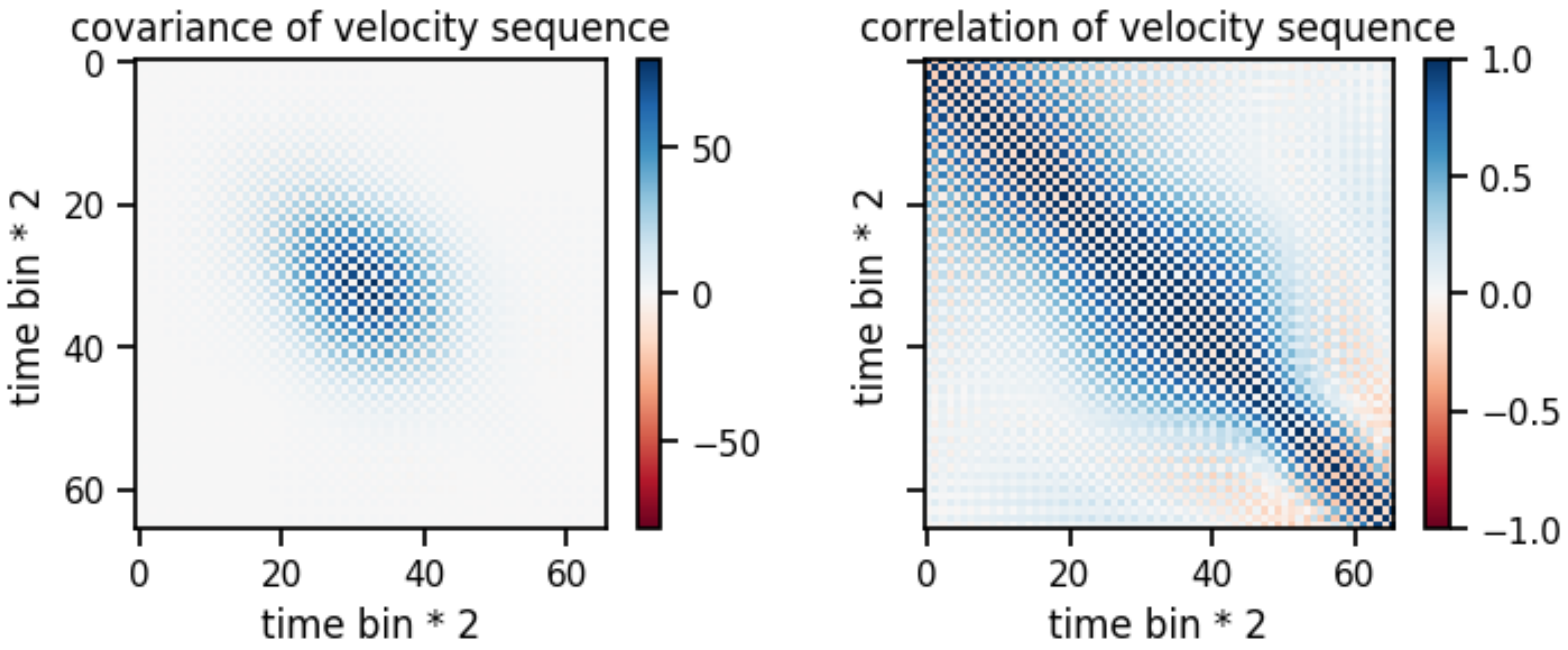
$$p(X) = \mathcal{N}(\mathrm{vec}(X) \mid m, Q)$$

where $\mathrm{vec}(X) = (x_1, \ldots, x_T) \in \mathbb{R}^{2T}$, and $m \in \mathbb{R}^{2T}$ and $Q \in \mathbb{R}^{2T \times 2T}$ are the mean and covariance of the prior, respectively.



Decoding movement from neural spike trains

Prior covariance



Decoding movement from neural spike trains

Derive the posterior under the new model

Decoding movement from neural spike trains

Derive the posterior under the new model

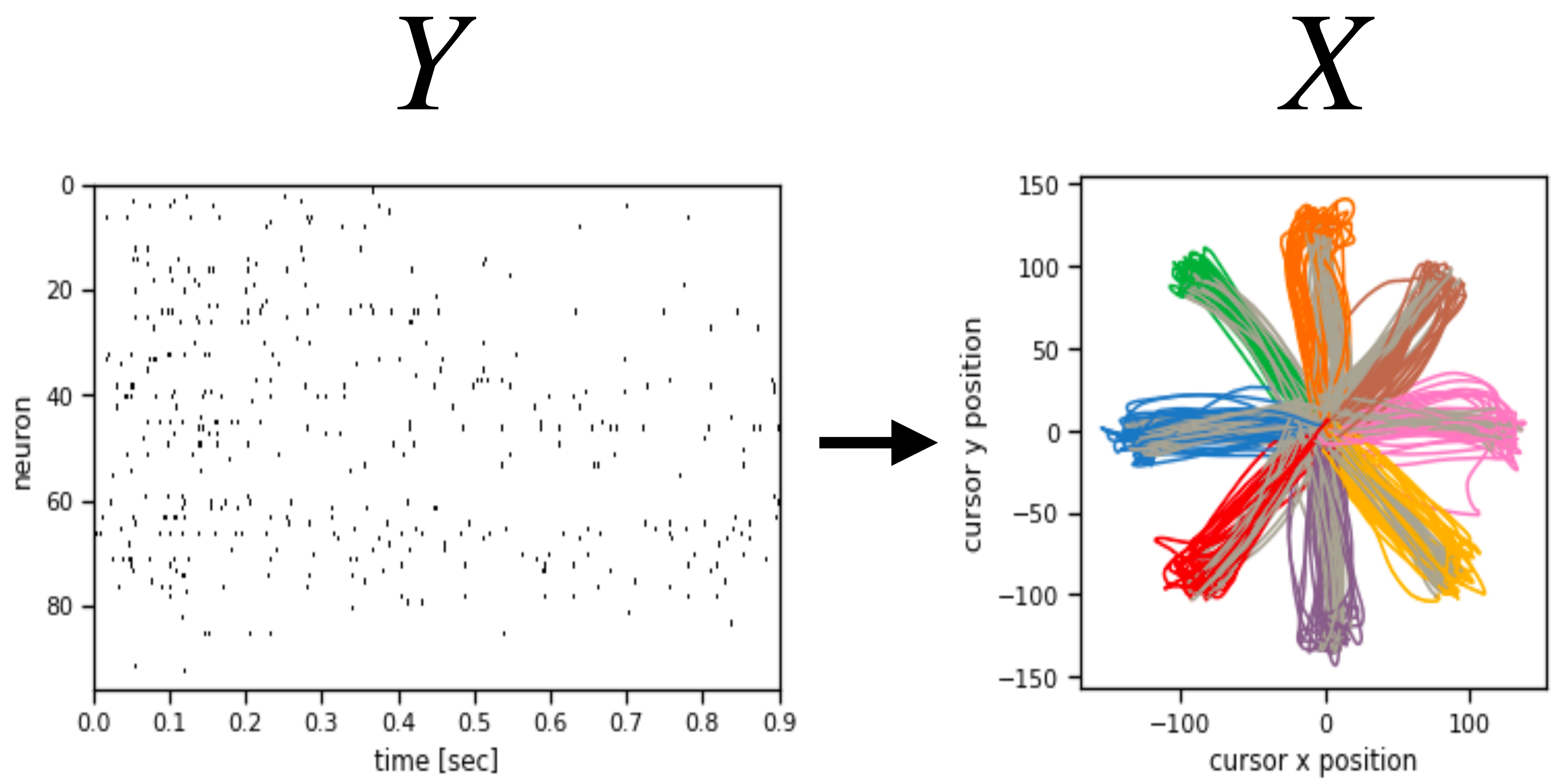
Decoding movement from neural spike trains

Poisson observations

- So far we've used a linear, Gaussian encoder for the spikes, even though they are counts!

- Suppose instead,

$$p(Y \mid X) = \prod_{t=1}^{T} \prod_{n=1}^{N} \mathrm{Po}\left(y_{tn} \mid f(c_n^\top x_t + d_n)\right)$$



- The posterior is no longer Gaussian, but it's common to approximate it as one.

Decoding movement from neural spike trains

Laplace approximation

Approximate the posterior as

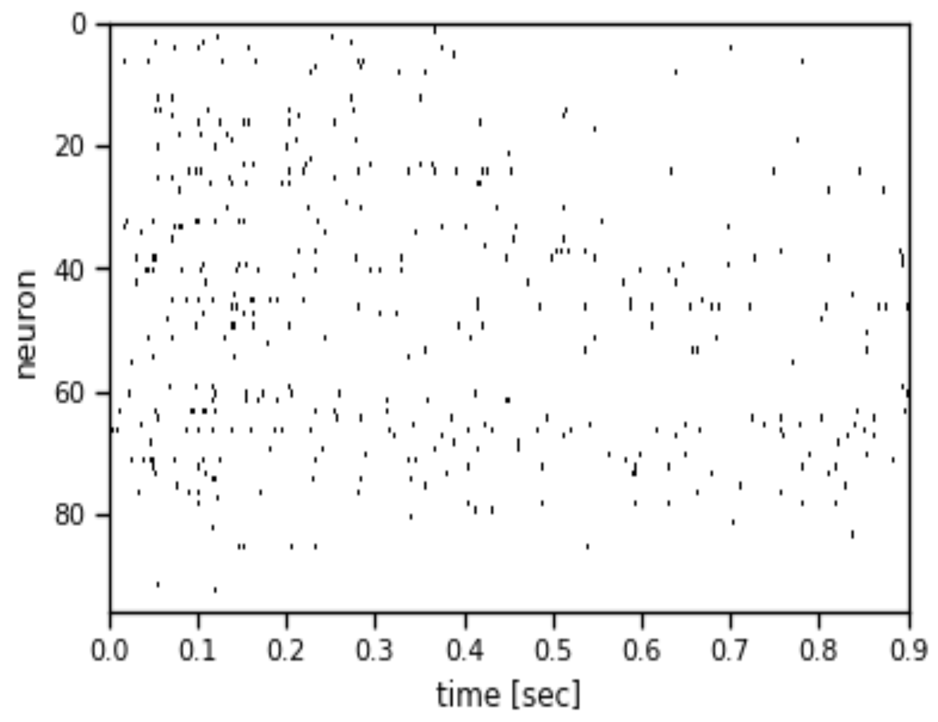
$$p(X \mid Y) \approx \mathcal{N}(\mu, \Sigma)$$

where

$$\mathcal{L}(X) = -\log p(X, Y)$$
$$\mu = \operatorname{argmin}_X \mathcal{L}(X)$$
$$\Sigma = \left[\nabla^2 \mathcal{L}(X) \Big|_{X=\mu} \right]^{-1}$$

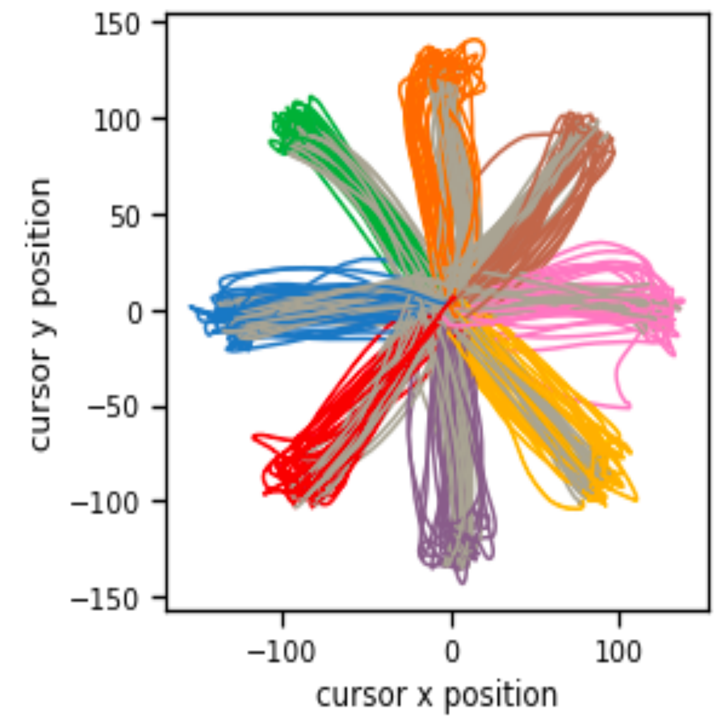
For GLM encoders, the log joint is concave and μ and Σ can be found efficiently.

Y



$\rightarrow$

X



Decoding movement from neural spike trains

Laplace approximation under a Poisson GLM encoder

Derive the Hessian under the Poisson GLM encoder, $\log p(Y \mid X) = \sum_{t=1}^{T} \sum_{n=1}^{N} \log \mathrm{Po}\left(y_{tn} \mid f(c_n^\top x_t + d_n)\right)$

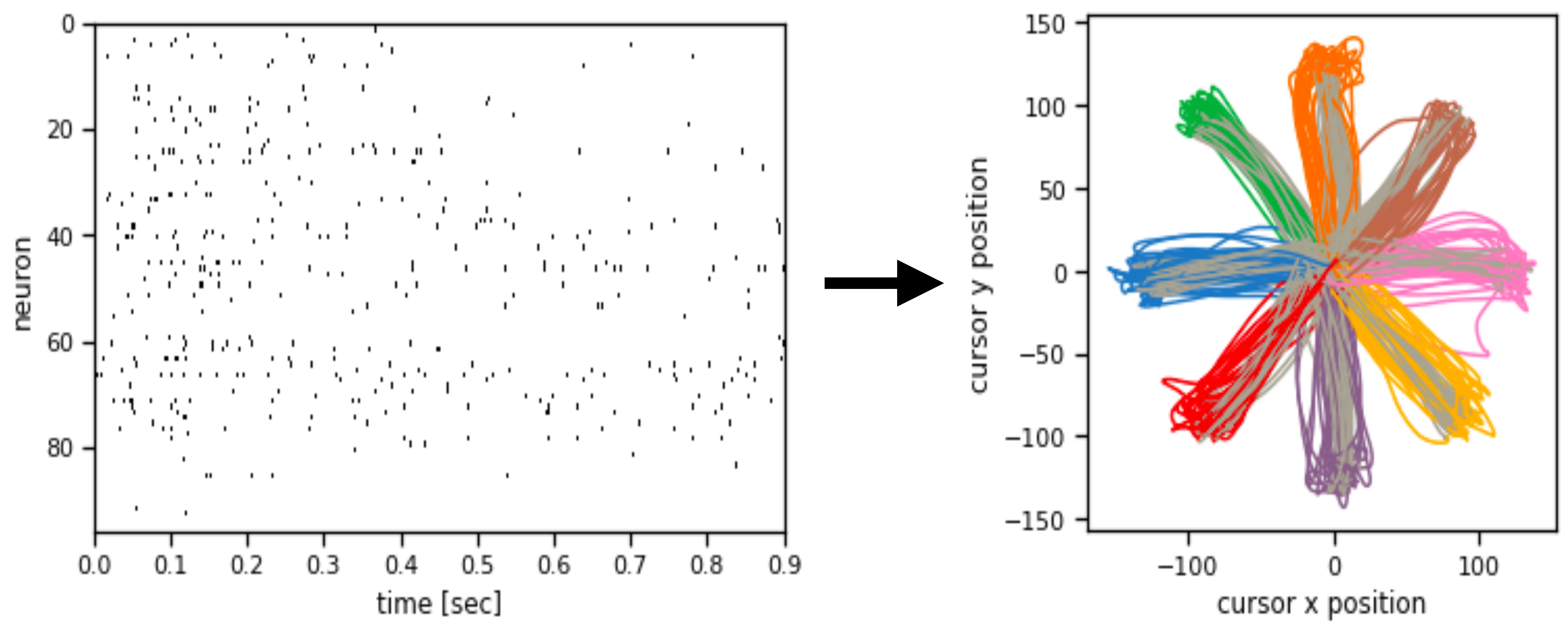
$$\nabla^2 \mathscr{L}(X) =$$

Decoding movement from neural spike trains

Further improvements

Question: We've added a prior on X and a Poisson GLM encoding model. How else could we improve the model?

Y X



Conclusion

- Encoding and decoding are two sides of the same coin.
- We can treat decoding as a simple regression problem, but sometimes we can improve performance by leveraging prior information about X or the encoder $p(Y \mid X)$.
- Bayes' rule tells how to combine prior and likelihood to obtain the posterior.
- However, the posterior rarely has a simple, closed form, so we need to approximate it instead. The Laplace approximation works well when the encoder is a Poisson GLM.