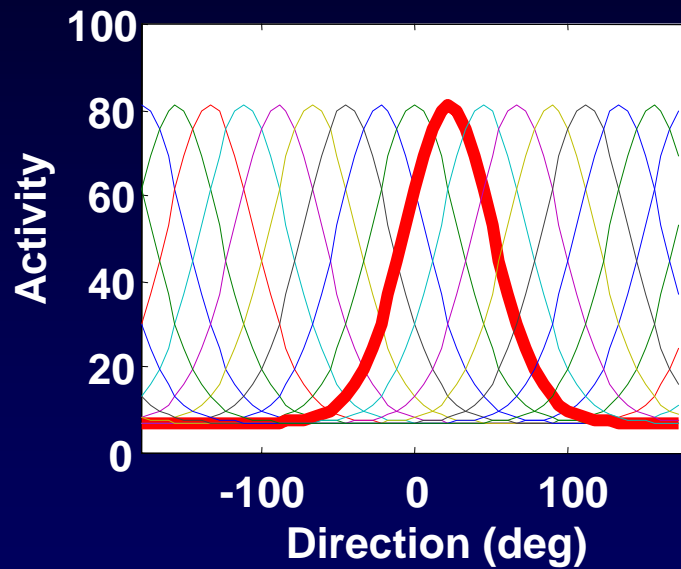


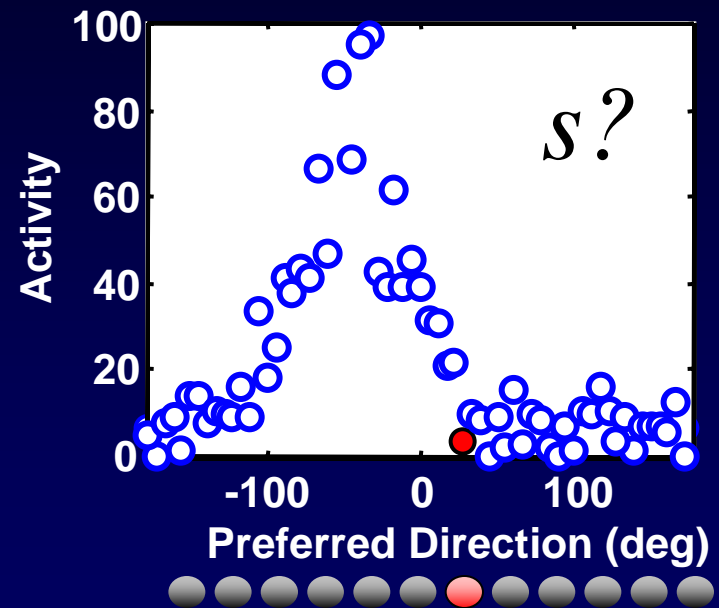
BCS547

Neural Decoding

Population Code



Tuning Curves



Pattern of activity (\mathbf{r})

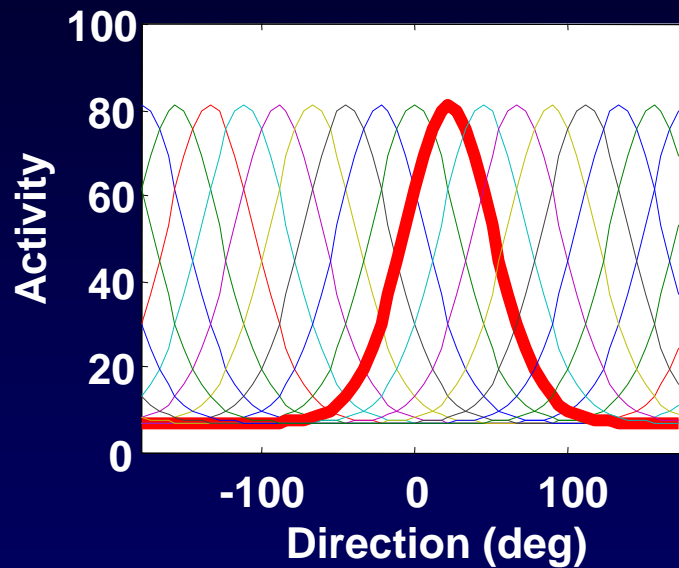
Nature of the problem

In response to a stimulus with unknown orientation s , you observe a pattern of activity \mathbf{r} . What can you say about s given \mathbf{r} ?

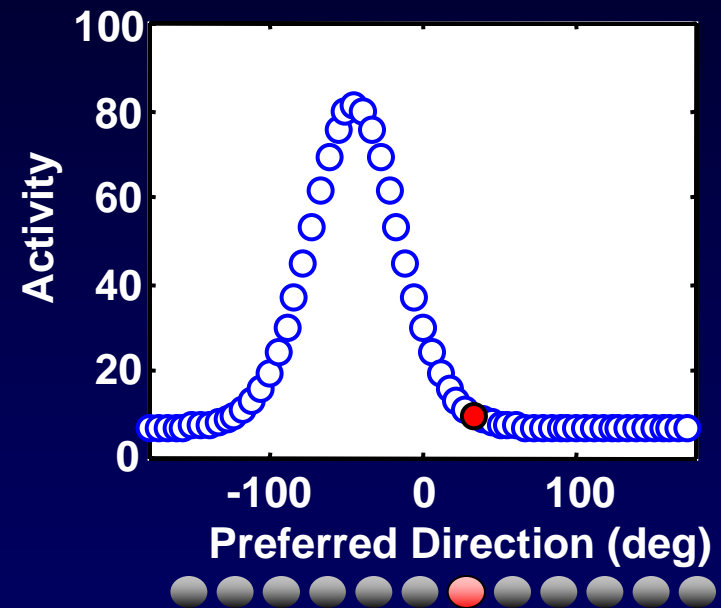
Estimation theory: come up with a single value estimate \hat{s} from \mathbf{r}

Bayesian approach: recover $p(s/\mathbf{r})$ (the posterior distribution)

Maximum Likelihood

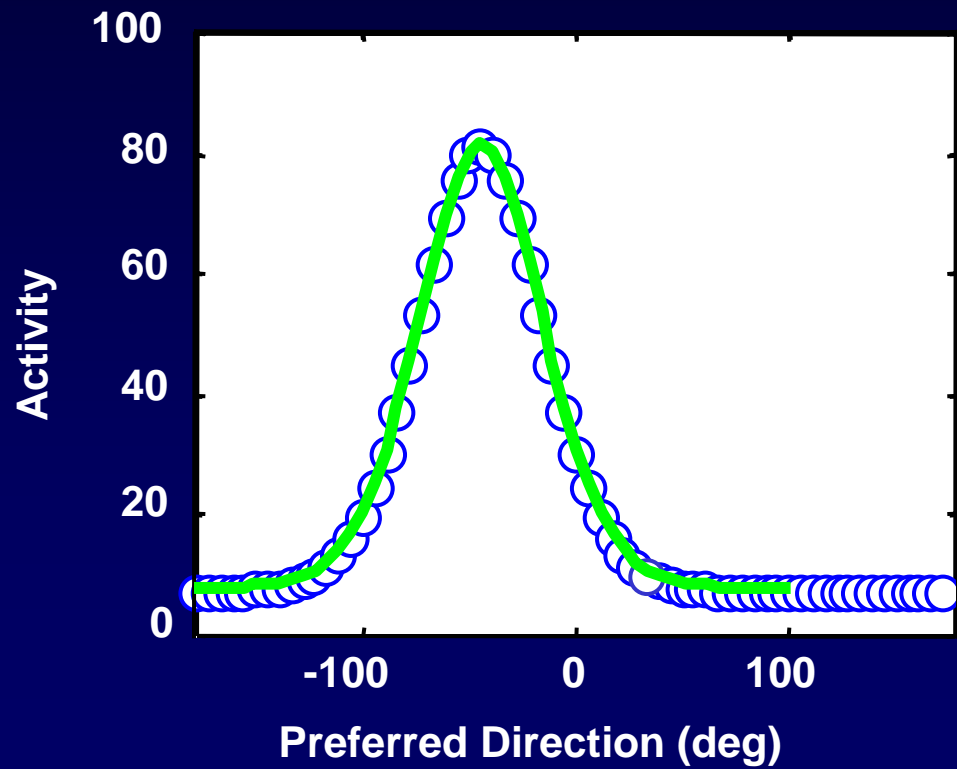


Tuning Curves



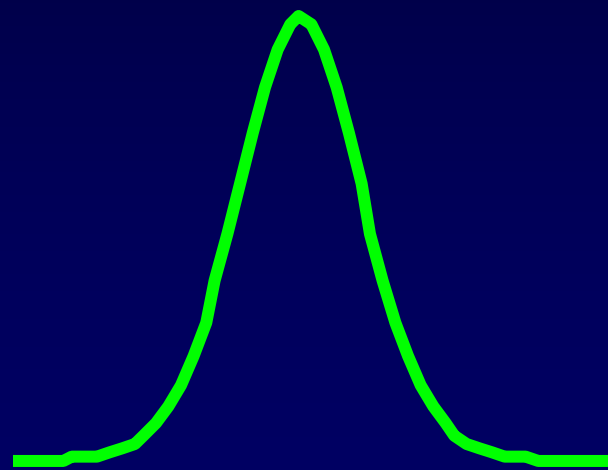
Pattern of activity (\mathbf{r})

Maximum Likelihood

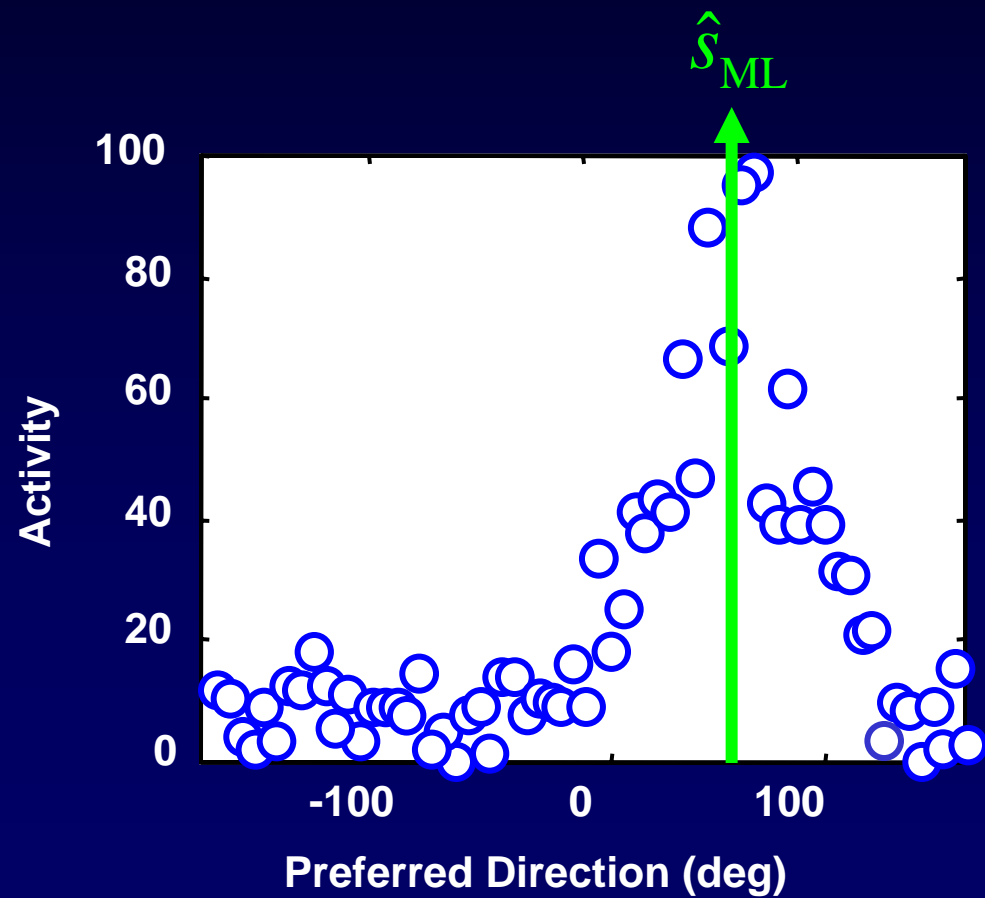


Template

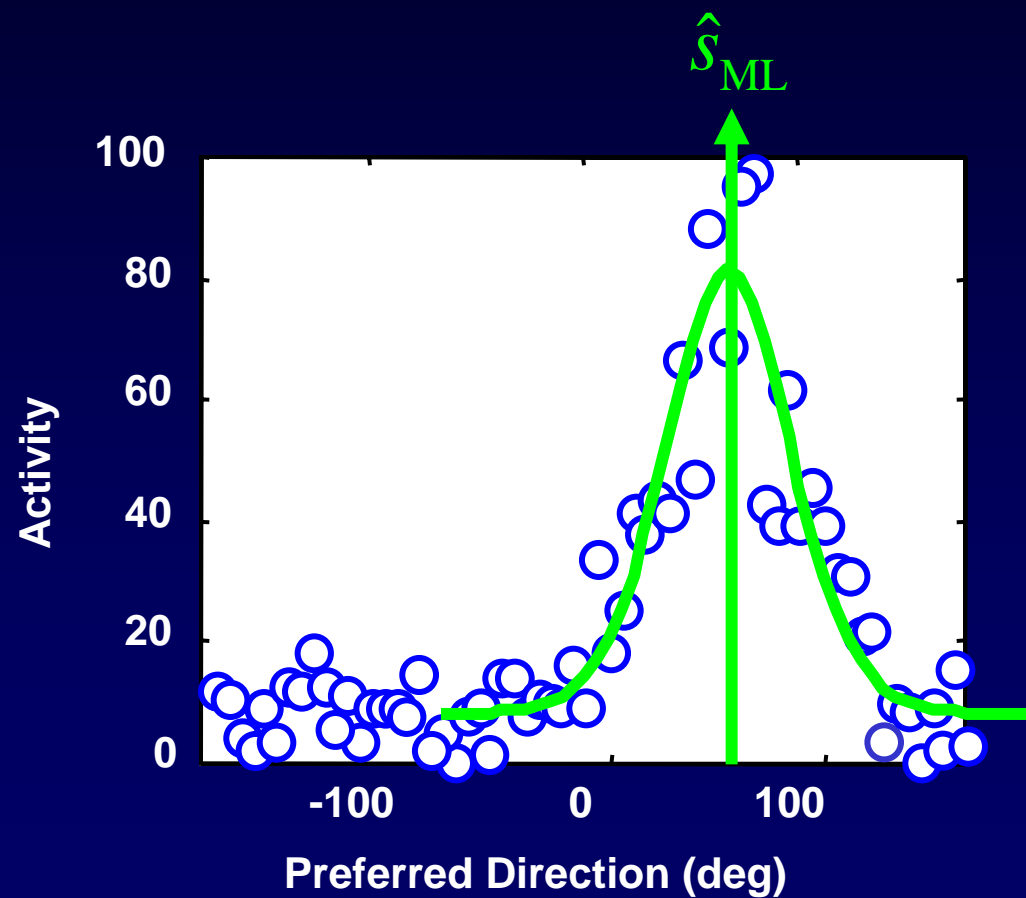
Maximum Likelihood



Template



Maximum Likelihood



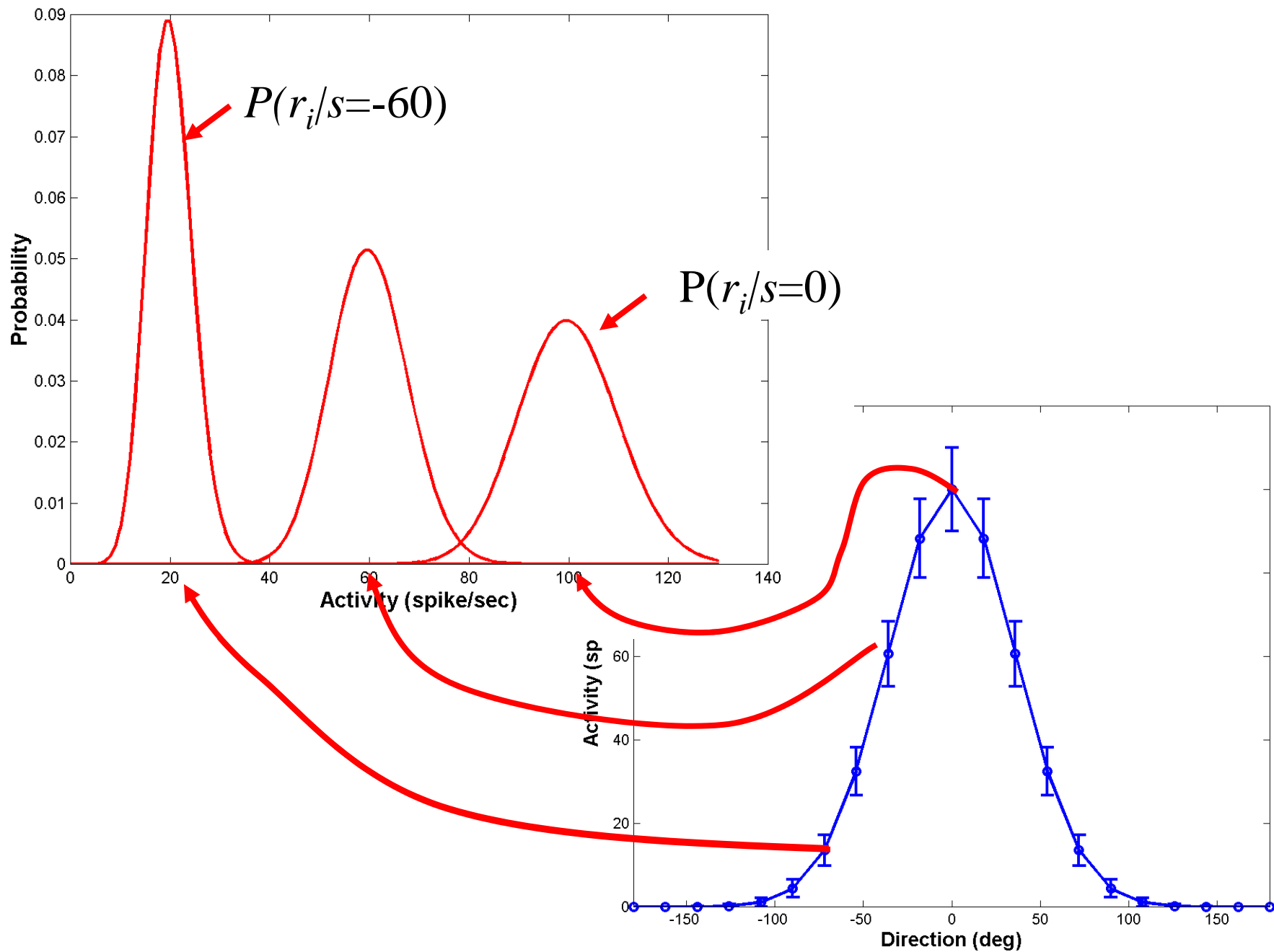
Maximum Likelihood

The maximum likelihood estimate is the value of s maximizing the likelihood $p(\mathbf{r}/s)$. Therefore, we seek \hat{s} such that:

$$\hat{s}_{\text{ML}} = \arg \max_s P(\mathbf{r} | s)$$

Noise distribution





Maximum Likelihood

The maximum likelihood estimate is the value of s maximizing the likelihood $p(s/r)$. Therefore, we seek \hat{s} such that:

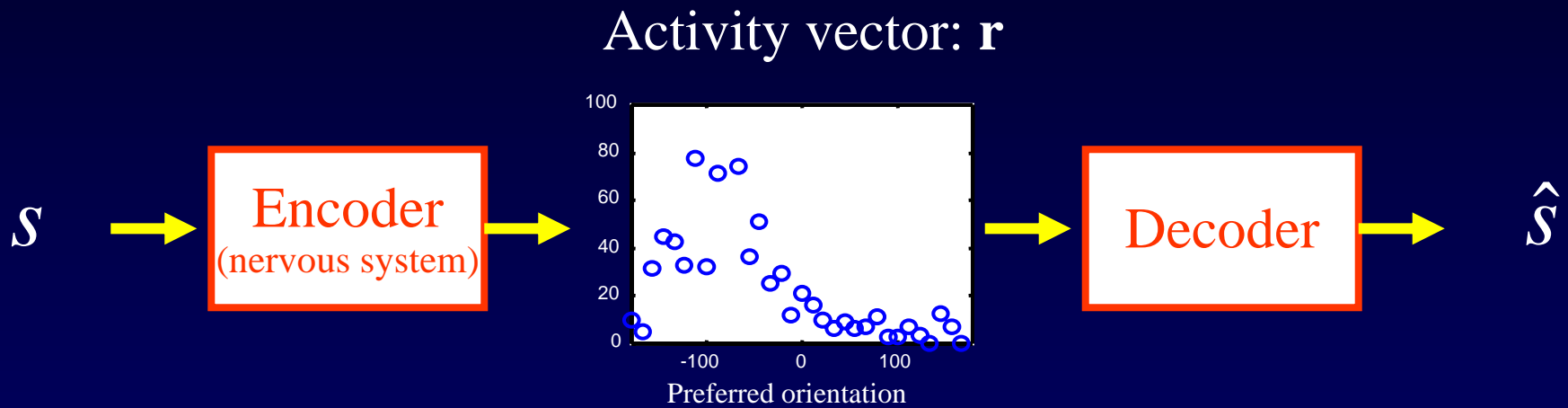
$$\hat{s}_{\text{ML}} = \arg \max_s P(\mathbf{r} | s)$$

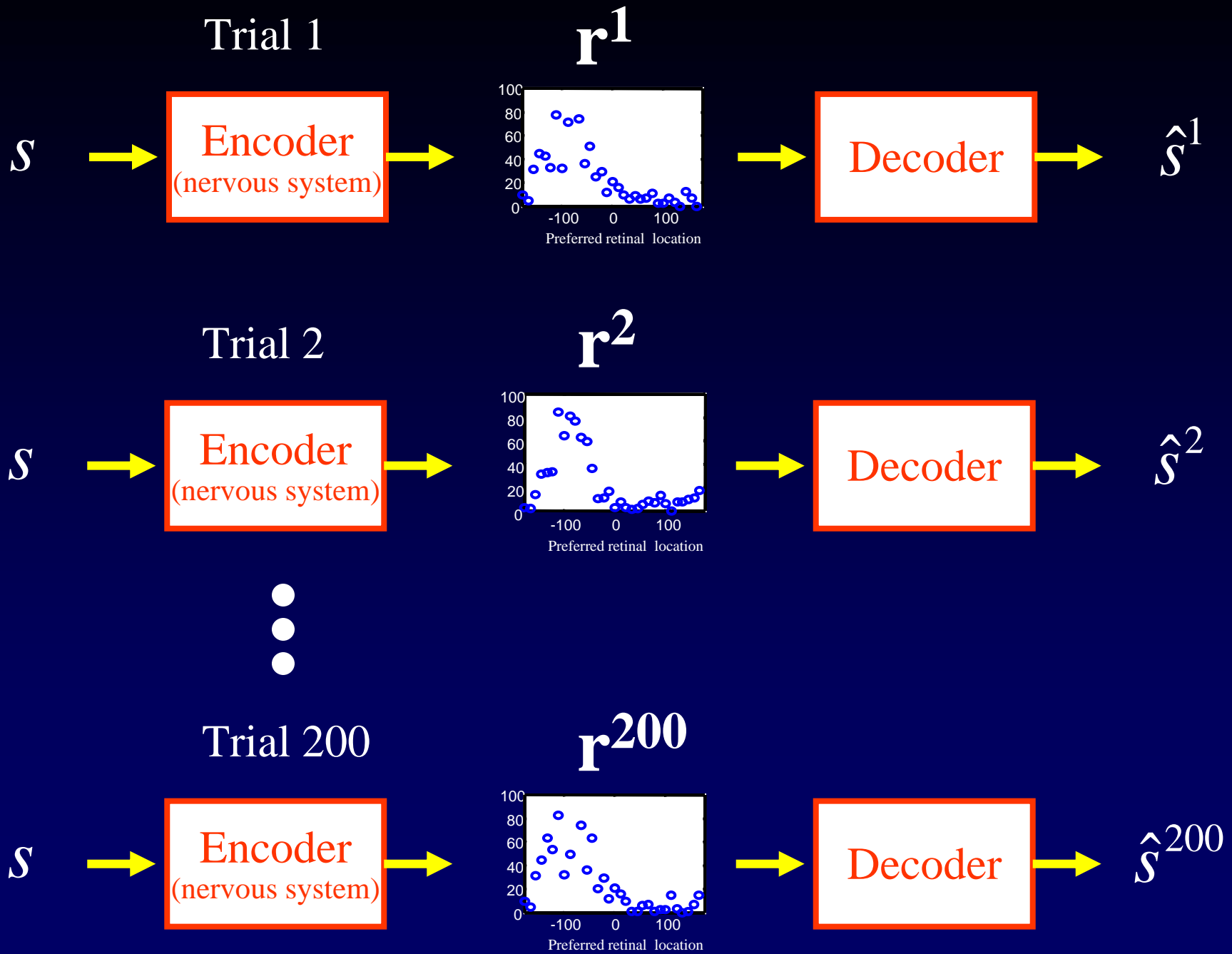
Noise distribution



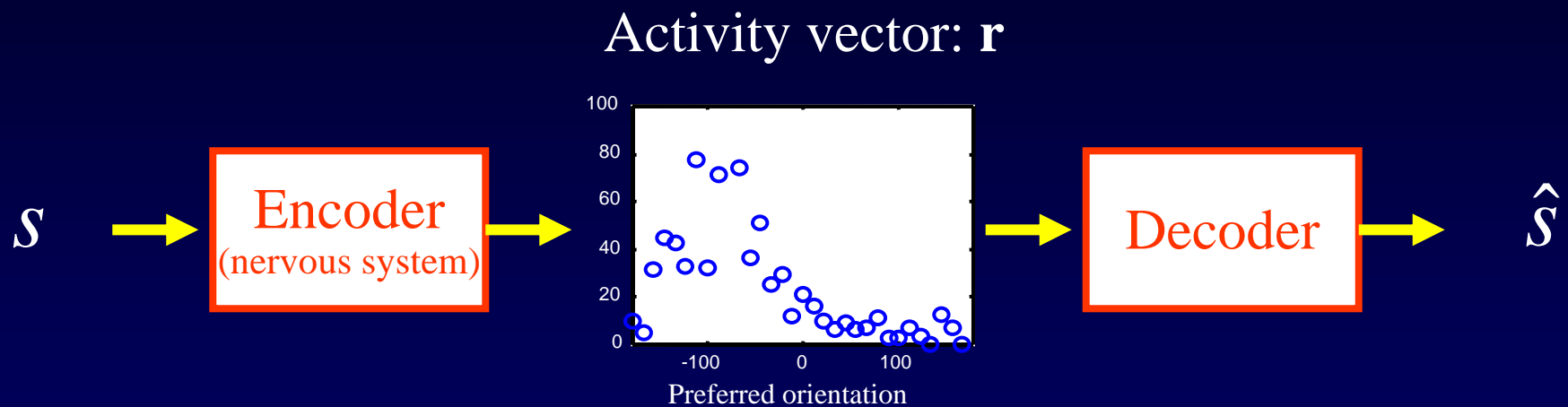
\hat{s}_{ML} is unbiased and efficient.

Estimation Theory





Estimation Theory



If $E[\hat{s} | s] = s$, the estimate is said to be **unbiased**

If $\sigma_{\hat{s}|s}^2$ is as small as possible, the estimate is said to be **efficient**

Estimation theory

- A common measure of decoding performance is the mean square error between the estimate and the true value

$$\text{MSE} = E\left[(\hat{s} - s)^2 \mid s\right]$$

- This error can be decomposed as:

$$\begin{aligned}\text{MSE} &= \left(E[\hat{s} \mid s] - s\right)^2 + \sigma_{\hat{s}|s}^2 \\ &= \textit{bias}^2 + \sigma_{\hat{s}|s}^2\end{aligned}$$

Efficient Estimators

The smallest achievable variance for an unbiased estimator is known as the Cramer-Rao bound, σ_{CR}^2 .

An efficient estimator is such that

$$\sigma_{\hat{s}|s}^2 = \sigma_{\text{CR}}^2$$

In general :

$$\sigma_{\hat{s}|s}^2 > \sigma_{\text{CR}}^2$$

Fisher Information

Fisher information is defined as:

$$I(s) = \frac{1}{\sigma_{CR}^2}$$

and it is equal to:

$$I(s) = -E \left[\frac{\partial^2 \ln P(\mathbf{r} | s)}{\partial s^2} \right]$$

where $p(\mathbf{r}|s)$ is the distribution of the neuronal noise.

Fisher Information

$$I = -E \left[\frac{\partial^2 \ln \mathbf{P}(\mathbf{r} | s)}{\partial s^2} \right]$$

$$\mathbf{P}(\mathbf{r} | s) = \prod_{i=1}^n \mathbf{P}(r_i = k_i | s) = \prod_{i=1}^n \frac{f_i(s)^{k_i} e^{-f_i(s)}}{k_i!}$$

$$\ln \mathbf{P}(\mathbf{r} | s) = \sum_{i=1}^n k_i \ln f_i(s) - f_i(s) - \ln(k_i!)$$

$$\frac{\partial \ln \mathbf{P}(\mathbf{r} | s)}{\partial s} = \sum_{i=1}^n \frac{k_i f_i'(s)}{f_i(s)} - f_i'(s)$$

$$\frac{\partial^2 \ln \mathbf{P}(\mathbf{r} | s)}{\partial s^2} = \sum_{i=1}^n \left[-\frac{k_i f_i'(s)^2}{f_i(s)^2} + \frac{k_i f_i''(s)}{f_i(s)} - f_i''(s) \right]$$

$$-E \left[\frac{\partial^2 \ln \mathbf{P}(\mathbf{r} | s)}{\partial s^2} \right] = \sum_{i=1}^n \frac{f_i(s) f_i'(s)^2}{f_i(s)^2} - \frac{f_i(s) f_i''(s)}{f_i(s)} + f_i''(s)$$

$$I = \sum_{i=1}^n \frac{f_i'(s)^2}{f_i(s)}$$

Fisher Information

- For one neuron with Poisson noise

$$I_i(s) = \frac{f_i'(s)^2}{f_i(s)} \propto d'^2$$

- For n independent neurons :

$$I(s) = \sum_i \frac{f_i'(s)^2}{f_i(s)}$$

Large slope is good!

The more neurons, the better!

Small variance is good!

Fisher Information and Tuning Curves

- Fisher information is maximum where the slope is maximum
- This is consistent with adaptation experiments

Fisher Information

- In 1D, Fisher information decreases as the width of the tuning curves increases
- In 2D, Fisher information does not depend on the width of the tuning curve
- In 3D and above, Fisher information increases as the width of the tuning curves increases
- WARNING: this is true for independent gaussian noise.

Ideal observer

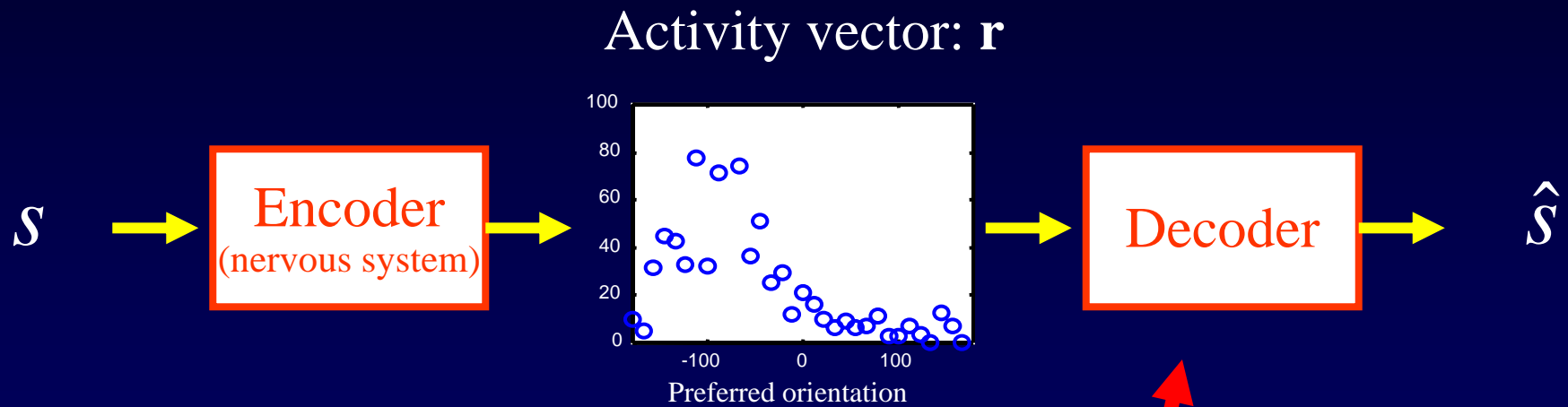
The discrimination threshold of an ideal observer, δs , is proportional to the variance of the Cramer-Rao Bound.

$$\delta s \propto \sigma_{CR}$$

In other words, an efficient estimator is an ideal observer.

- An ideal observer is an observer that can recover all the Fisher information in the activity (easy link between Fisher information and behavioral performance)
- If all distributions are gaussians, Fisher information is the same as Shannon information.

Estimation theory



Other examples of decoders

Voting Methods

Optimal Linear Estimator

$$\hat{s} = \sum_i w_i r_i$$

Linear Estimators

$$\mathbf{X} = \{x_1, \dots, x_n\}$$

$$\mathbf{Y} = \{y_1, \dots, y_n\}$$

$$y^* = ax + b$$

$$E = \frac{1}{2} \sum_{i=1}^n (y_i^* - y_i)^2$$

$$= \frac{1}{2} \sum_{i=1}^n (ax_i + b - y_i)^2$$

$$\frac{\partial E}{\partial b} = \sum_{i=1}^n (ax_i + b - y_i)$$

$$\frac{\partial E}{\partial b} = 0$$

$$\sum_{i=1}^n (ax_i + b - y_i) = 0$$

$$b = \frac{1}{n} \sum_{i=1}^n (y_i - ax_i)$$

$$b = \langle y \rangle - a \langle x \rangle$$

$$y^* - \langle y \rangle = a(x - \langle x \rangle)$$

$$y_0^* = ax_0$$

Linear Estimators

$$y^* = ax$$

$$E = \frac{1}{2} \sum_{i=1}^n (y_i^* - y_i)^2$$
$$= \frac{1}{2} \sum_{i=1}^n (ax_i - y_i)^2$$

$$\frac{\partial E}{\partial a} = \sum_{i=1}^n x_i (ax_i - y_i)$$

$$\frac{\partial E}{\partial a} = 0$$

$$\sum_{i=1}^n x_i (ax_i - y_i) = 0$$

$$a = \frac{\sum_{i=1}^n x_i y_i}{\sum_{i=1}^n x_i^2} = \frac{C_{xy}}{\sigma_x^2}$$

Linear Estimators

$$\mathbf{X} = \begin{Bmatrix} x_1^1 & \dots & x_1^n \\ \dots & \dots & \dots \\ x_m^1 & \dots & x_m^n \end{Bmatrix} m \times n$$

$$\mathbf{Y} = \begin{Bmatrix} y_1^1 & \dots & y_1^n \\ \dots & \dots & \dots \\ y_p^1 & \dots & y_p^n \end{Bmatrix} p \times n$$

$$\mathbf{y}_i = \begin{Bmatrix} y_1^i \\ \dots \\ y_p^i \end{Bmatrix} p \times 1$$

$$\mathbf{y}^* = \mathbf{W}^T \mathbf{x}$$

$$\mathbf{W}^T \sim p \times m$$

$$E = \frac{1}{2} \sum_{i=1}^n \|\mathbf{y}_i^* - \mathbf{y}_i\|^2$$

$$n \gg mp$$

$$\mathbf{W} = \mathbf{C}_{XX}^{-1} \mathbf{C}_{XY} = [\mathbf{X}\mathbf{X}^T]^{-1} \mathbf{X}\mathbf{Y}^T$$

$$m \times p = m \times m \cdot m \times p$$

$$\mathbf{W}^T = \begin{bmatrix} \frac{C_{x_1 y}}{\sigma_{x_1}^2} & \dots & \frac{C_{x_m y}}{\sigma_{x_m}^2} \end{bmatrix}$$

$$\mathbf{y}^* = \sum_{i=1}^m \frac{C_{x_i y}}{\sigma_{x_i}^2} x_i$$

X and Y must be zero mean

Trust cells that have small variances
and large covariances

Voting Methods

Optimal Linear Estimator

$$\hat{s} = \sum_i w_i r_i = \mathbf{W}^T \mathbf{r}, \mathbf{W} = [\mathbf{C}_{rr}]^{-1} \mathbf{C}_{rs}$$

Voting Methods


Optimal Linear Estimator

$$\hat{s} = \sum_i w_i r_i = \mathbf{W}^T \mathbf{r}, \mathbf{W} = [\mathbf{C}_{rr}]^{-1} \mathbf{C}_{rs}$$

Center of Mass

$$\hat{s} = \frac{\sum_i r_i s_i}{\sum_j r_j} = \sum_i s_i \frac{r_i}{\sum_j r_j}$$

Linear in $r_i / \sum_j r_j$
Weights set to s_i



Center of Mass/Population Vector

- The center of mass is optimal (unbiased and efficient) iff: The tuning curves are gaussian with a zero baseline, uniformly distributed and the noise follows a Poisson distribution
- In general, the center of mass has a large bias and a large variance

Voting Methods

Optimal Linear Estimator

$$\hat{s} = \sum_i w_i r_i = \mathbf{W}^T \mathbf{r}, \mathbf{W} = [\mathbf{C}_{rr}]^{-1} \mathbf{C}_{rs}$$

Center of Mass

$$\hat{s} = \frac{\sum_i r_i s_i}{\sum_i r_i}$$

Population Vector

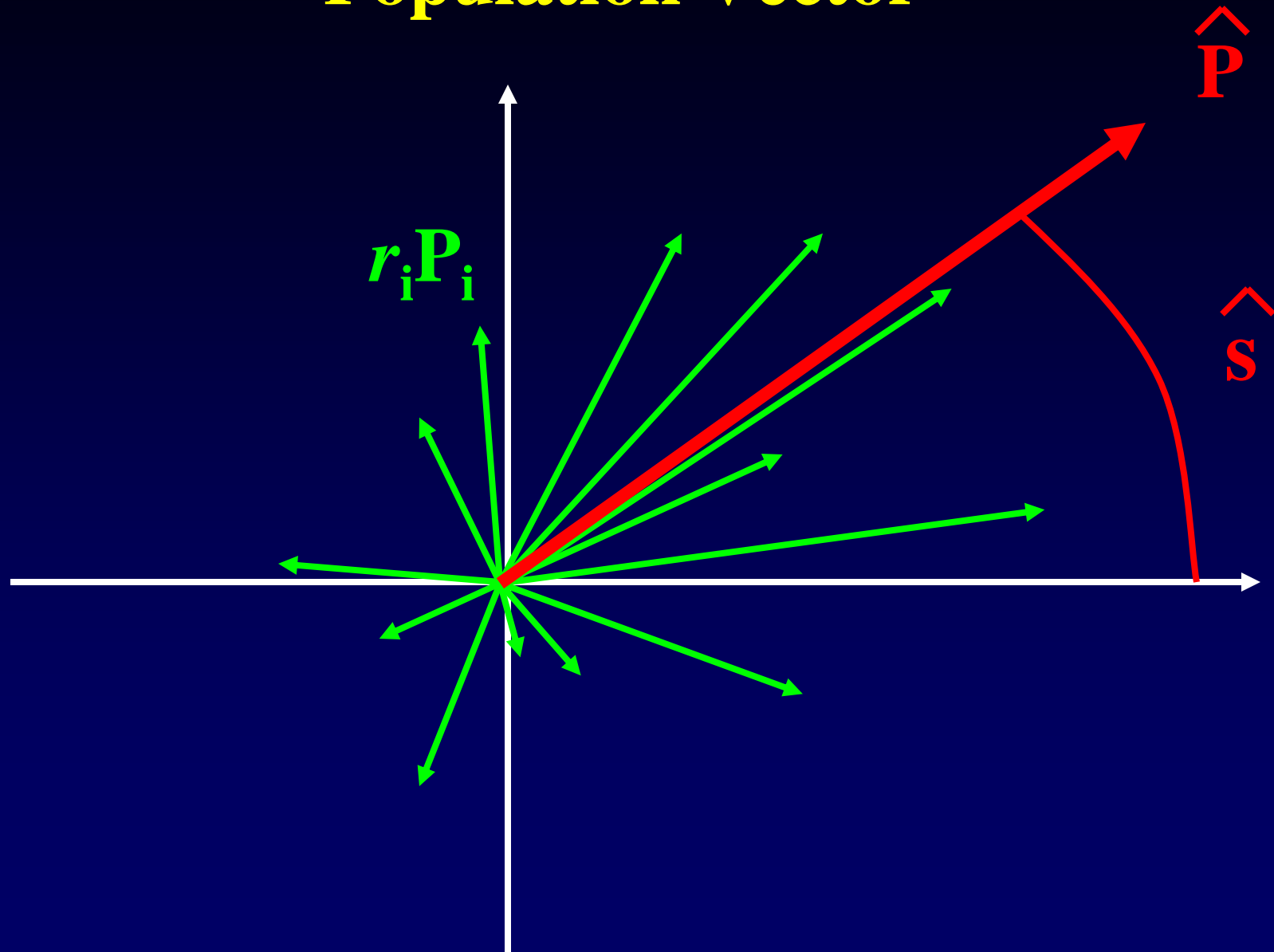
$$\hat{\mathbf{P}} = \sum_i r_i \mathbf{P}_i = \sum_i \mathbf{P}_i r_i$$

Linear in r_i
Weights set to \mathbf{P}_i

$$\hat{s} = \text{angle}(\hat{\mathbf{P}})$$

Nonlinear step

Population Vector



Population Vector

$$\hat{\mathbf{P}} = \sum_i r_i \mathbf{P}_i = \begin{bmatrix} p_1^1 & \cdots & p_m^1 \\ p_1^2 & \cdots & p_m^2 \\ \vdots & \ddots & \vdots \\ p_1^m & \cdots & p_m^m \end{bmatrix} \begin{bmatrix} r_1 \\ \vdots \\ r_m \end{bmatrix} = \mathbf{W}_P^T \mathbf{r}$$

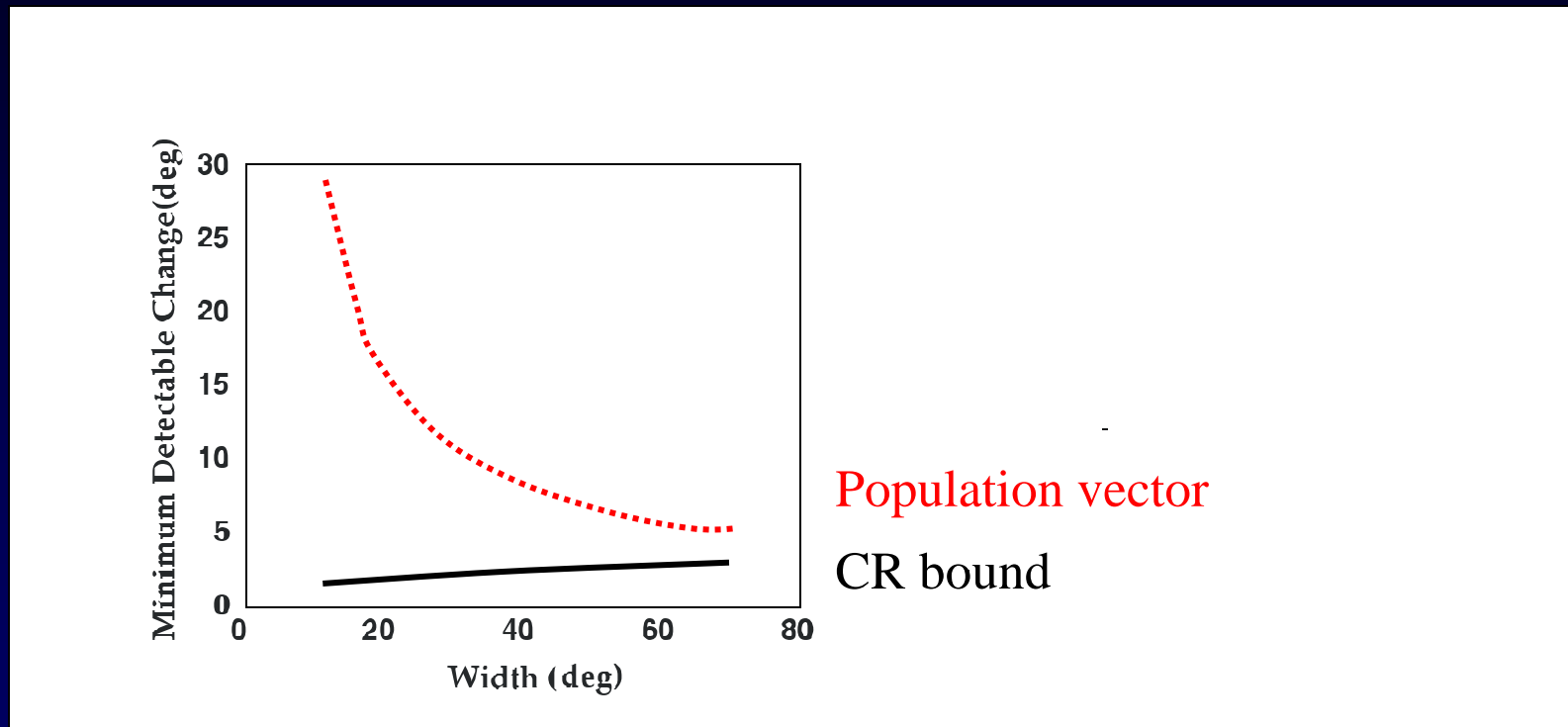
$$\mathbf{W} = [\mathbf{C}_{rr}]^{-1} \mathbf{C}_{rs} \stackrel{?}{=} \mathbf{W}_P$$

Typically, Population vector is not the optimal linear estimator.

Population Vector

- Population vector is optimal iff: The tuning curves are cosine, uniformly distributed and the noise follows a normal distribution with fixed variance
- In most cases, the population vector is biased and has a large variance
- The variance of the population vector estimate does not reflect Fisher information

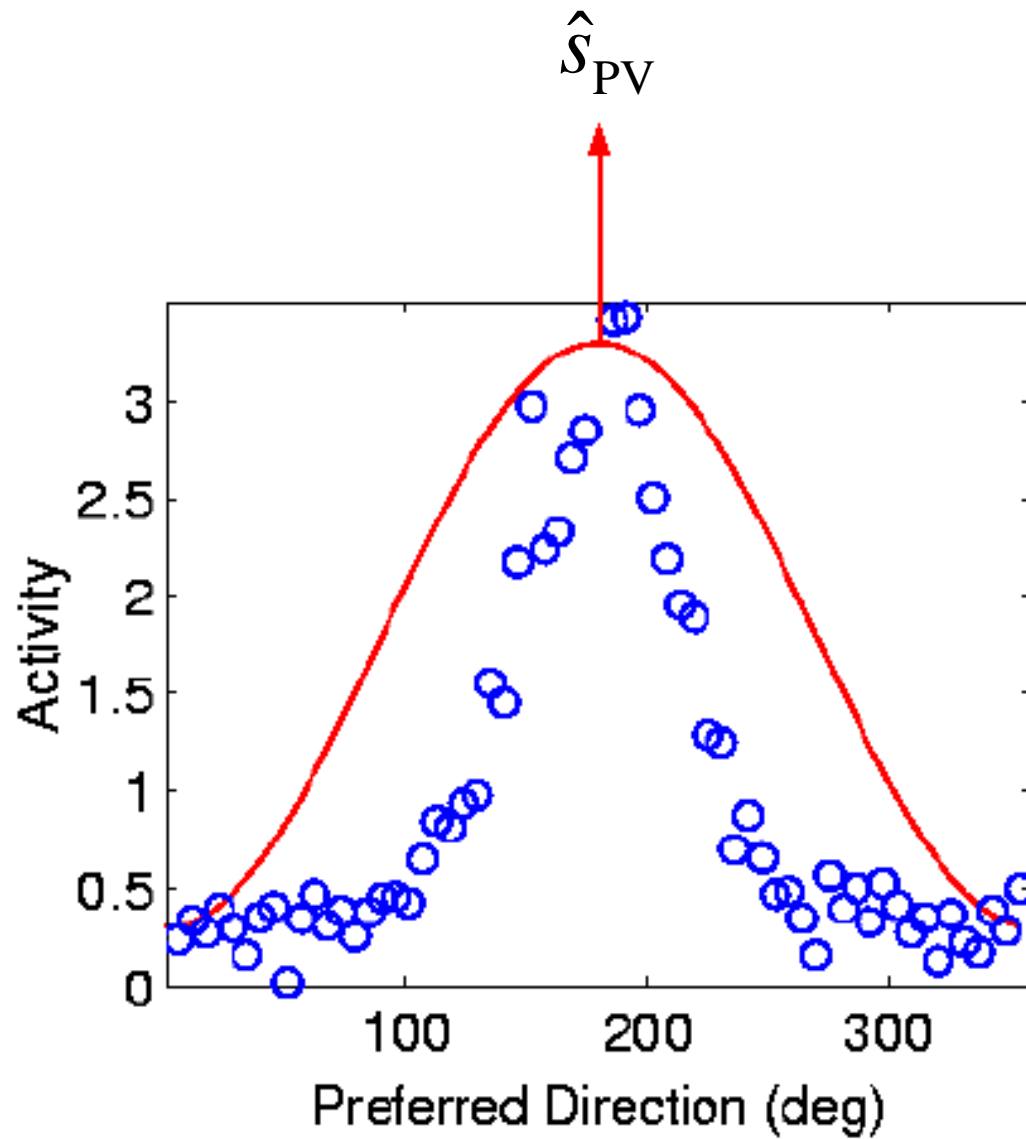
Population Vector



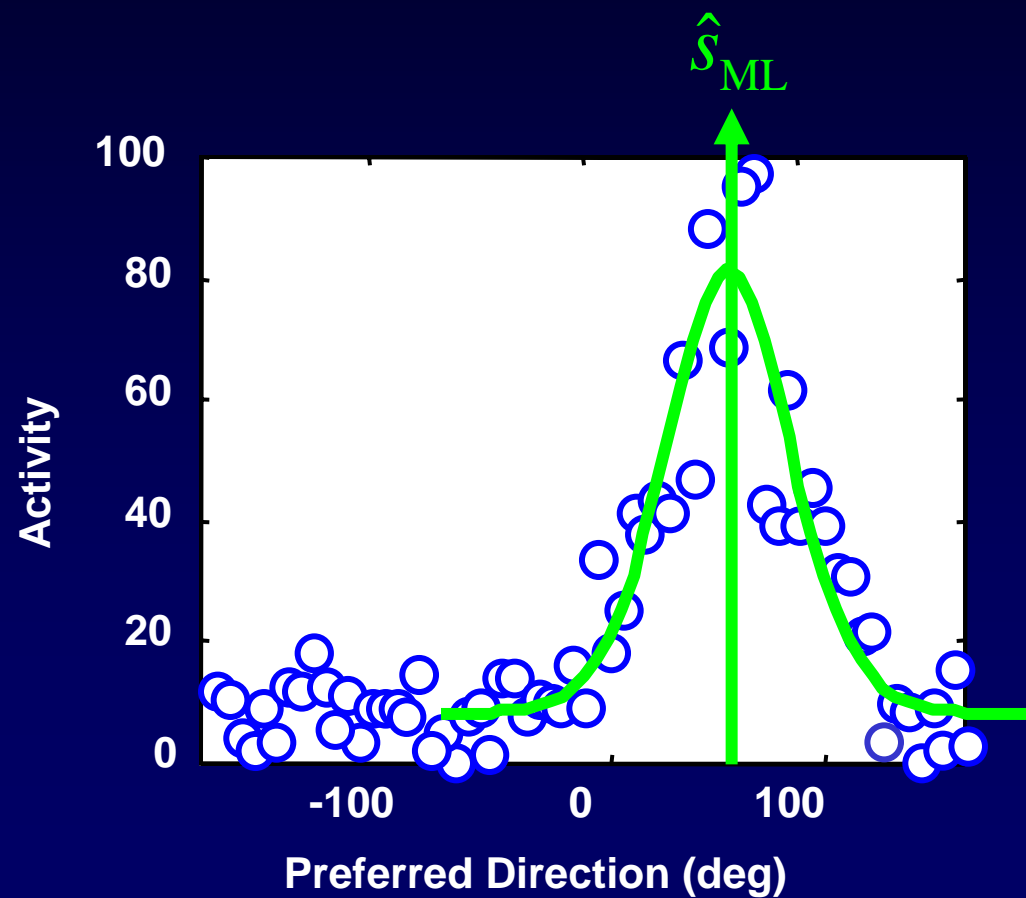
Population vector should NEVER be used to estimate information content!!!!

The indirect method is prone to severe problems...

Population Vector



Maximum Likelihood



Maximum Likelihood

If the noise is gaussian and independent

$$P(\mathbf{r} | s) = \prod_i \exp\left(-\frac{(r_i - f_i(s))^2}{2\sigma^2}\right)$$


Therefore

$$\log P(\mathbf{r} | s) = \sum_i -\frac{(r_i - f_i(s))^2}{2\sigma^2}$$

and the estimate is given by:

$$\hat{s} = \arg \min_s \sum_i \frac{(r_i - f_i(s))^2}{2\sigma^2}$$

Distance measure:
Template matching



Gradient descent for ML

- To minimize the likelihood function with respect to s , one can use a gradient descent technique in which s is updated according to:

$$s_{t+1} = s_t + \delta s_t$$

$$\delta s_t = -\frac{\partial L}{\partial s}$$

Gaussian noise with variance proportional to the mean

If the noise is gaussian with variance proportional to the mean, the distance being minimized changes to:

$$\hat{s} = \arg \min_s \sum_i \frac{(r_i - f_i(s))^2}{2f_i(s)}$$



Data point with small variance are weighted more heavily

Poisson noise

If the noise is Poisson then

$$p(r_i | s) = \frac{f_i(s)^{r_i} e^{-f_i(s)}}{r_i!}$$

And :

$$\begin{aligned} p(\mathbf{r} | s) &= \prod_i p(r_i | s) \\ &= \frac{e^{-\sum_i f_i(s)} \prod_i f_i(s)^{r_i}}{\prod_i r_i!} \end{aligned}$$

ML and template matching

Maximum likelihood is a template matching procedure BUT the metric used is not always the Euclidean distance, it depends on the noise distribution.

Bayesian approach

We want to recover $p(s/\mathbf{r})$. Using Bayes theorem, we have:

$$p(s | \mathbf{r}) = \frac{p(\mathbf{r} | s) p(s)}{p(\mathbf{r})}$$

likelihood of s → $p(\mathbf{r} | s)$

prior distribution over s → $p(s)$

posterior distribution over s → $p(s | \mathbf{r})$

prior distribution over \mathbf{r} → $p(\mathbf{r})$

Bayesian approach

What is the likelihood of s , $p(\mathbf{r} | s)$? It is the distribution of the noise... It is the same distribution we used for maximum likelihood.

Bayesian approach

- The prior $p(s)$ correspond to any knowledge we may have about s before we get to see any activity.
- Ex: prior for smooth and slow motions

Using the prior: Zhang et al

- For a time varying variable, one can use the distribution over the previous estimate as a prior for the next one.

$$P(s_{t+1} | \mathbf{r}_t, s_t) = \frac{P(\mathbf{r}_t | s_{t+1}, s_t) P(s_{t+1} | s_t)}{P(\mathbf{r}_t | s_t)}$$

$$= \frac{P(\mathbf{r}_t | s_{t+1}) P(s_{t+1} | s_t)}{P(\mathbf{r}_t | s_t)}$$

Nasty but independent
of s_{t+1}

Prior

Bayesian approach

Once we have $p(s|\mathbf{r})$, we can proceed in two different ways. We can keep this distribution for Bayesian inferences (as we would do in a Bayesian network) or we can make a decision about s . For instance, we can estimate s as being the value that maximizes $p(s|\mathbf{r})$. This is known as the maximum a posteriori estimate (MAP). For flat prior, ML and MAP are equivalent.

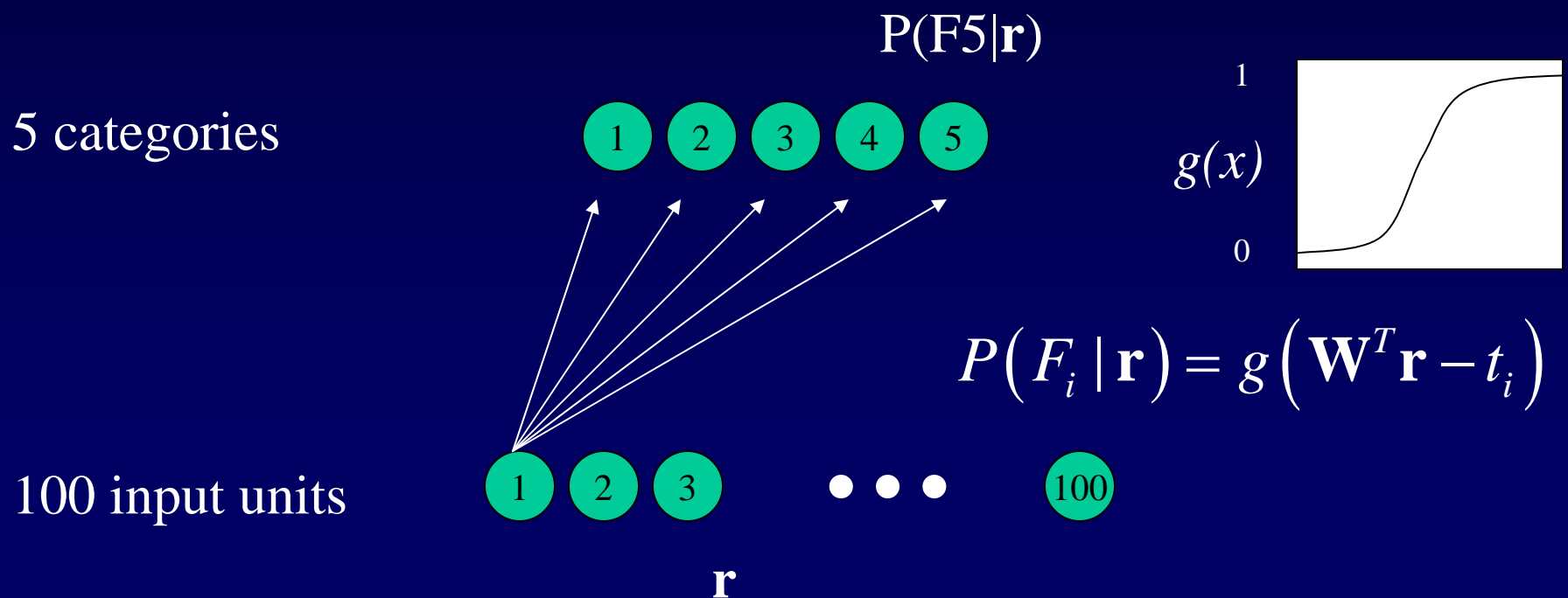
Bayesian approach

Limitations: the Bayesian approach and ML require a *lot* of data (estimating $p(\mathbf{r}|s)$ requires at least $n+(n-1)(n-1)/2$ parameters)...

Alternative: estimate $p(s|\mathbf{r})$ directly using a nonlinear estimate.

Bayesian approach: logistic regression

Example: Decoding finger movements in M1. On each trial, we observe 100 cells and we want to know which one of the 5 fingers is being moved.

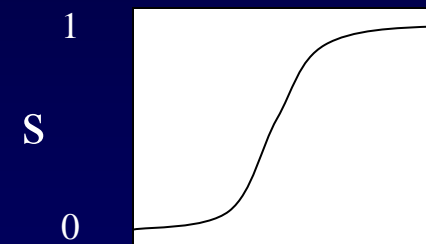
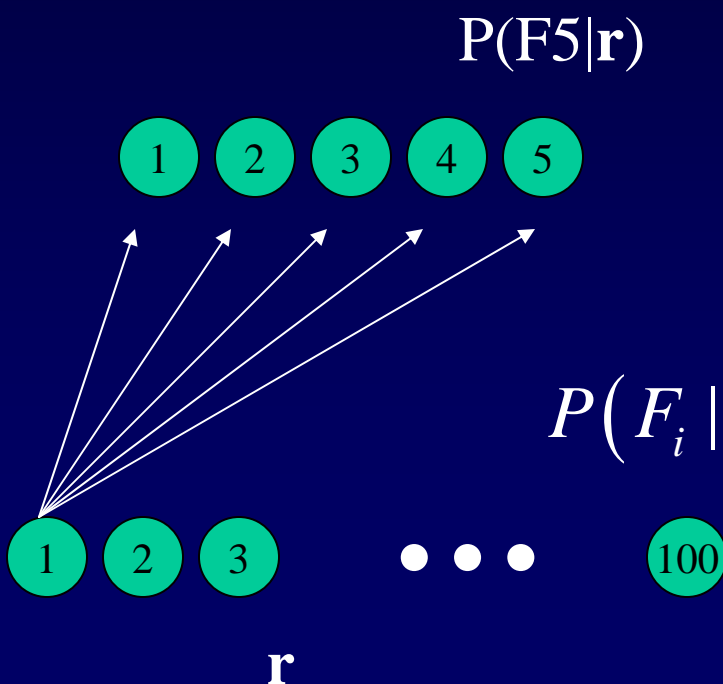


Bayesian approach: logistic regression

Example: $5N$ free parameters instead of $O(N^2)$

5 categories

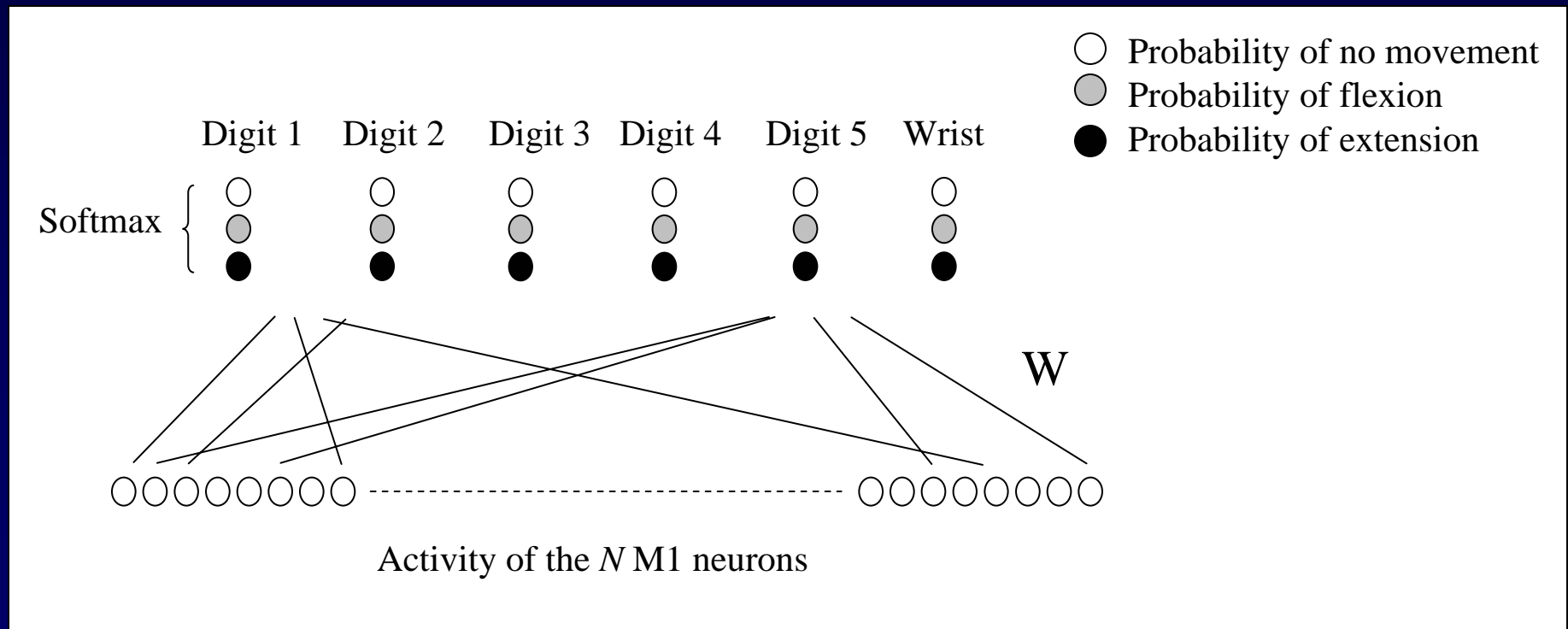
100 input units



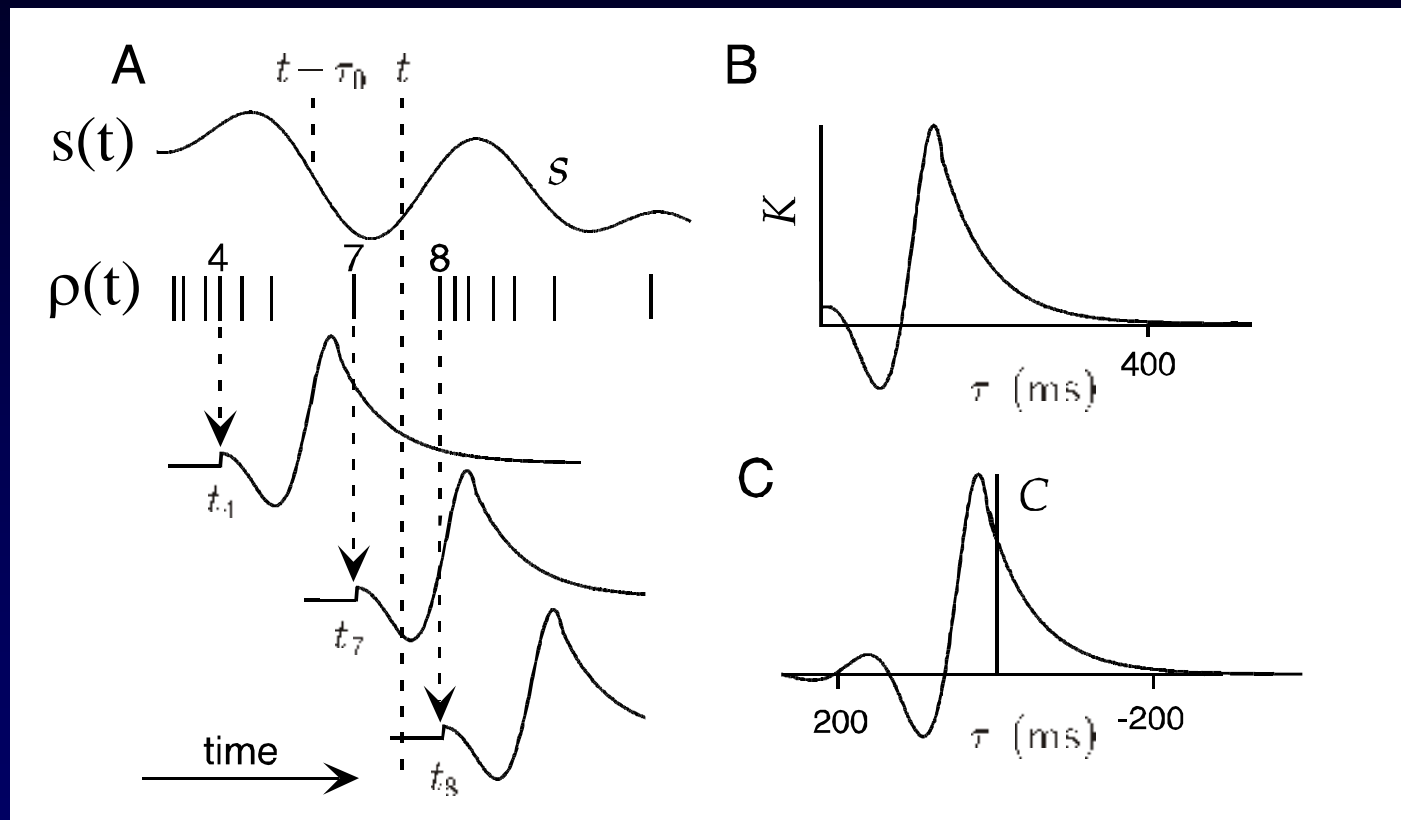
$$P(F_i | \mathbf{r}) = s(\mathbf{W}^T \mathbf{r} - t_i)$$

Bayesian approach: multinomial distributions

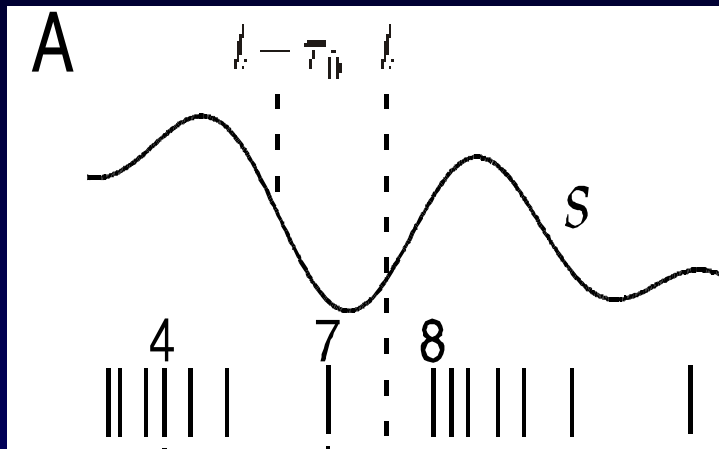
Example: Decoding finger movements in M1. Each finger can take 3 mutually exclusive states: no movement, flexion, extension.



Decoding time varying signals



Decoding time varying signals



$$\hat{s}(t - \tau_o) = k(t) * \rho(t) = \int_{-\infty}^t k(t - \tau) \rho(\tau) d\tau$$

Note the time shift...

Decoding time varying signals

$$\hat{s}(t - \tau_o) = \rho(t) * k(t)$$

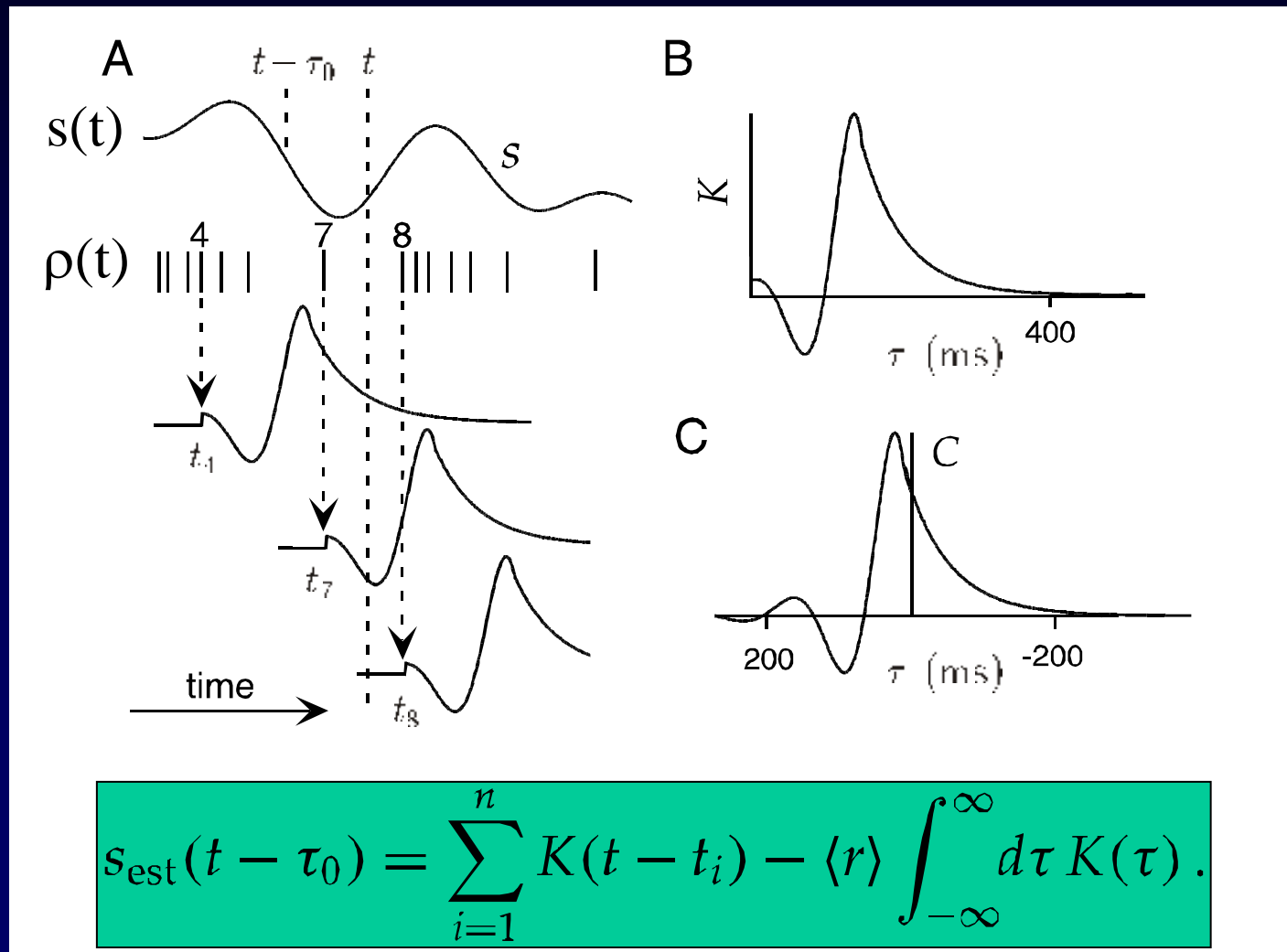
$$= \int_{-\infty}^t k(t - \tau) \rho(\tau) d\tau$$

$$= \int_{-\infty}^t k(t - \tau) \sum_{i=1}^n \delta(t_i) d\tau$$

$$= \sum_{i=1}^n k(t - t_i)$$

Discrete sum of templates
centered on spikes

Decoding time varying signals



Decoding time varying signals

- Finding the optimal kernel (similar to OLE)

$$\hat{s}(t) = \rho(t) * k(t)$$

$$\hat{\tilde{s}}(\omega) = \tilde{\rho}(\omega) \tilde{k}(\omega)$$

$$\tilde{k}(\omega) = \frac{\tilde{Q}_{\rho s}(-\omega)}{\tilde{Q}_{\rho\rho}(\omega)}$$

$$s_{\text{est}}(t - \tau_0) = \sum_{i=1}^n K(t - t_i) - \bar{r} \int d\tau K(\tau)$$

$$s_{\text{est}}(t - \tau_0) = \int d\tau (\rho(t - \tau) - \bar{r}) K(\tau)$$

$$E = \frac{1}{T} \int_0^T dt \left\langle (s_{\text{est}}(t - \tau_0) - s(t - \tau_0))^2 \right\rangle$$

Autocorrelation function of the spike train

$$E = \frac{1}{T} \int_0^T dt \left\langle \left(\int d\tau (\rho(t - \tau) - \bar{r}) K(\tau) - s(t - \tau_0) \right)^2 \right\rangle$$

$$\int_{-\infty}^{\infty} d\tau' Q_{\rho\rho}(\tau - \tau') K(\tau') = Q_{\rho s}(\tau - \tau_0)$$

Appendix A chapter 2

$$Q_{\rho\rho}(\tau - \tau') = \frac{1}{T} \int_0^T dt (\rho(t - \tau) - \bar{r})(\rho(t - \tau') - \bar{r})$$

Correlation of the firing rate and stimulus

$$\text{if } Q_{\rho\rho}(\tau) = \bar{r} \delta(\tau)$$

$$\text{then } K(\tau) = \frac{1}{\bar{r}} Q_{\rho s}(\tau - \tau_0) = C(\tau - \tau_0) = \frac{1}{\langle n \rangle} \left\langle \sum_{i=1}^n s(t_i + \tau - \tau_0) \right\rangle$$

otherwise

$$K(\tau) = \frac{1}{2\pi} \int d\omega \tilde{K}(\omega) \exp(-i\omega\tau)$$

$$\tilde{K}(\omega) = \frac{\tilde{Q}_{\rho s}(\omega) \exp(-i\omega\tau_0)}{\tilde{Q}_{\rho\rho}(\omega)}$$

If the spike train is uncorrelated, the optimal kernel is the spike triggered average of the stimulus

$$Q_{\rho s}(\tau - \tau') = \frac{1}{T} \int_{-\infty}^T dt (\rho(t - \tau) - \bar{r}) s(t - \tau')$$

$$\begin{aligned}
Q_{\rho s}(\tau - \tau') &= \frac{1}{T} \int_0^T dt (\rho(t - \tau') - \bar{r}) s(t - \tau) \\
&= \frac{1}{T} \int_0^T dt \left(\sum_{i=1}^N \delta(t_i - \tau') - \bar{r} \right) s(t - \tau) \\
&= \frac{1}{T} \sum_{i=1}^N \int_0^T dt \delta(t_i - \tau') s(t - \tau) - \frac{1}{T} \int_0^T dt \bar{r} s(t - \tau) \\
&= \frac{1}{T} \sum_{i=1}^N s(t_i + \tau - \tau')
\end{aligned}$$

