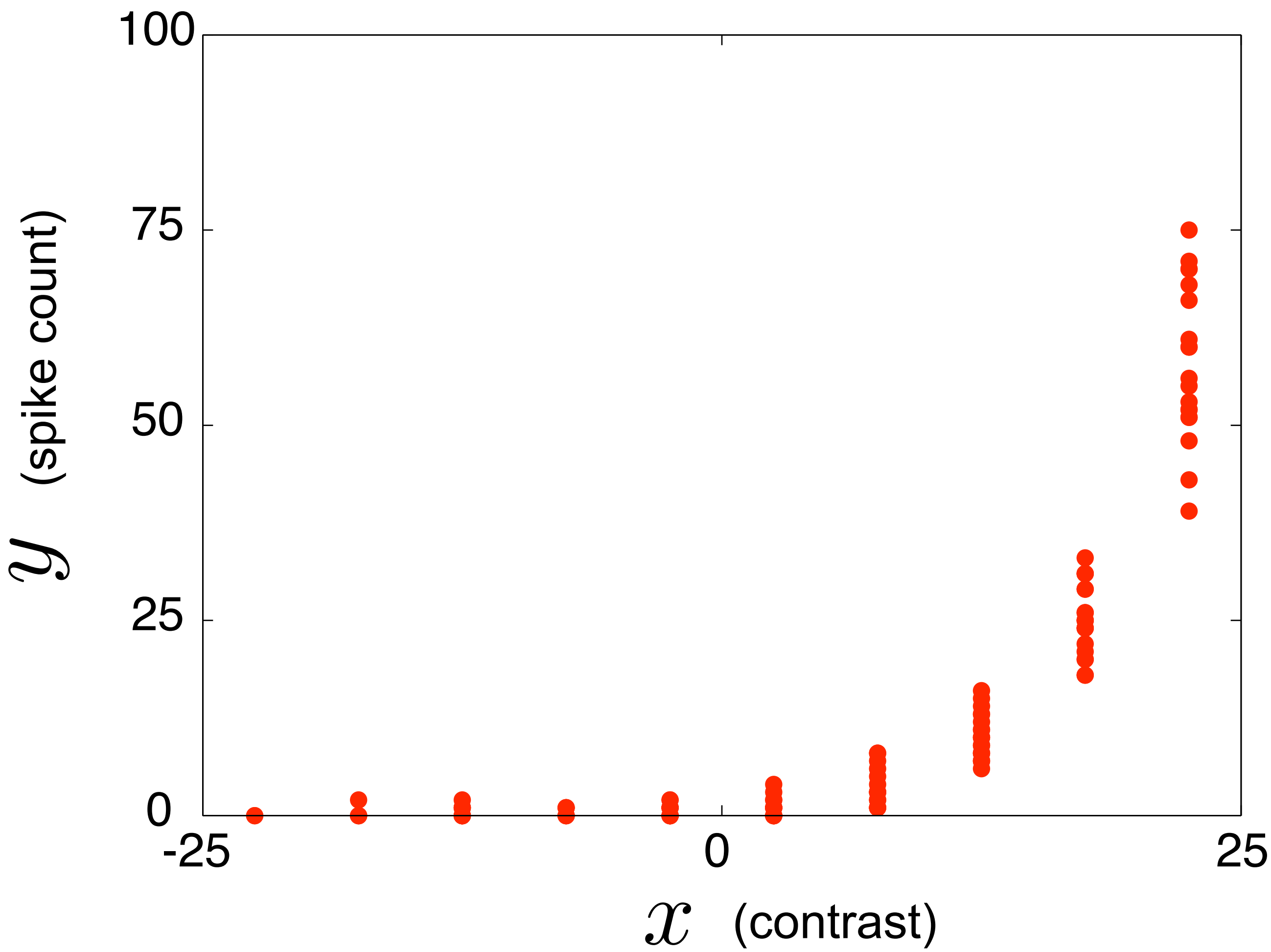
Statistical modeling and analysis of neural data
NEU 560, Spring 2018
Lecture 9

Generalized Linear Models (GLMs)

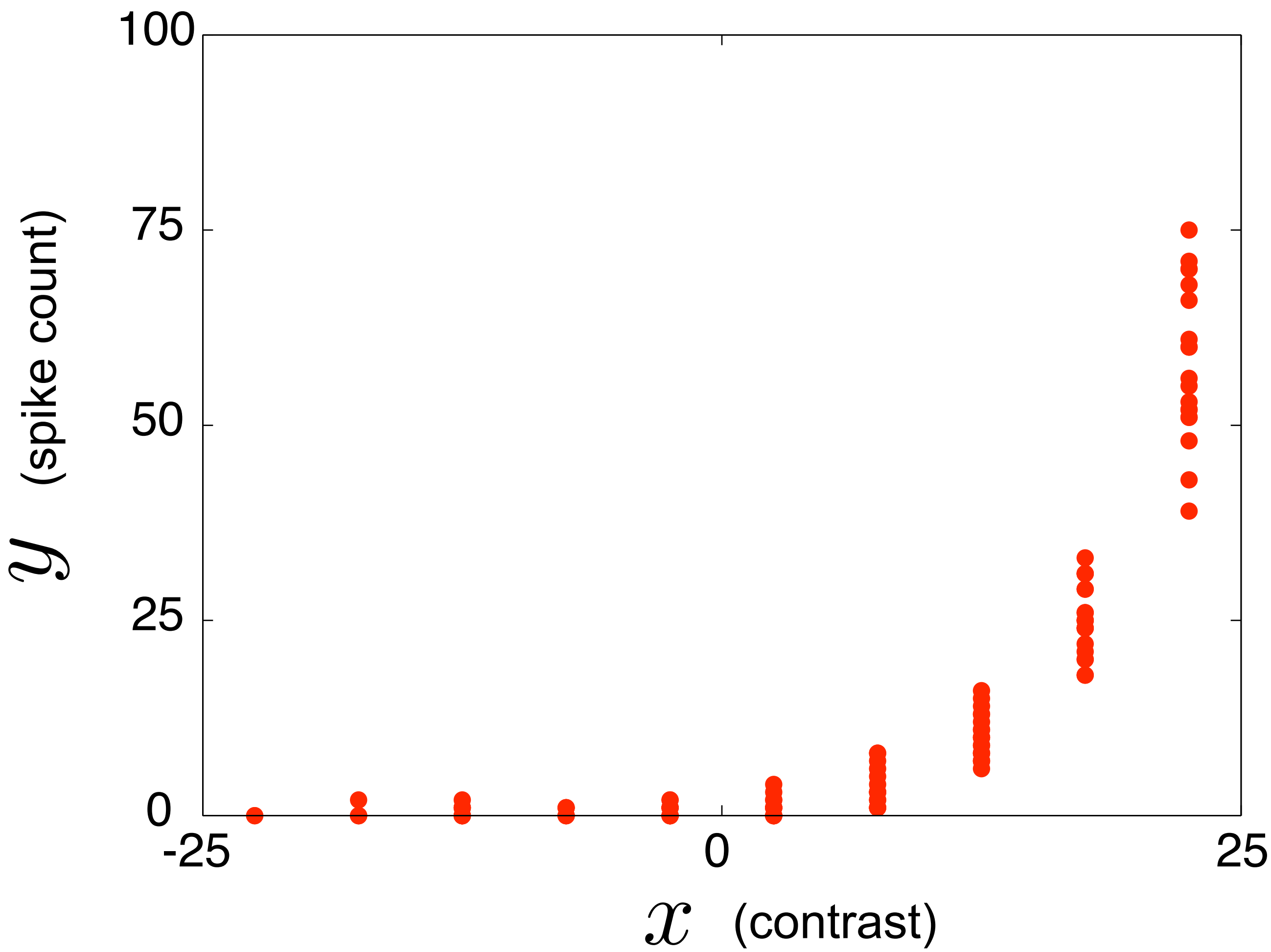
Jonathan Pillow

Example 3: unknown neuron



Be the computational neuroscientist: what model would you use?

Example 3: unknown neuron



More general setup: $y \sim Poiss(\lambda)$

$$\lambda = f(\theta x)$$

for some nonlinear function f

Quick Quiz:

The distribution $P(y|x, \theta)$ can be considered as a function of y, x, or θ.

spikes stimulus parameters

What is $P(y|x, \theta)$:

1. as a function of y?

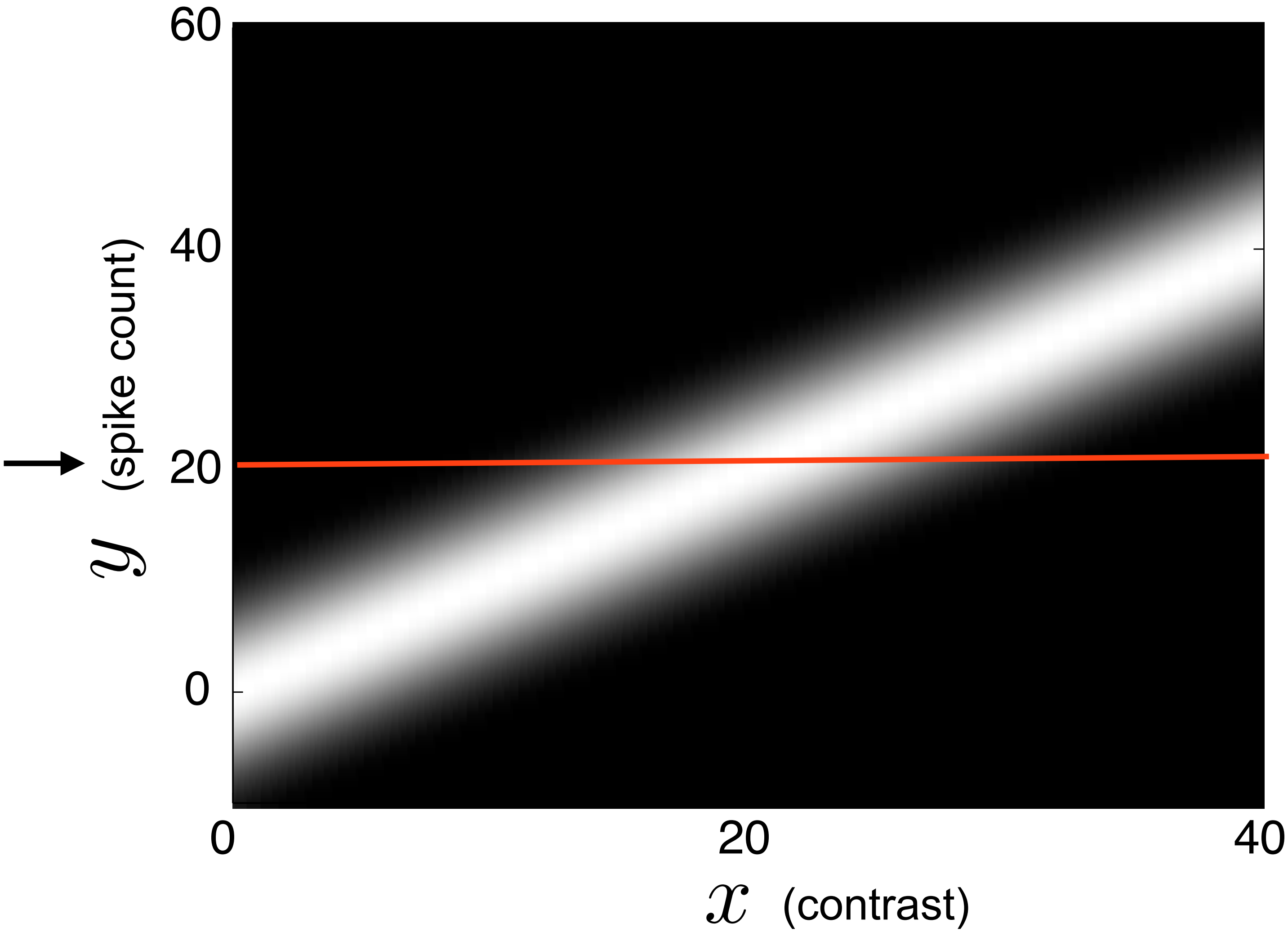
Answer: **encoding distribution** - probability distribution over spike counts

2. as a function of θ ?

Answer: **likelihood function** - the probability of the data given model params

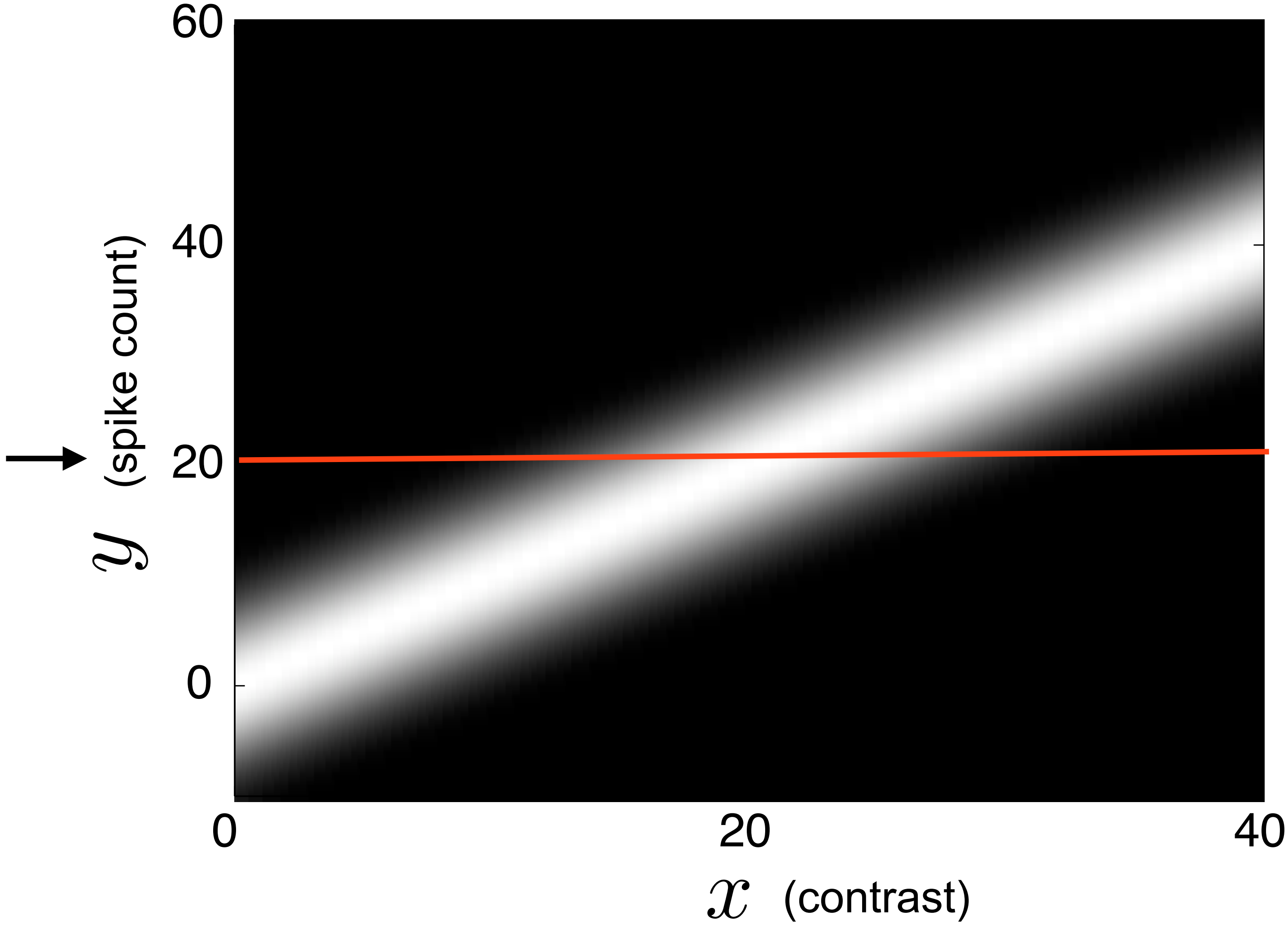
3. as a function of x?

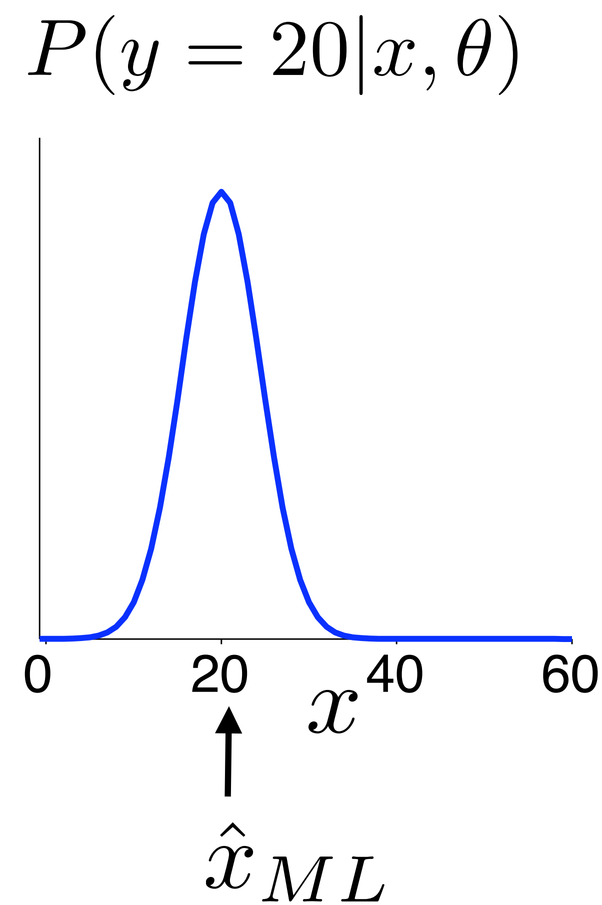
Answer: **stimulus likelihood function** - useful for ML stimulus decoding!



stimulus
decoding
likelihood

What is this?





Stimulus likelihood function
(for decoding)

GLMs

- Be careful about terminology:

GLM ≠ GLM

General Linear Model

Generalized Linear Model

(Nelder 1972)

Linear

~~Linear~~

2003 interview with John Nelder...

Stephen Senn: I must confess to having some confusion when I was a young statistician between general linear models and generalized linear models. Do you regret the terminology?

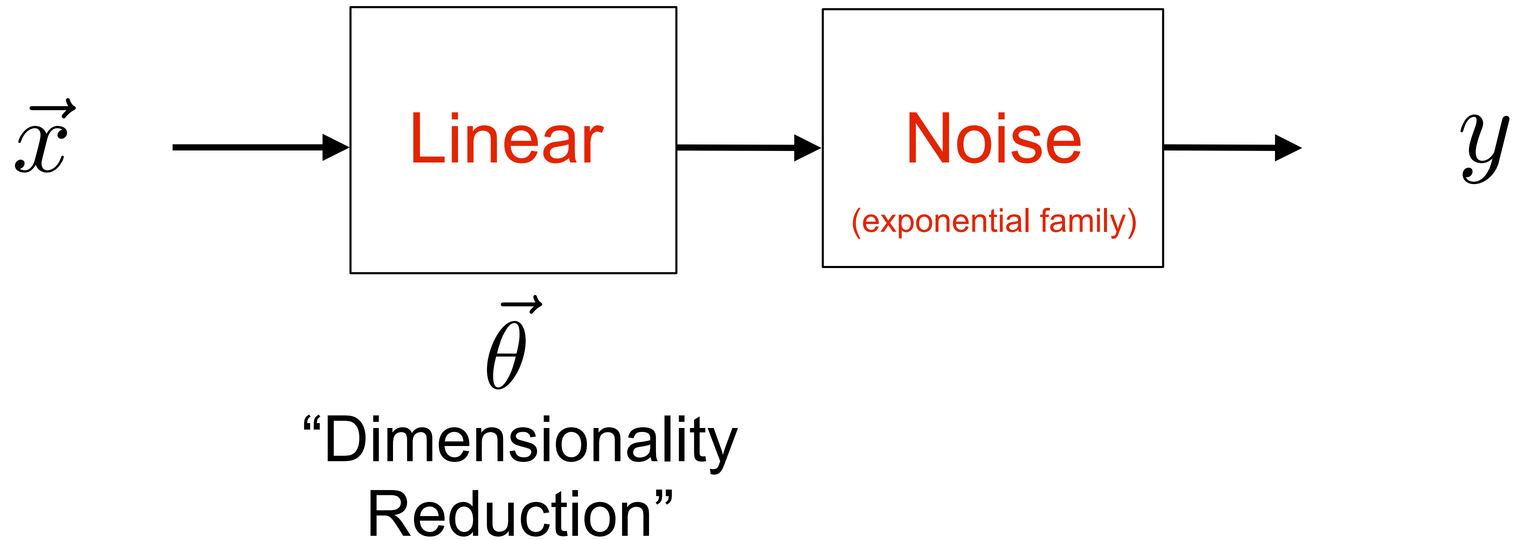
John Nelder: I think probably I do. I suspect we should have found some more fancy name for it that would have stuck and not been confused with the general linear model, although general and generalized are not quite the same. I can see why it might have been better to have thought of something else.

Senn, (2003). *Statistical Science*

Moral:

Be careful when naming your model!

1. General Linear Model

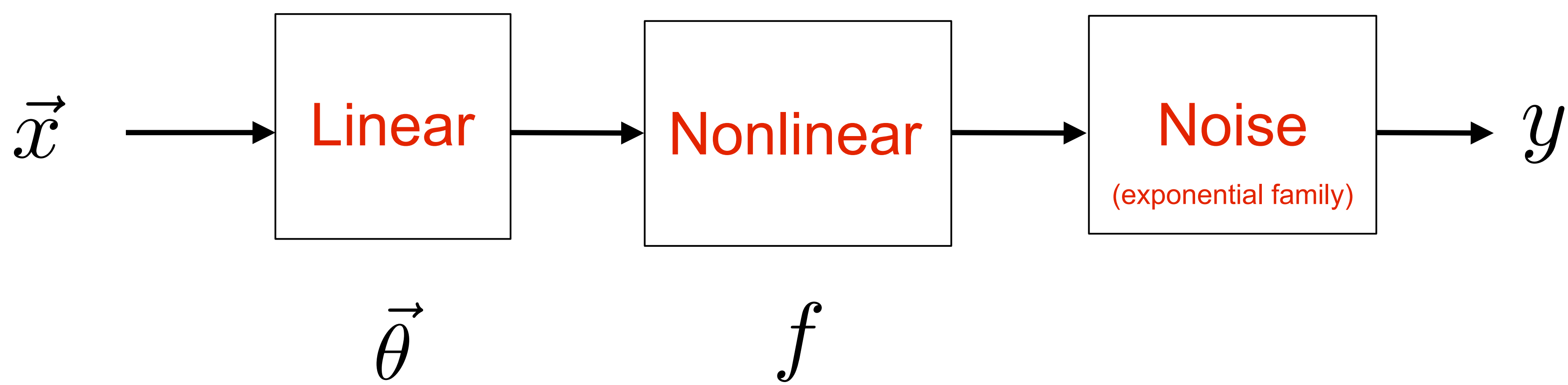


Examples:

1. Gaussian $\quad y = \vec{\theta} \cdot \vec{x} + \epsilon$

2. Poisson $\quad y \sim \mathrm{Poiss}(\vec{\theta} \cdot \vec{x})$

2. Generalized Linear Model

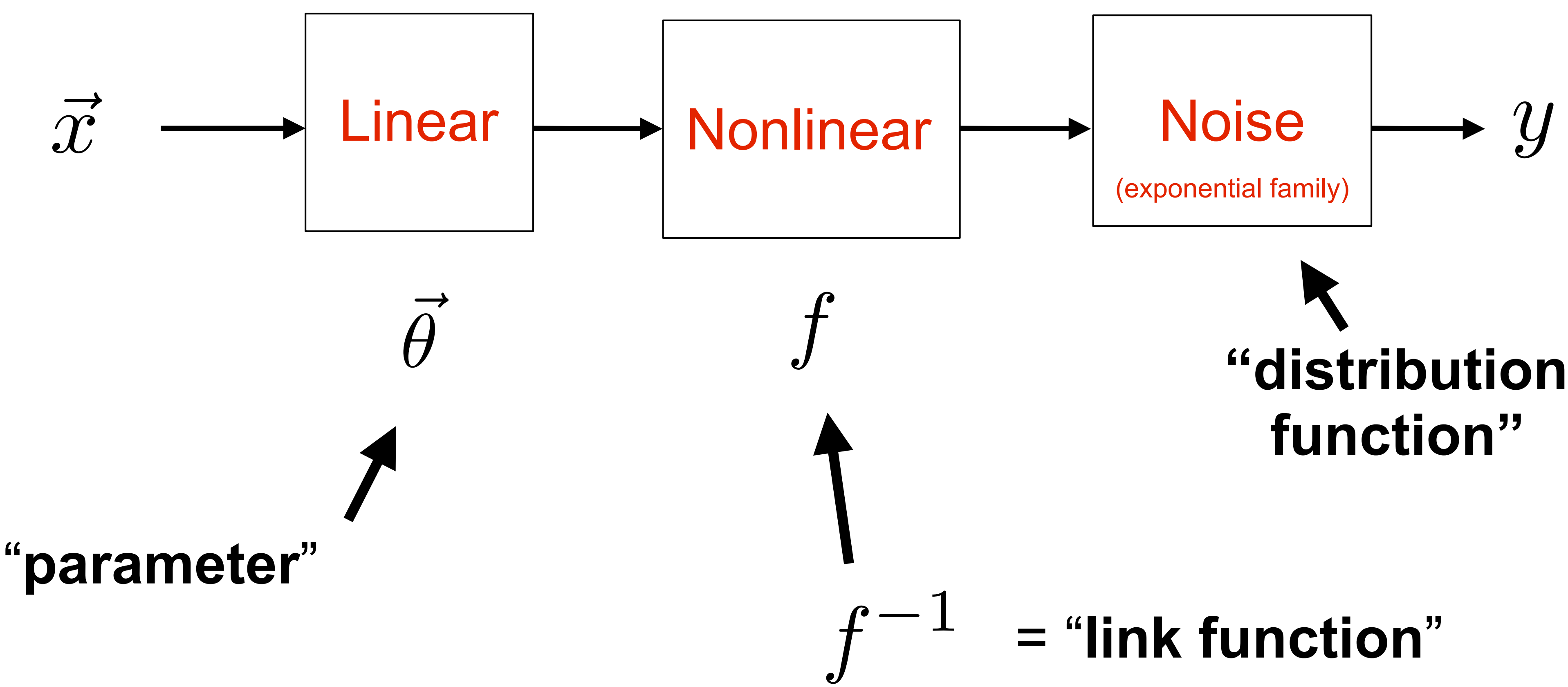


Examples:

1. Gaussian $\quad y = f(\vec{\theta} \cdot \vec{x}) + \epsilon$
2. Poisson $\quad y \sim \mathrm{Poiss}(f(\vec{\theta} \cdot \vec{x}))$

2. Generalized Linear Model

Terminology:



From spike counts to spike trains:

$N(0, \sigma^2)$

response
at time t

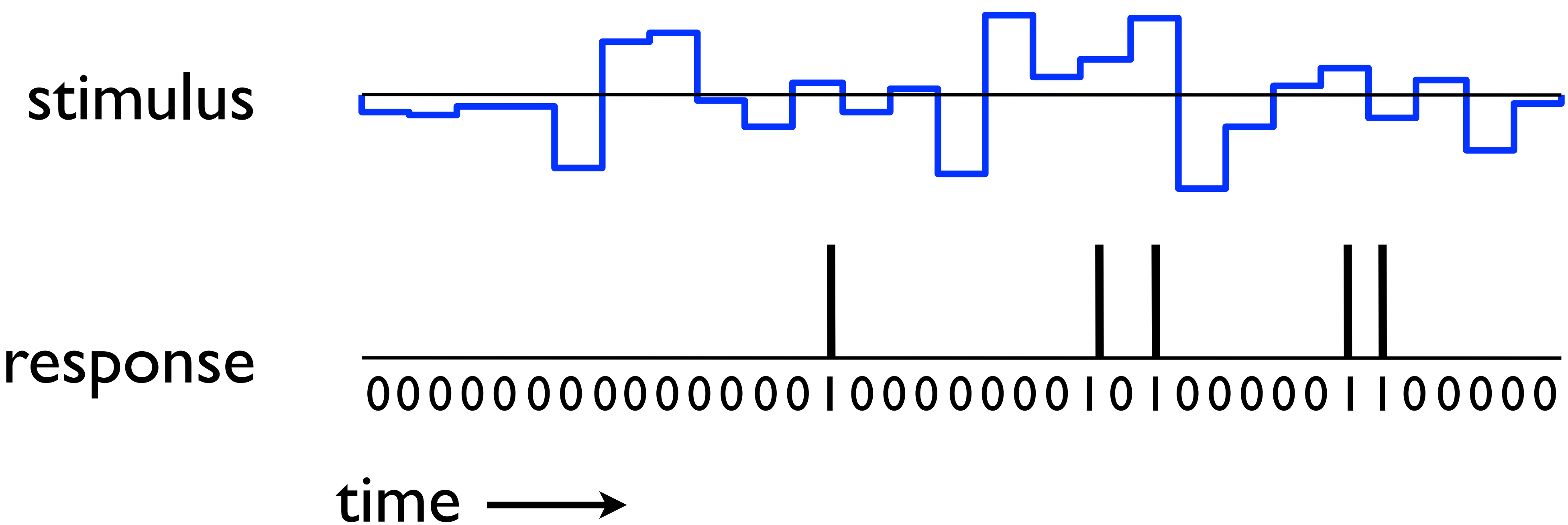
$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \text{noise}$$

linear
filter

vector stimulus
at time t

$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \epsilon_t$$

first idea: linear-Gaussian model!



response at time t

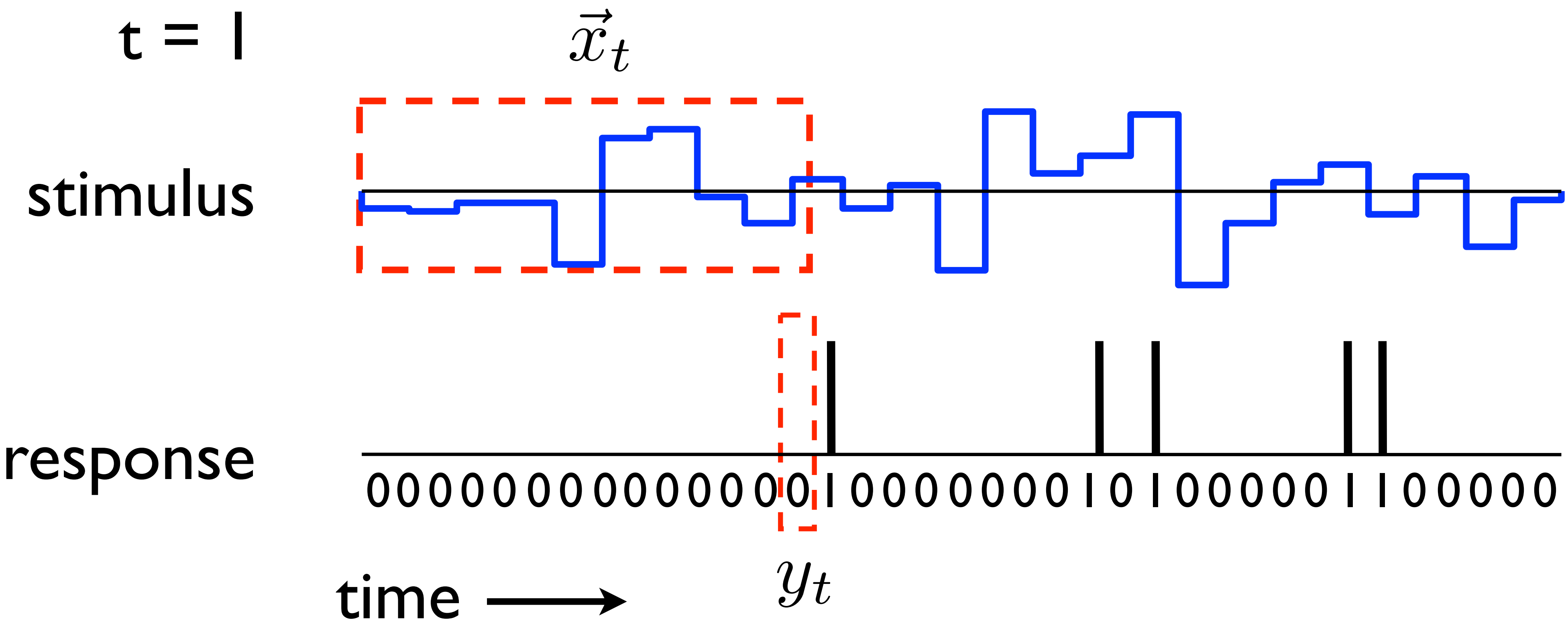
$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \text{noise}$$

$N(0, \sigma^2)$

linear filter

vector stimulus at time t

walk through the data one time bin at a time



response
at time t

$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \text{noise}$$

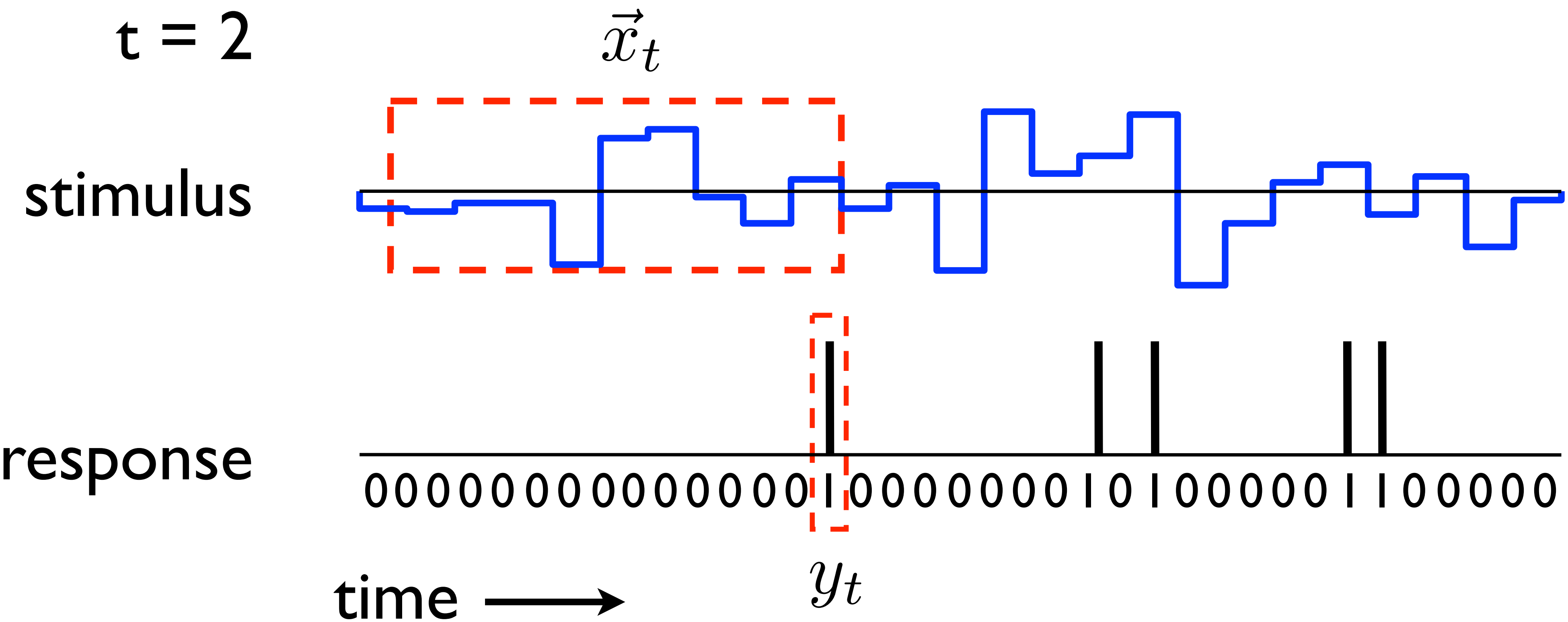
$N(0, \sigma^2)$

linear
filter

vector stimulus
at time t

walk through the data
one time bin at a time

t = 2



response at time t

$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \text{noise}$$

$N(0, \sigma^2)$

linear filter

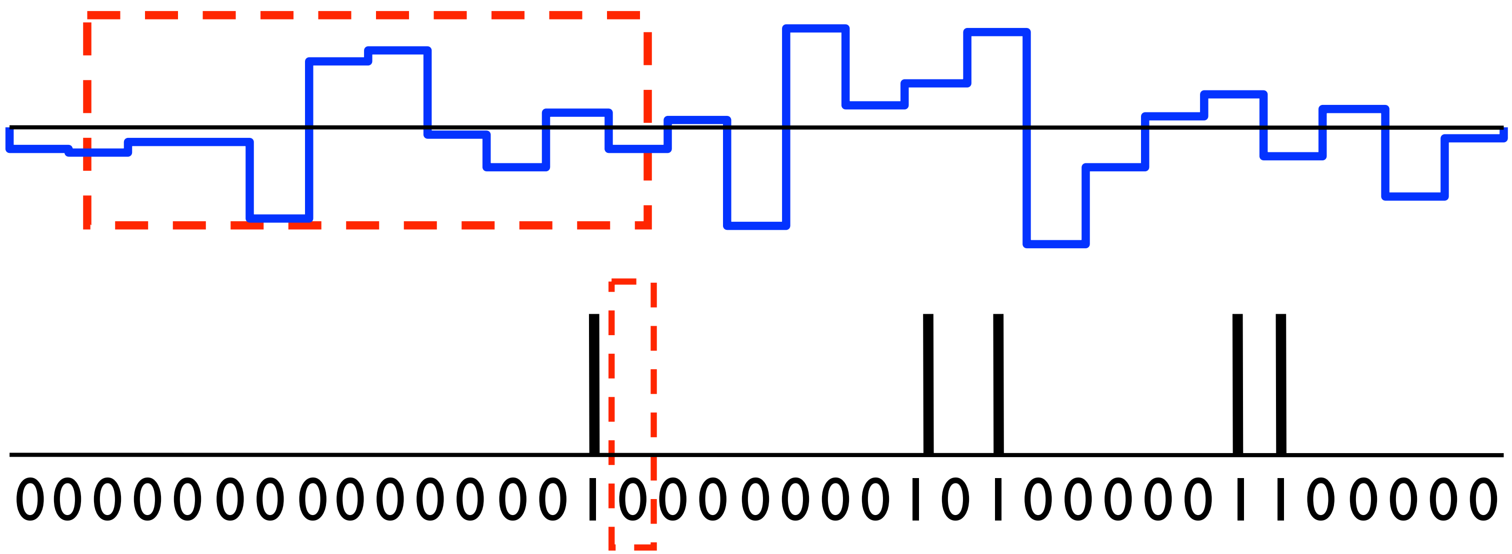
vector stimulus at time t

walk through the data one time bin at a time

t = 3

$\vec{x}_t$

stimulus



response

y_t

time →

response at time t

$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \text{noise}$$

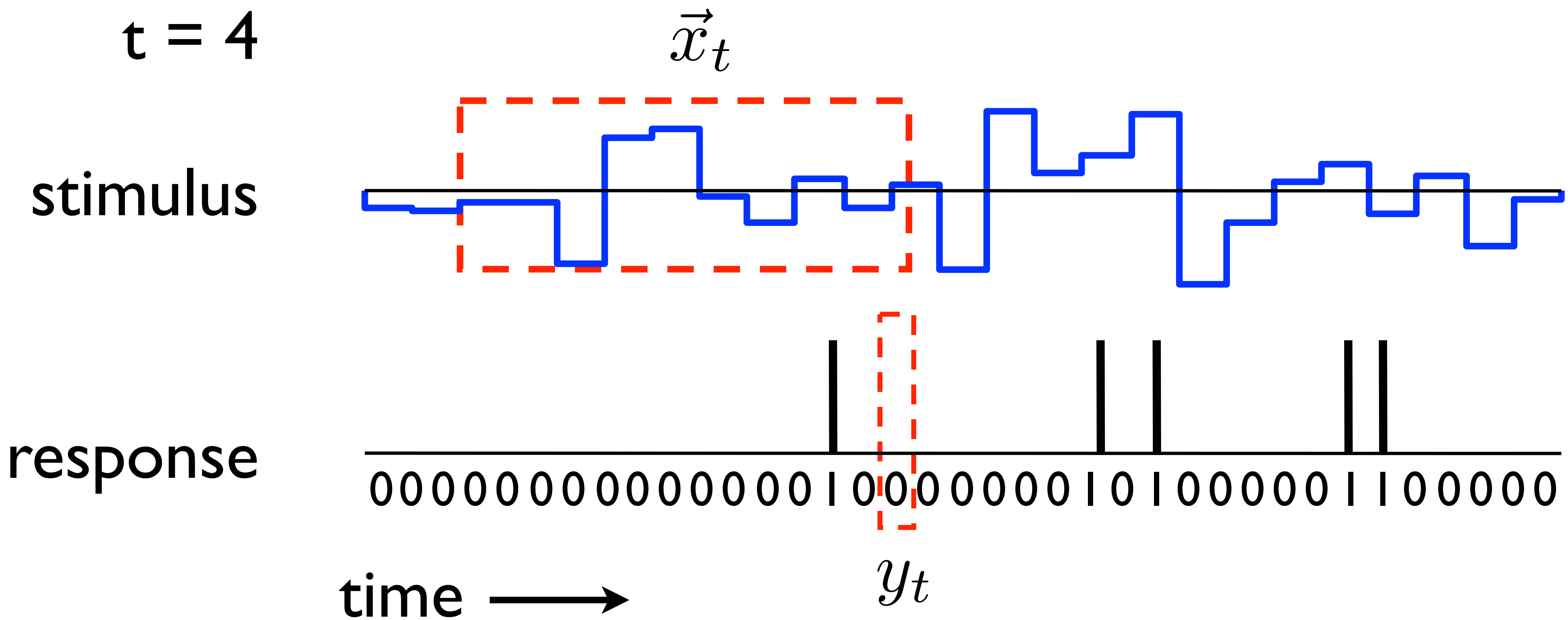
$N(0, \sigma^2)$

linear filter

vector stimulus at time t

walk through the data one time bin at a time

t = 4



response
at time t

$$y_t = \vec{k} \cdot \vec{x}_t \; + \; \text{noise}$$

$N(0, \sigma^2)$

linear
filter

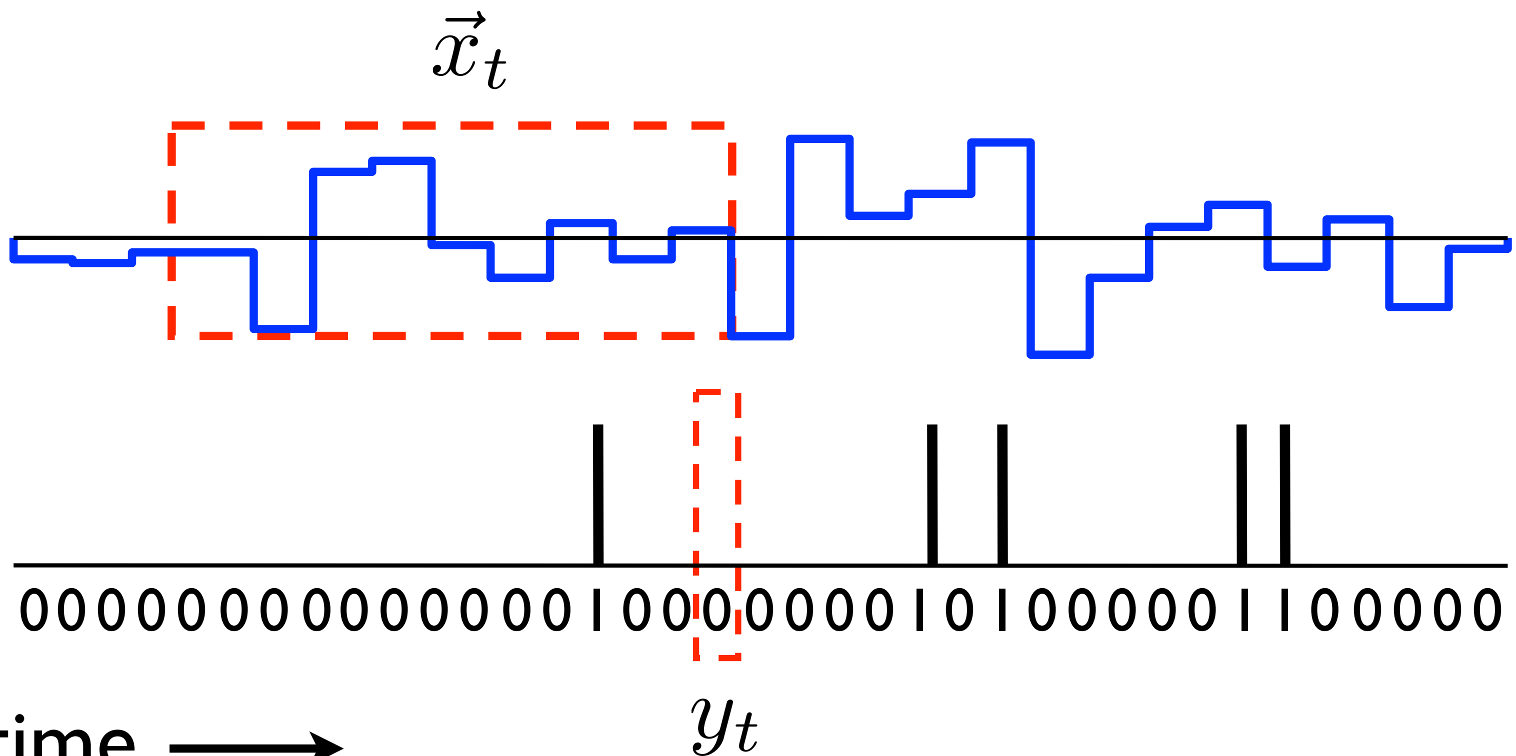
vector stimulus
at time t

walk through the data
one time bin at a time

t = 5

stimulus

response



time ⟶

$$y_t = \vec{k} \cdot \vec{x}_t + \text{noise}$$

response at time t

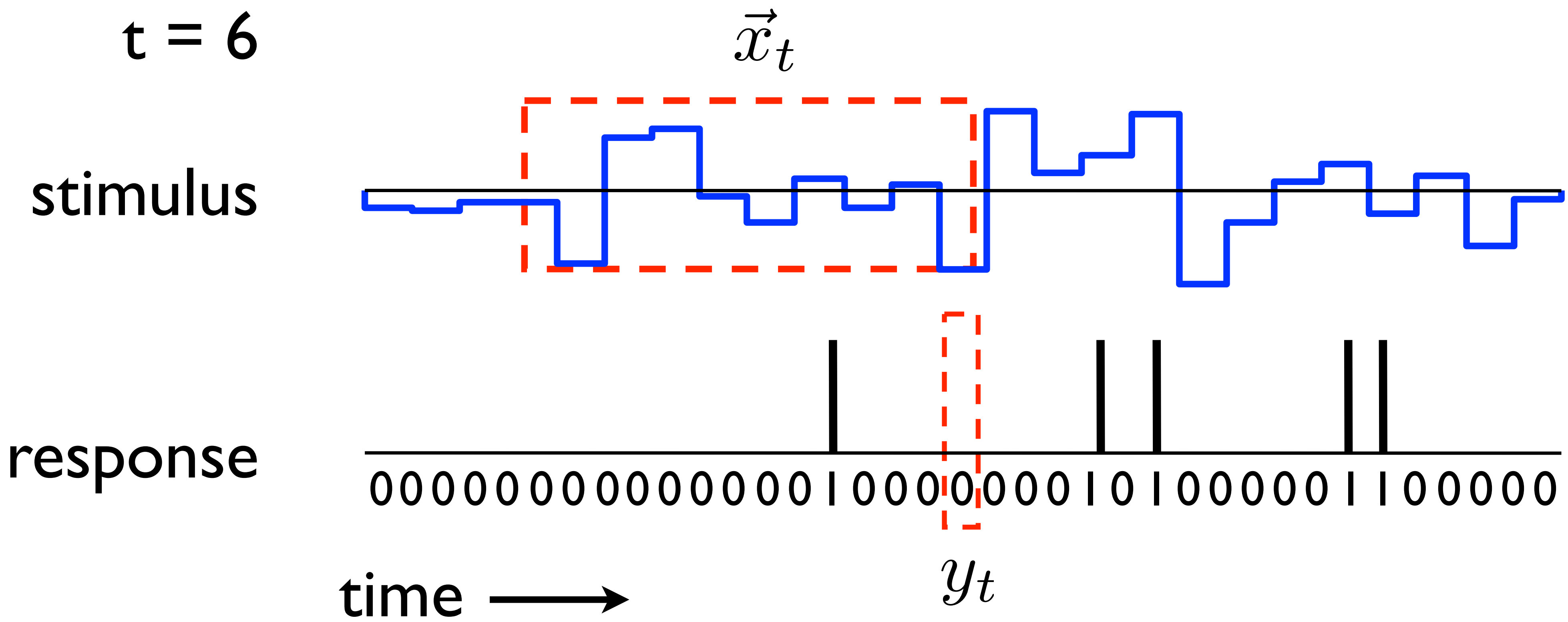
linear filter

vector stimulus at time t

$N(0, \sigma^2)$

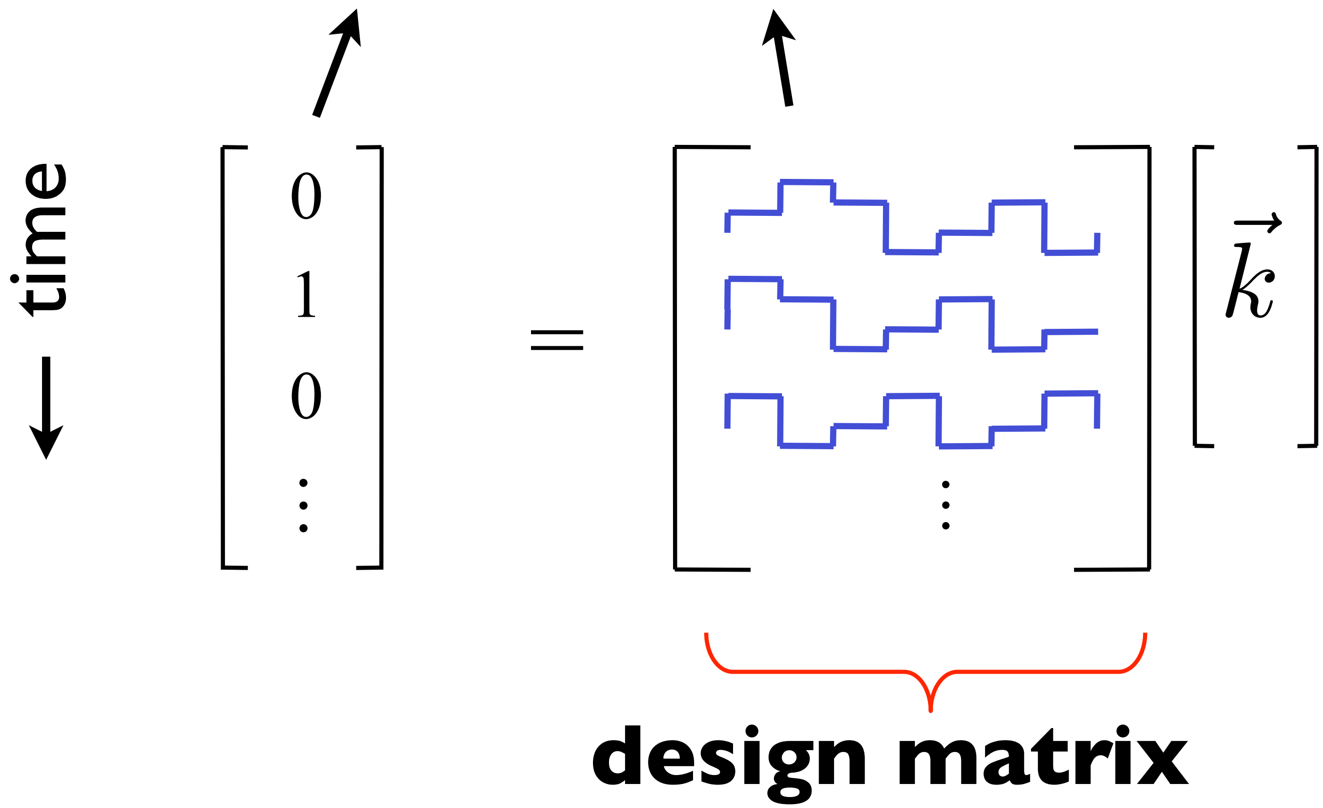
walk through the data one time bin at a time

t = 6



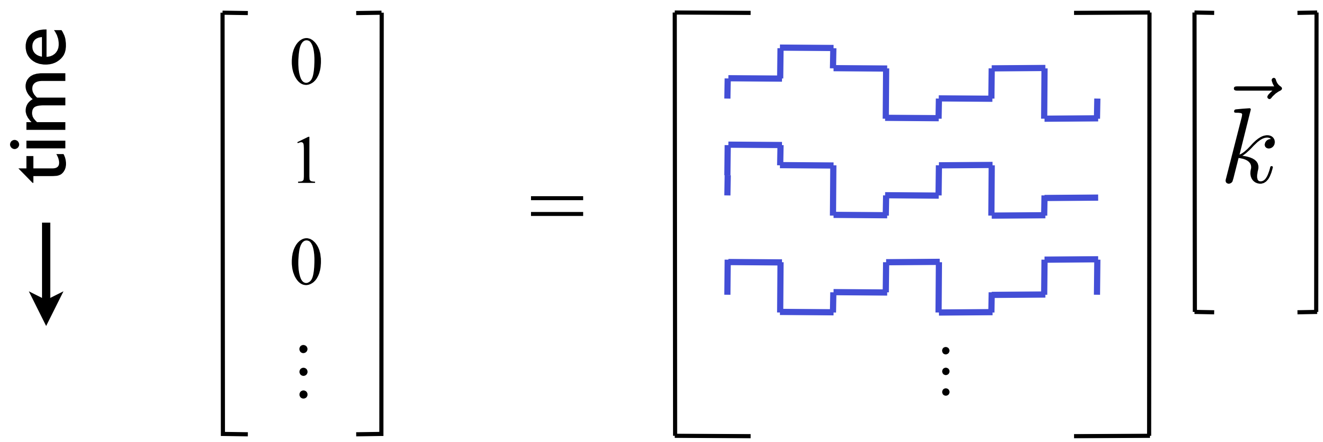
Build up to following matrix version:

$$Y \quad = \quad X\vec{k} \quad + \quad noise$$



Build up to following matrix version:

$$Y = X\vec{k} + noise$$



least squares solution: $$\hat{k} = (\underbrace{X^T X}_{\text{stimulus covariance}})^{-1} \underbrace{X^T Y}_{\text{spike-triggered avg (STA)}}$$

(maximum likelihood estimate for "Linear-Gaussian" GLM)

Formal treatment: scalar version

model: $$y_t = \vec{k} \cdot \vec{x}_t \; + \; \epsilon_t$$

$N(0, \sigma^2)$

Guassian noise with variance σ^2

equivalent to writing: $$y_t | \vec{x}_t, \vec{k} \; \sim \; \mathcal{N}(\vec{x}_t \cdot \vec{k}, \sigma^2)$$

or

$$p(y_t | \vec{x}_t, \vec{k}) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_t - \vec{x}_t \cdot \vec{k})^2}{2\sigma^2}}$$

For entire dataset: $$p(Y | X, \vec{k}) = \prod_{t=1}^{T} p(y_t | \vec{x}_t, \vec{k})$$

(independence across time bins)

$$= (2\pi\sigma^2)^{-\frac{T}{2}} \exp(-\textstyle\sum_{t=1}^{T} \frac{(y_t - \vec{x}_t \cdot \vec{k})^2}{2\sigma^2})$$

$$\log P(Y | X, \vec{k}) = -\textstyle\sum_{t=1}^{T} \frac{(y_t - \vec{x}_t \cdot \vec{k})^2}{2\sigma^2} \; + \; const$$

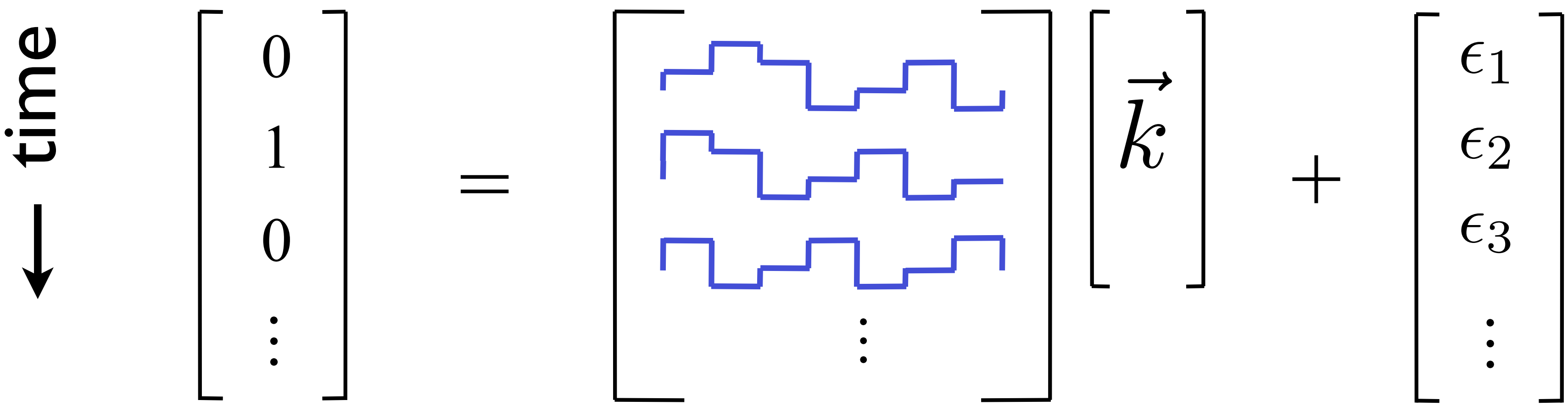
log-likelihood

Formal treatment: vector version

$$Y = X\vec{k} + \vec{\epsilon}$$

$N(0, \sigma^2 I)$

iid Gaussian noise vector



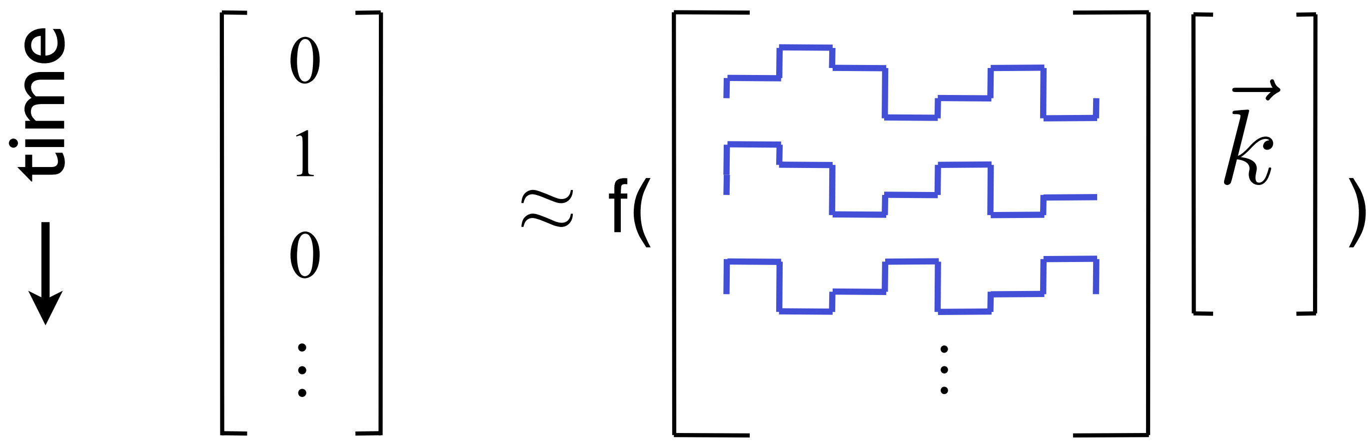
equivalent to writing: $Y|X, \vec{k} \sim \mathcal{N}(X\vec{k}, \sigma^2 I)$

or

$$P(Y|X,\vec{k}) = \frac{1}{|2\pi\sigma^2 I|^{\frac{T}{2}}} \exp\left(-\frac{1}{2\sigma^2}(Y - X\vec{k})^\top (Y - X\vec{k}) \right)$$

Take log, differentiate and set to zero.

But noise is *not* Gaussian!



nonlinearity

Bernoulli GLM:

(coin flipping model, y = 0 or 1)

$$p_t = f(\vec{x}_t \cdot \vec{k})$$

probability of spike at bin t

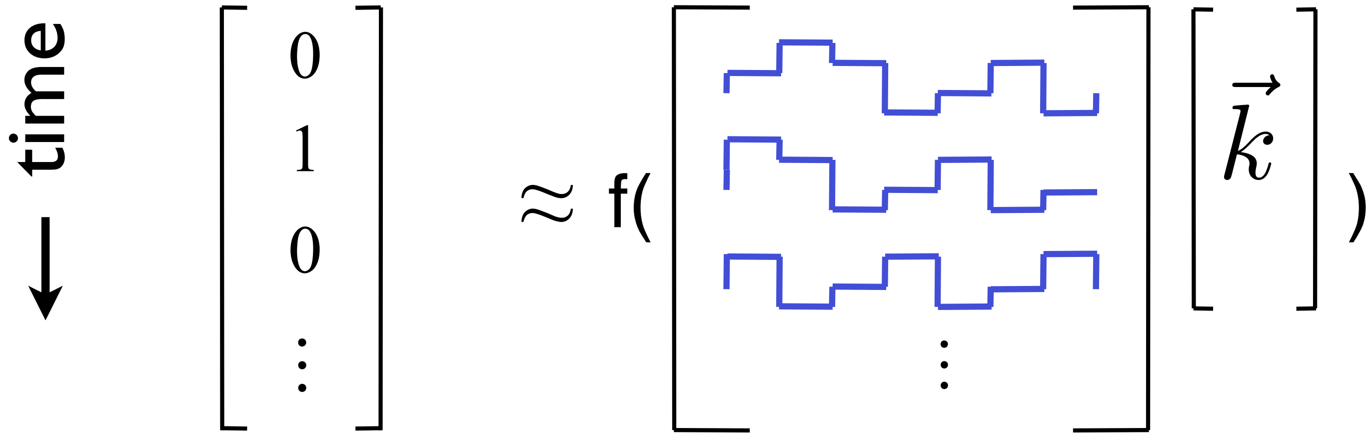
$$p(y_t = 1 | \vec{x}_t) = p_t$$

Equivalent ways of writing: $y_t | \vec{x}_t, \vec{k} \;\sim\; \mathrm{Ber}(f(\vec{x}_t \cdot \vec{k}))$

or $p(y_t | \vec{x}_t, \vec{k}) = f(\vec{x}_t \cdot \vec{k})^{y_t} \left(1 - f(\vec{x}_t \cdot \vec{k})\right)^{1-y_t}$

log-likelihood: $\mathcal{L} = \sum_{t=1}^{T} \left(y_t \log f(\vec{x}_t \cdot \vec{k}) + (1 - y_t) \log(1 - f(\vec{x}_t \cdot \vec{k}))\right)$

Logistic regression



nonlinearity

Bernoulli GLM:

(coin flipping model,
y = 0 or 1)

$$p_t = f(\vec{x}_t \cdot \vec{k})$$

$$p(y_t = 1|\vec{x}_t) = p_t$$

probability of
spike at bin t

Logistic regression:

$$f(x) = \frac{1}{1 + e^{-x}}$$

logistic function

- so logistic regression is a special case of a Bernoulli GLM

Poisson regression

nonlinearity

Poisson GLM:

(integer y≥0)

$$\lambda_t = f(\vec{x}_t \cdot \vec{k})$$ firing rate

$$y_t | \vec{x}_t, \vec{k} \sim \mathrm{Poiss}(\Delta \lambda_t)$$

time bin size

encoding distribution:

$$p(y_t | \vec{x}_t, \vec{k}) = \frac{(\Delta \lambda_t)^{y_t}}{y_t!} e^{-\Delta \lambda_t}$$

log-likelihood:

$$\mathcal{L} = \log p(Y|X, \vec{k}) = \sum_t \Big(y_t \log f(\vec{x}_t \cdot \vec{k}) - f(\vec{x}_t \cdot \vec{k}) \Big) + const$$

$$= Y^\top \log f(X\vec{k}) - \mathbf{1}^\top f(X\vec{k}) + const$$

Summary:

1. "Linear-Gaussian" GLM: $$Y|X,\vec{k} \sim \mathcal{N}(X\vec{k}, \sigma^2 I)$$

$$\hat{k} = (X^T X)^{-1} X^T Y$$

2. Bernoulli GLM: $$y_t|\vec{x}_t, \vec{k} \sim \mathrm{Ber}(f(\vec{x}_t \cdot \vec{k}))$$

$$\mathcal{L} = Y^\top \log f(X\vec{k}) - (1-Y)^\top \log(1 - f(X\vec{k}))$$

3. Poisson GLM: $$y_t|\vec{x}_t, \vec{k} \sim \mathrm{Poiss}(\Delta\lambda_t)$$

$$\mathcal{L} = Y^\top \log f(X\vec{k}) - \mathbf{1}^\top f(X\vec{k})$$