Generalized Linear Model

By Anqi Wu

neuromatch academy

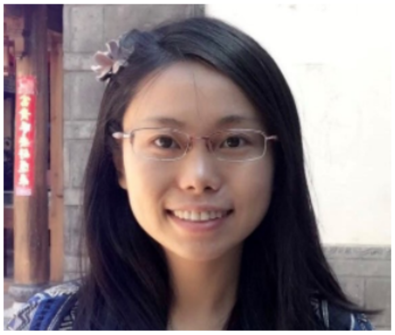
Who is Anqi Wu?

- Postdoc at Columbia with Liam Paninski and John Cunningham

- PhD at Princeton with Jonathan Pillow

- Research interest: neural sensory encoding, latent variable models for large-scale neural recordings, fMRI decoding, behavior analysis

- Modeling: probabilistic graphical model, Gaussian process, latent variable/dynamic model, Bayesian deep learning, Bayesian optimization and active learning



Roadmap of Week 1 Day 4

Tutorial part 1

- Linear Gaussian model
- Poisson GLM

Tutorial part 2

- Logistic regression
- Regularization
  - Ridge (L2)
  - Lasso (L1)

Linear Gaussian Model

On day 3, we talked about the multiple linear model :

Vector form

neural response

$$y_i = \boldsymbol{\theta}^\top \boldsymbol{x}_i \qquad \forall i = 1, ..., N$$

$\boldsymbol{\theta} = [\theta_0, \theta_1, \theta_2, ...]^\top$

linear weights

$\boldsymbol{x}_i = [1, x_{i,1}, x_{i,2}, ...]^\top$

multiple stimulus features (e.g., orientation, contrast, etc.)

number of data points

Matrix form

$$\boldsymbol{y} = X\boldsymbol{\theta}$$

Index i

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} \underbrace{\begin{bmatrix} — & \boldsymbol{x}_0 & — \\ — & \boldsymbol{x}_1 & — \\ — & \boldsymbol{x}_2 & — \\ & \vdots & \end{bmatrix}}_{\text{design matrix}} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \end{bmatrix}$$

If we assume the observation noise to be Gaussian, then

Vector form

neural response

Gaussian noise
$\eta \sim \mathcal{N}(0, \sigma^2)$

$$y_i = \boldsymbol{\theta}^\top \boldsymbol{x}_i + \eta \quad \forall i = 1, ..., N$$

$\boldsymbol{\theta} = [\theta_0, \theta_1, \theta_2, ...]^\top$
linear weights

$\boldsymbol{x}_i = [1, x_{i,1}, x_{i,2}, ...]^\top$
multiple stimulus features (e.g., orientation, contrast, etc.)

number of data points

Matrix form

$$\boldsymbol{y} = X\boldsymbol{\theta} + \boldsymbol{\eta}$$

Index i

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \end{bmatrix} = \underbrace{\begin{bmatrix} — & \boldsymbol{x}_0 & — \\ — & \boldsymbol{x}_1 & — \\ — & \boldsymbol{x}_2 & — \\ & \vdots & \end{bmatrix}}_{\text{design matrix}} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \end{bmatrix} + \begin{bmatrix} \eta_0 \\ \eta_1 \\ \eta_2 \\ \vdots \end{bmatrix}$$

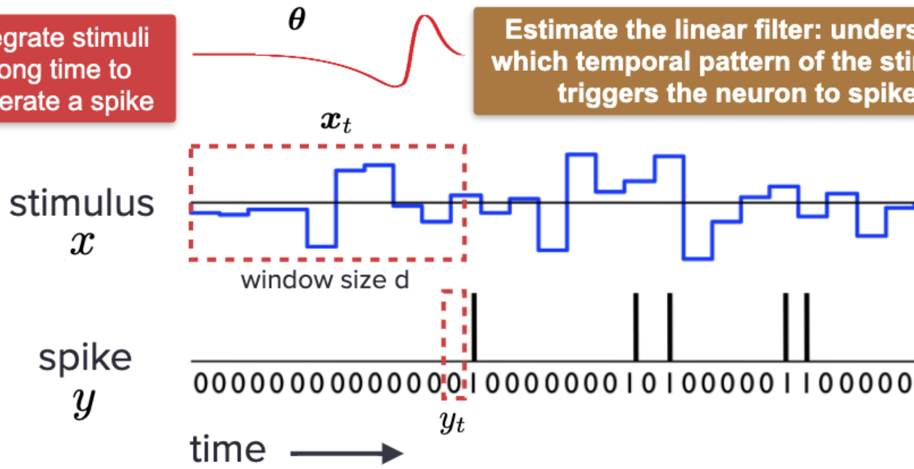
$\eta \sim \mathcal{N}(0, \sigma^2)$

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points

Integrate stimuli along time to generate a spike

Estimate the linear filter: understand which temporal pattern of the stimulus triggers the neuron to spike.



window size

$$\lambda_t = \theta^\top x_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t; linear filter; vector stimulus at time t

$$y_i = \theta^\top x_i + \eta$$

$$y_t = \lambda_t + \eta_t$$

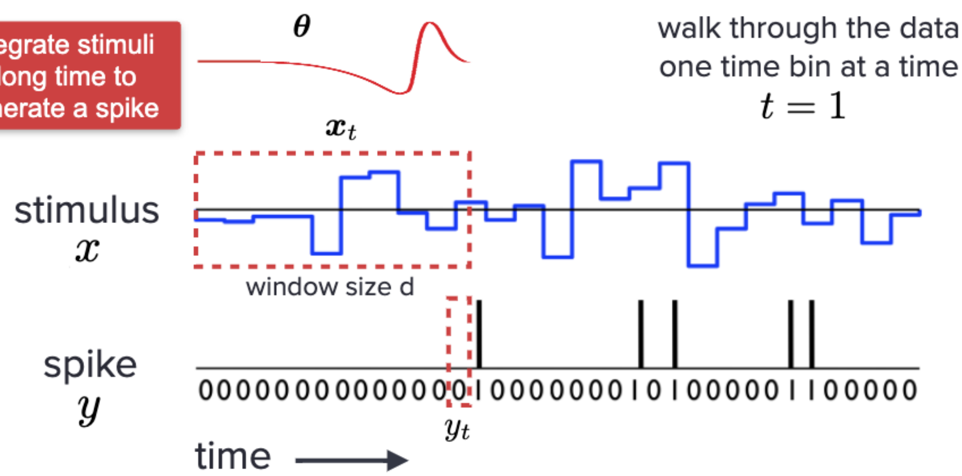
spike at time t; Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points

Integrate stimuli along time to generate a spike



window size

$$\lambda_t = \boldsymbol{\theta}^\top \boldsymbol{x}_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t — linear filter — vector stimulus at time t

$$y_t = \lambda_t + \eta_t$$

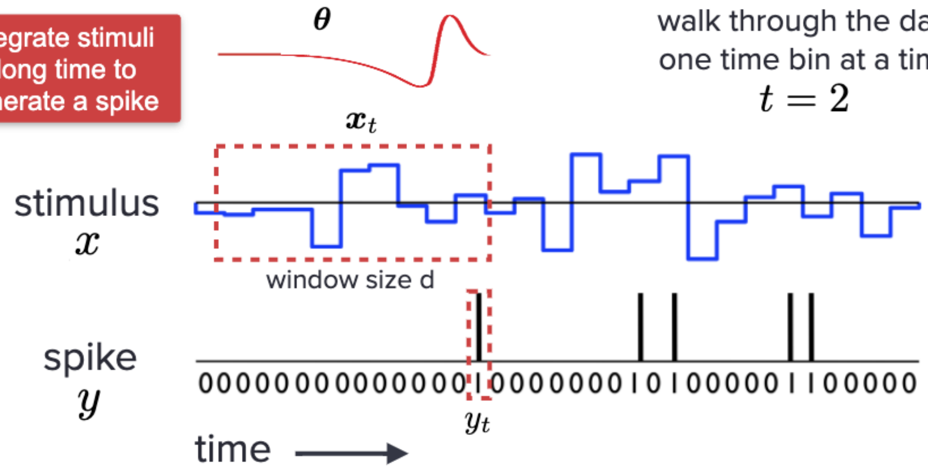
spike at time t — Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points

Integrate stimuli along time to generate a spike



window size

$$\lambda_t = \boldsymbol{\theta}^\top \boldsymbol{x}_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t · linear filter · vector stimulus at time t

$$y_t = \lambda_t + \eta_t$$

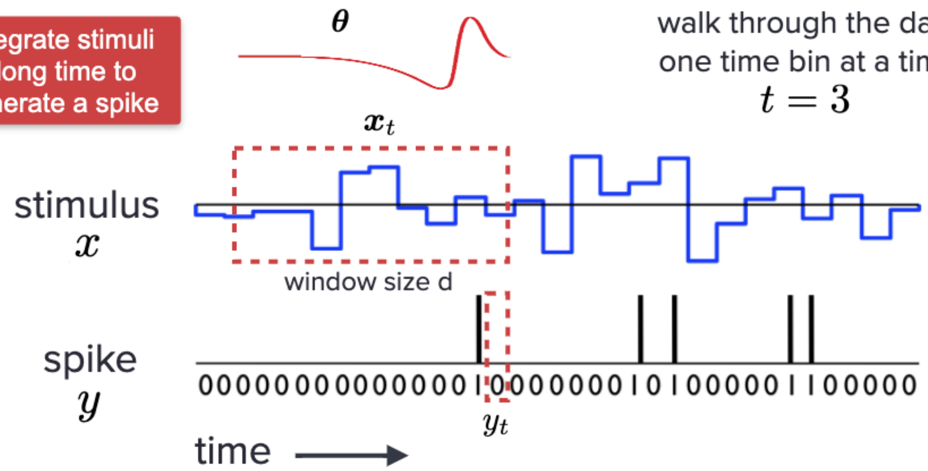
spike at time t · Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points

Integrate stimuli along time to generate a spike



walk through the data one time bin at a time
$t = 3$

window size

$$\lambda_t = \boldsymbol{\theta}^\top \boldsymbol{x}_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t

linear filter

vector stimulus at time t

$$y_t = \lambda_t + \eta_t$$

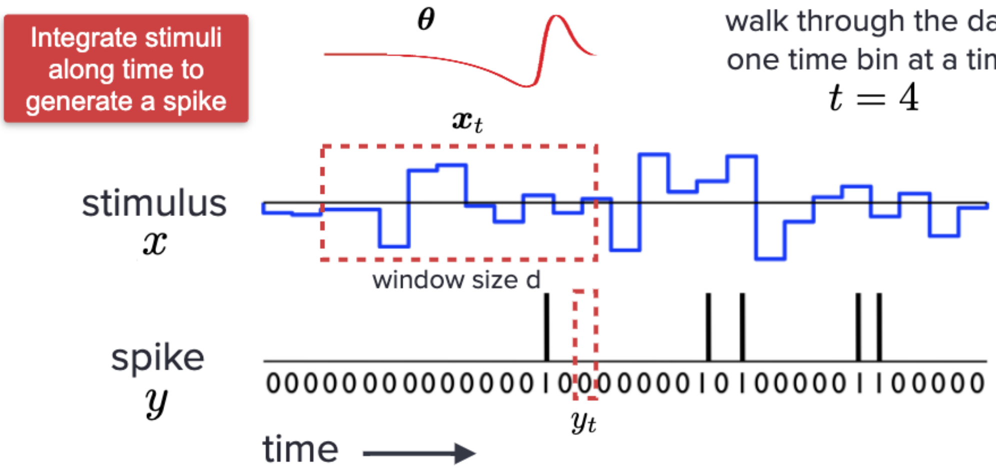
spike at time t

Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points



window size

$$\lambda_t = \boldsymbol{\theta}^\top \boldsymbol{x}_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t — linear filter — vector stimulus at time t

$$y_t = \lambda_t + \eta_t$$

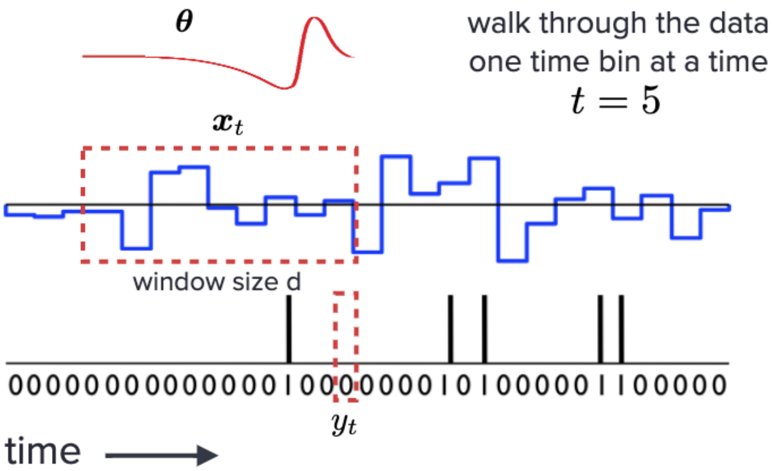
spike at time t — Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points

Integrate stimuli along time to generate a spike



window size

$$\lambda_t = \boldsymbol{\theta}^\top \boldsymbol{x}_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t — linear filter — vector stimulus at time t

$$y_t = \lambda_t + \eta_t$$

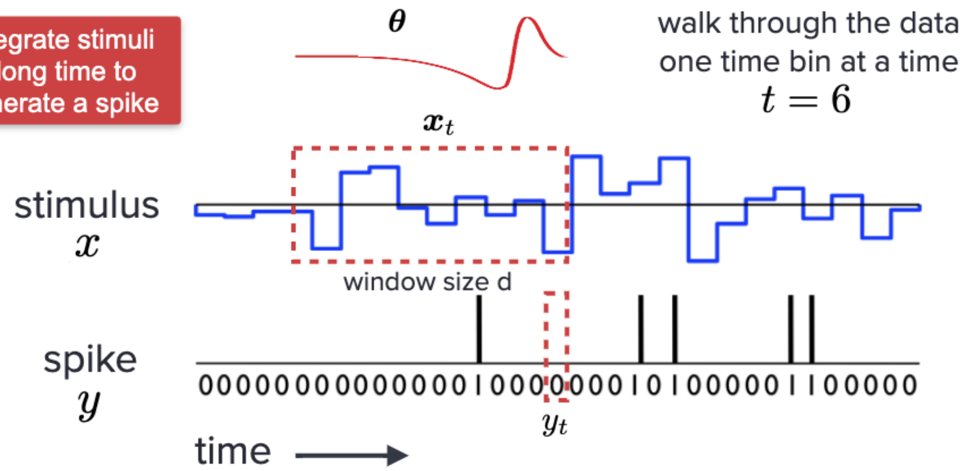
spike at time t — Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Task: predict neural spikes from stimuli at all time points

Integrate stimuli along time to generate a spike



window size

$$\lambda_t = \boldsymbol{\theta}^\top \boldsymbol{x}_t = \sum_{i=0}^{d-1} \theta_i x_{t-i}$$

firing rate at time t — linear filter — vector stimulus at time t

$$y_t = \lambda_t + \eta_t$$

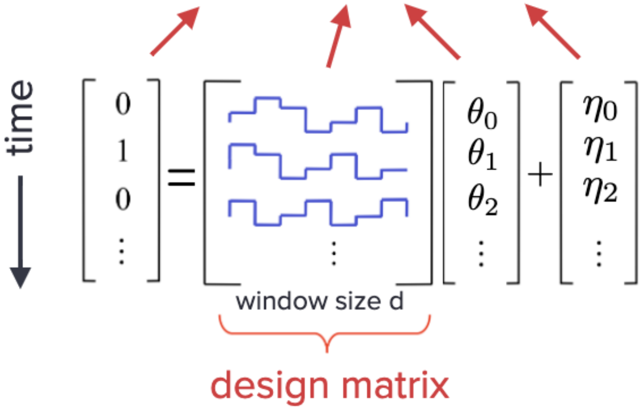
spike at time t — Gaussian noise

source: http://pillowlab.princeton.edu/teaching/statneuro2018/slides/lec09_GLMs1.pdf

Temporal Filtering Model

Build up to the following matrix version along time axis:

$$\boldsymbol{y} = X\boldsymbol{\theta} + \boldsymbol{\eta}$$



MSE solution, ignoring the noise $\boldsymbol{\eta}$ (revisit D3):

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\text{argmin}} \, ||X\boldsymbol{\theta} - \boldsymbol{y}||_2^2 = \sum_{t=1}^{T} (y_t - \boldsymbol{\theta}^\top \boldsymbol{x}_t)^2$$

Differentiate and set to zero

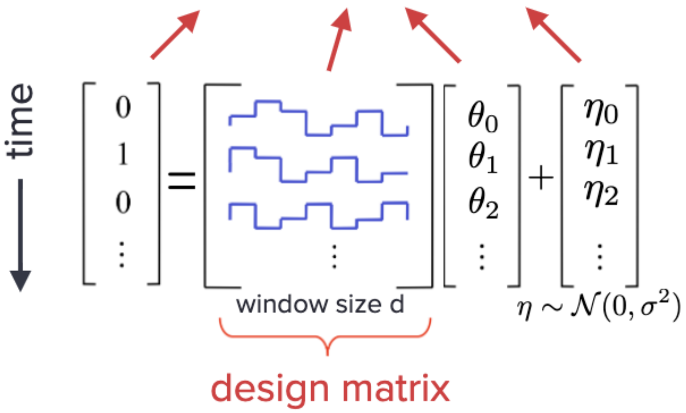
$$\frac{\partial \, ||X\boldsymbol{\theta} - \boldsymbol{y}||_2^2}{\partial \, \boldsymbol{\theta}} = 2X^\top (X\boldsymbol{\theta} - \boldsymbol{y}) = 0$$

$$\Rightarrow \boldsymbol{\theta}_{\text{MSE}} = \underbrace{(X^\top X)}_{\text{stimulus covariance}}{}^{-1} \underbrace{X^\top \boldsymbol{y}}_{\text{spike-triggered avg (STA)}}$$

Temporal Filtering Model

Build up to the following matrix version along time axis:

$$\boldsymbol{y} = X\boldsymbol{\theta} + \boldsymbol{\eta}$$



MLE solution (revisit D3):

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \; \log \mathcal{L}(\boldsymbol{\theta}|X, \boldsymbol{y}) = -\frac{\overset{\text{the number of time points}}{T}\overset{\text{window size}}{d}}{2} \log 2\pi\sigma^2 - \frac{1}{2\sigma^2}(\boldsymbol{y} - X\boldsymbol{\theta})^\top(\boldsymbol{y} - X\boldsymbol{\theta})$$

Differentiate and set to zero

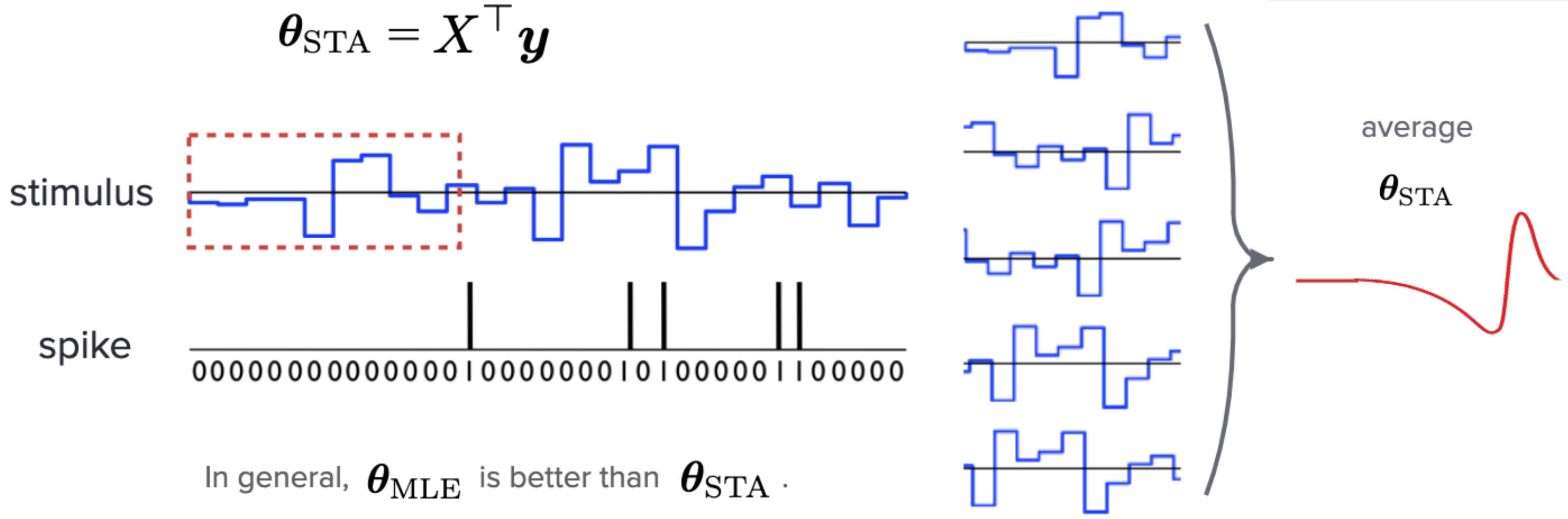
$$\frac{\partial \log \mathcal{L}(\boldsymbol{\theta}|X, \boldsymbol{y})}{\partial \boldsymbol{\theta}} = -\frac{1}{\sigma^2} X^\top (X\boldsymbol{\theta} - \boldsymbol{y}) = 0$$

$$\Rightarrow \boldsymbol{\theta}_{\text{MLE}} = \underbrace{(X^\top X)^{-1}}_{\text{stimulus covariance}} \underbrace{X^\top \boldsymbol{y}}_{\text{spike-triggered avg (STA)}}$$

(same as MSE when the noise is Gaussian)

Spike-triggered Average (STA)

$$\boldsymbol{\theta}_{\mathrm{STA}} = X^{\top}\boldsymbol{y}$$



In general, $\boldsymbol{\theta}_{\mathrm{MLE}}$ is better than $\boldsymbol{\theta}_{\mathrm{STA}}$.

Notebook

In the following notebook, you will
1. Formulate the design matrix from the stimulus vector.
2. Fit the linear Gaussian model to the stimulus and spike count data.
3. Predict spike counts using the fitted linear Gaussian model.

Enjoy!!

Generalized Linear Model

Temporal Filtering Model

Build up to following matrix version along time axis:

The noise is not Gaussian!
The neural response is spike count which is non-negative and discrete!
We need other noise distribution and nonlinear function.

$^2 - \frac{1}{2\sigma^2}(\boldsymbol{y} - X\boldsymbol{\theta})^\top(\boldsymbol{y} - X\boldsymbol{\theta})$

ts and d is the window

time

$$\begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \end{bmatrix} = \begin{bmatrix} \\ \vdots \end{bmatrix} \begin{bmatrix} \theta_2 \\ \vdots \end{bmatrix} + \begin{bmatrix} \eta_2 \\ \vdots \end{bmatrix}$$

window size d

design matrix

$\eta \sim \mathcal{N}(0, \sigma^2)$

$$\frac{\partial \log \mathcal{L}(\boldsymbol{\theta}|X, \boldsymbol{y})}{\partial\, \boldsymbol{\theta}} = -\frac{1}{\sigma^2} X^\top (X\boldsymbol{\theta} - \boldsymbol{y}) = 0$$

$$\Rightarrow \boldsymbol{\theta}_{\text{MLE}} = \underbrace{(X^\top X)^{-1}}_{\text{stimulus covariance}} \underbrace{X^\top \boldsymbol{y}}_{\text{spike-triggered avg (STA)}}$$

(same as MSE when the noise is Gaussian)

Linear model $x \rightarrow$ Linear $\rightarrow$ Noise (exponential family) $\rightarrow y$

Example: linear Gaussian model $y = \theta x + \eta$ where $\eta \sim \mathcal{N}(0, \sigma^2)$

Generalized linear model $x \rightarrow$ Linear $\rightarrow$ Nonlinear $\rightarrow$ Noise (exponential family) $\rightarrow y$

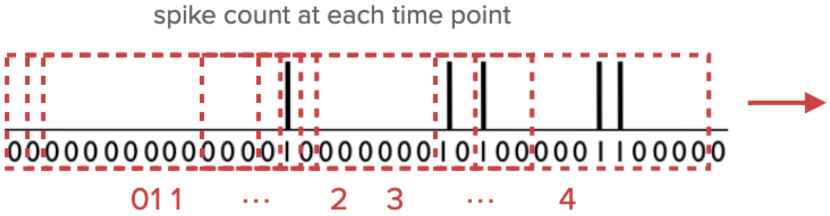
Example: nonlinear Gaussian model $y = f(\theta x) + \eta$ where $\eta \sim \mathcal{N}(0, \sigma^2)$

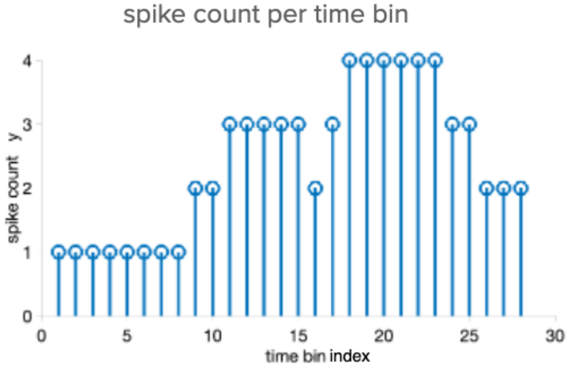
nonlinear
f^{-1}: link function

Poisson GLM $y \sim \mathrm{Poisson}\left(f(\theta x)\right)$ for spike train encoding

Poisson GLM

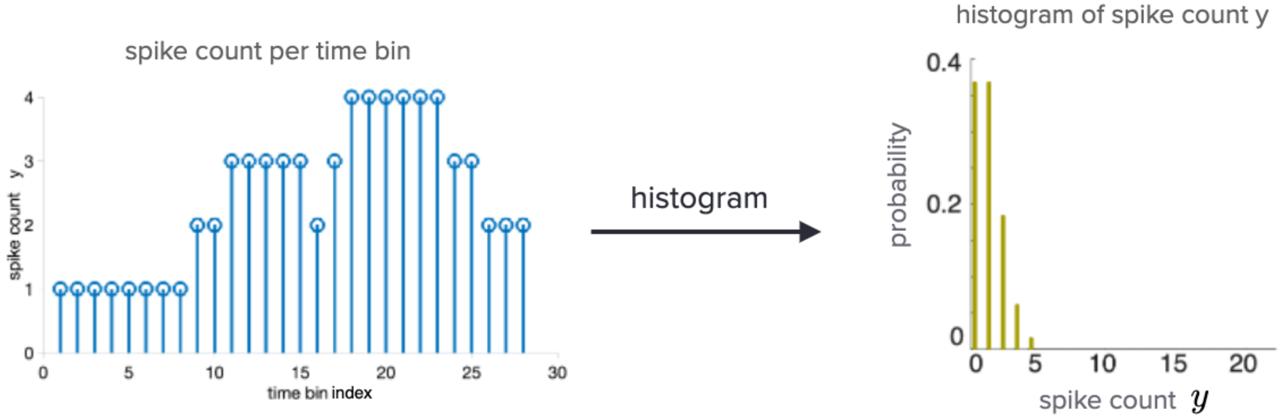
Poisson distribution is used to model the number of events (spikes) occurring within a given time interval.





Poisson GLM

Poisson distribution is used to model the number of events (spikes) occurring within a given time interval.



Poisson GLM

Poisson distribution is used to model the number of events (spikes) occurring within a given time interval.

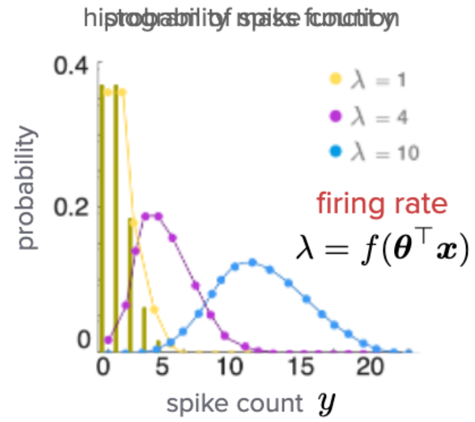
The probability mass function (pmf) of Y is given by

$$P(Y = y) = \frac{\lambda^y e^{-\lambda}}{y!}$$

unknown parameter, >0

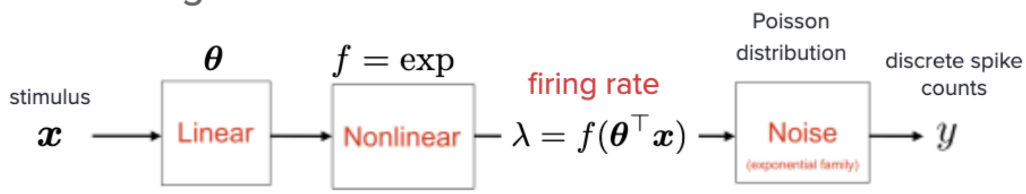
observed discrete spike count y=0,1,2,3,...

not contain λ



Poisson GLM

Temporal filtering model



Given the probability mass function for Poisson distribution $P(Y = y) = \dfrac{\lambda^y e^{-\lambda}}{y!}$

We can now assume the encoding distribution to be

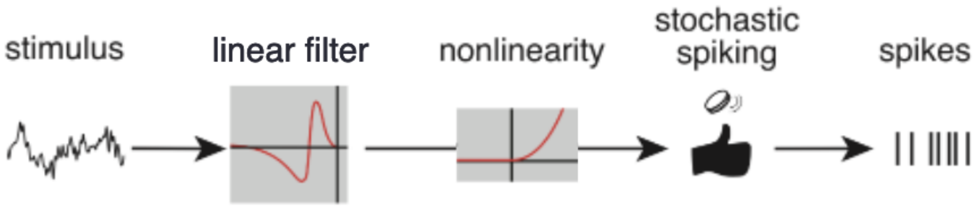
for time t $$p(y_t|\boldsymbol{\theta}, \boldsymbol{x}_t) = P(Y = y_t|\lambda_t = f(\boldsymbol{\theta}^\top \boldsymbol{x}_t)) = \frac{[f(\boldsymbol{\theta}^\top \boldsymbol{x}_t)]^{y_t}}{y_t!} e^{-f(\boldsymbol{\theta}^\top \boldsymbol{x}_t)}$$

Poisson GLM

Temporal filtering model

log likelihood $\log \mathcal{L}(\boldsymbol{\theta}|X, \boldsymbol{y}) = \log p(\boldsymbol{y}|X, \boldsymbol{\theta}) = \boldsymbol{y}^\top \log f(X\boldsymbol{\theta}) - \mathbf{1}^\top f(X\boldsymbol{\theta}) + const$

- Solving $\frac{\partial \log \mathcal{L}(\boldsymbol{\theta}|X, \boldsymbol{y})}{\partial \boldsymbol{\theta}} = 0$ has no closed-form
- $-\log \mathcal{L}(\boldsymbol{\theta}|X, \boldsymbol{y})$ is a convex function
- Convex optimization such as gradient descent



Notebook

In the following notebook, you will

1. Continue the fitting for the same stimulus and spike count data.
2. Fit Poisson GLM by optimization and predict spike counts.
3. Compare the prediction from linear Gaussian and Poisson GLM.

Enjoy!!