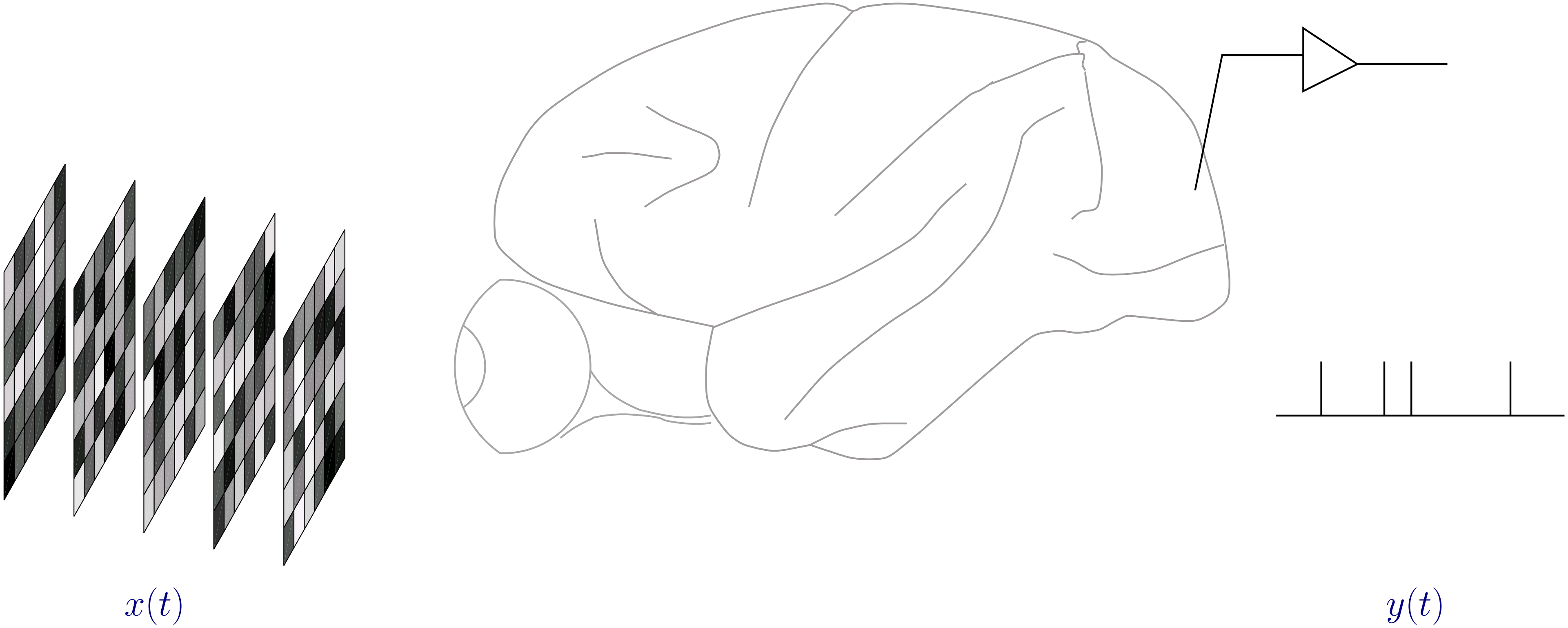
Robust regression and non-linear kernel methods for characterization of neuronal response functions from limited data

Maneesh Sahani
Gatsby Computational Neuroscience Unit
University College, London

Jennifer Linden
Keck Center for Integrative Neurosciences
University of California at San Francsico

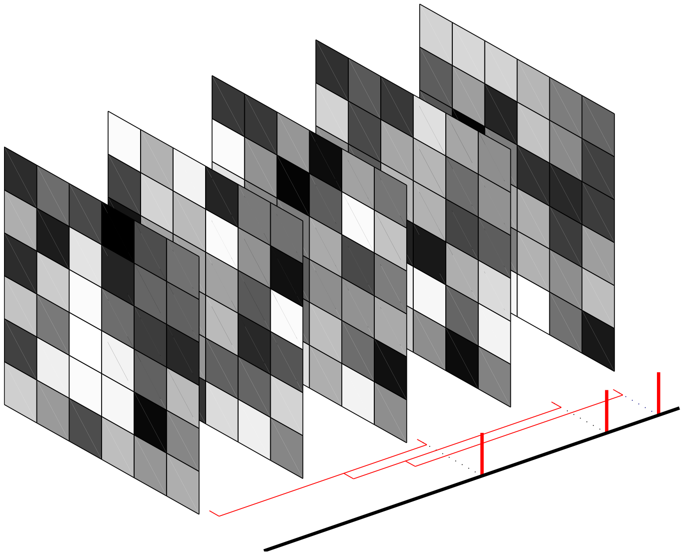
Studying perceptual systems

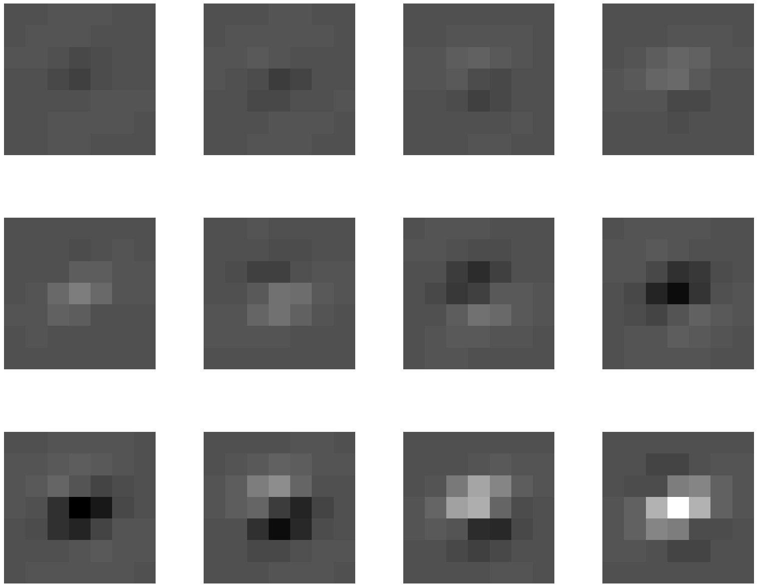


Decoding: $\hat{x}(t) = G[y(t)]$ (reconstruction)

Encoding: $\hat{y}(t) = F[x(t)]$ (systems identification)

Spike-triggered average

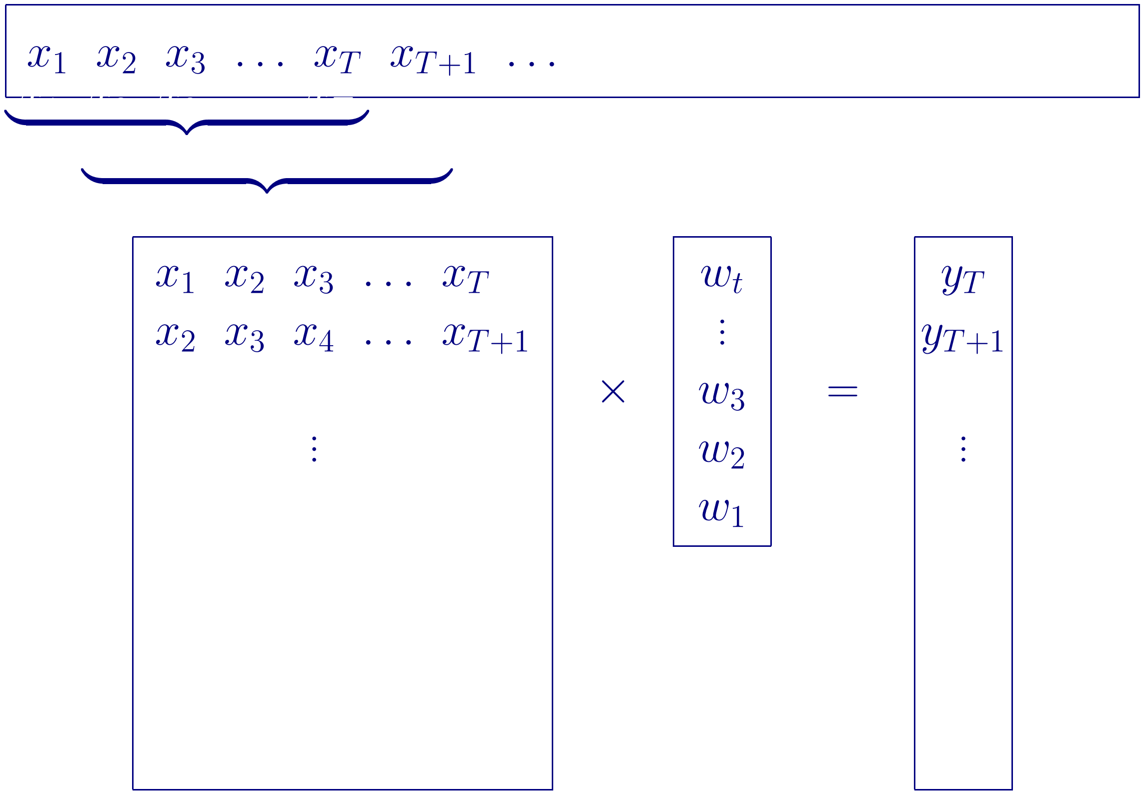




Decoding:	mean of $\mathsf{P}\,(x \mid y = 1)$
Encoding:	predictive filter

Wiener Filtering is Linear regression

$$y(t) = \int_0^T x(t-\tau)w(\tau)d\tau$$



$$\boxed{x_1 \;\; x_2 \;\; x_3 \;\; \dots \;\; x_T \;\; x_{T+1} \;\; \dots}$$

$$\begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_T \\ x_2 & x_3 & x_4 & \dots & x_{T+1} \\ & & \vdots & & \end{bmatrix} \times \begin{bmatrix} w_t \\ \vdots \\ w_3 \\ w_2 \\ w_1 \end{bmatrix} = \begin{bmatrix} y_T \\ y_{T+1} \\ \vdots \end{bmatrix}$$

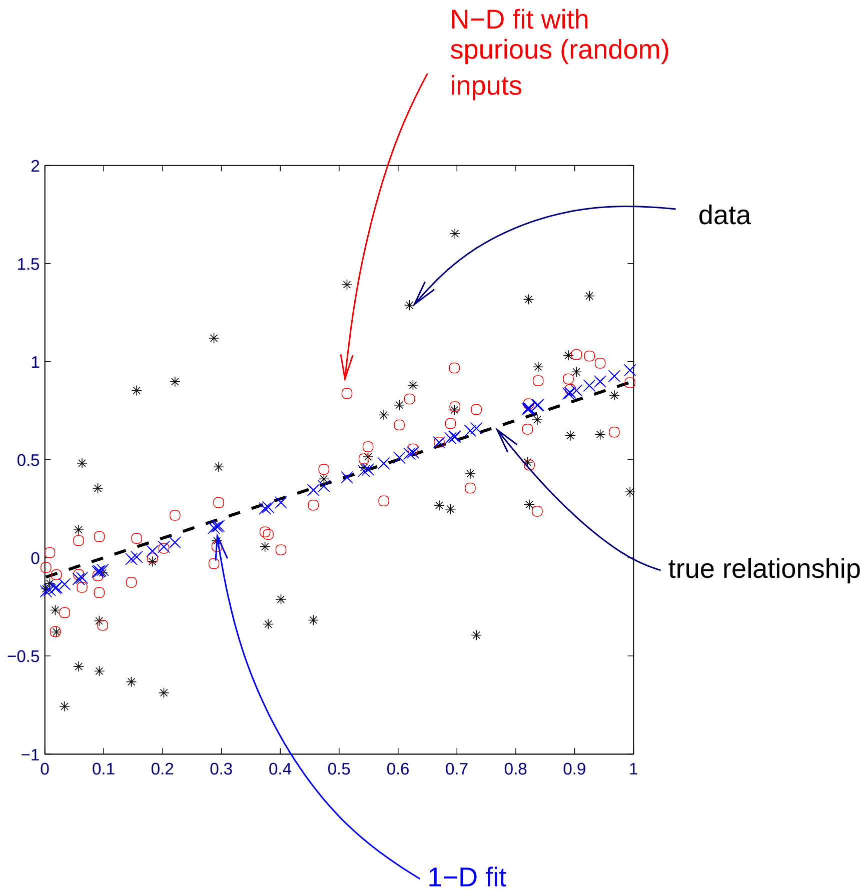
$$XW = Y$$

$$W(\omega) = \frac{X(\omega)^* Y(\omega)}{|X(\omega)|^2}$$

$$W = \underbrace{(X'X)}_{\Sigma_{SS}}{}^{-1} \underbrace{(X'Y)}_{\text{STA}}$$

Overfitting

- Maximum-likelihood estimates often overfit to noise in the training data.
- Overfitting is a fundamental problem in data modelling. It can never be completely overcome. Even the correct model, with the correct priors will overfit.
- One common signature of overfitting in (quasi-)linear models is the appearance of large weights of opposite signs: the difference has been tuned by likelihood optimization to model the training data.
- Such solutions are also often unstable: change the data a little and the weights fluctuate alarmingly.
- Such overfitting is often combatted by penalizing large weights: called "weight decay" in the neural-network literature, "ridge regression" in the statistics literature, and equivalent to a prior distribution centered on zero weight in the Bayesian literature.



ARD

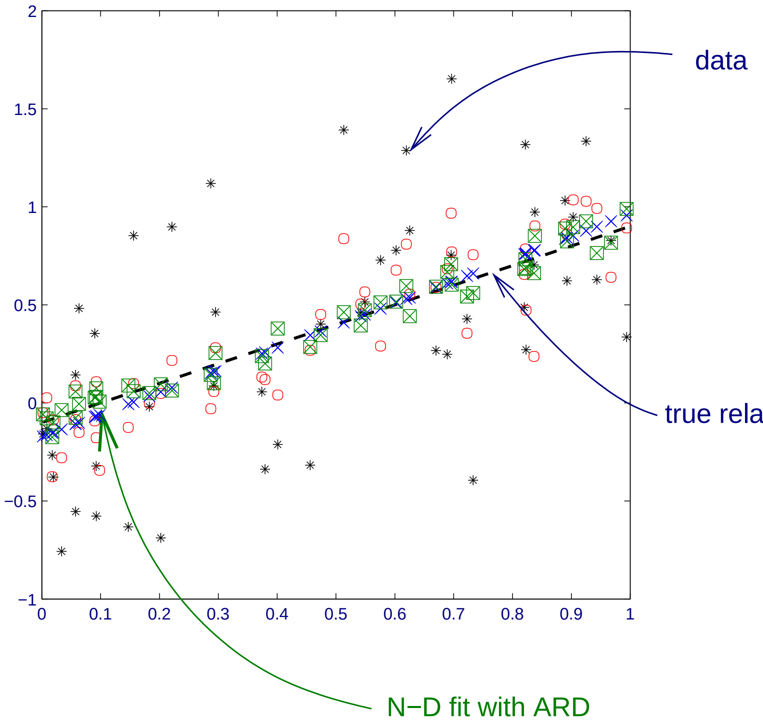
Maximum-likelihood regression can be improved by a Bayesian analysis called Automatic Relevance Determination (ARD) due to MacKay and Neal.

Assume

$$w_i \sim \mathcal{N}(0, 1/\alpha_i) \qquad \text{independently}$$

and define

$$L_2(\{\alpha_i\}) = \int \cdots \int dw_1 \ldots dw_D \underbrace{\mathsf{P}\left(Y \mid X; W\right)}_{L(W)} \mathsf{P}\left(W \mid \{\alpha_i\}\right)$$



data

true rela

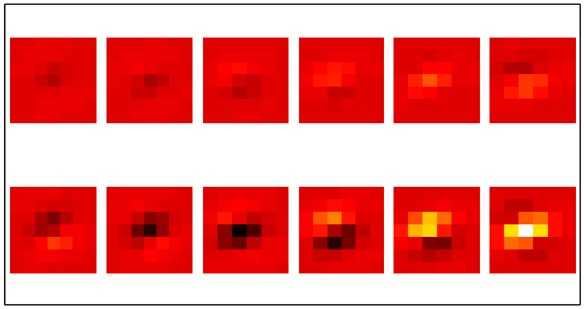
N−D fit with ARD

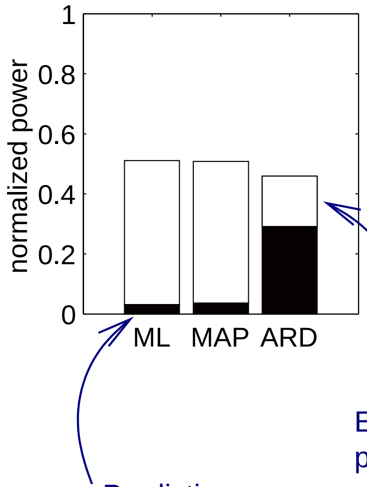
Optimize L_2 with respect to $\{\alpha_i\}$ (ML-2)

$$\begin{array}{llll} \alpha_i \to \infty & \Rightarrow & w_i = 0 & \textit{irrelevant} \\ \alpha_i \text{ finite} & \Rightarrow & w_i = \operatorname{argmax} \mathsf{P}\left(w_i \mid X, Y, \alpha_i\right) & \textit{relevant} \end{array}$$

Simulated simple cell

actual filter

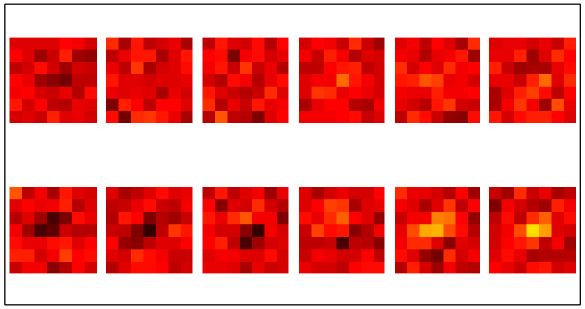




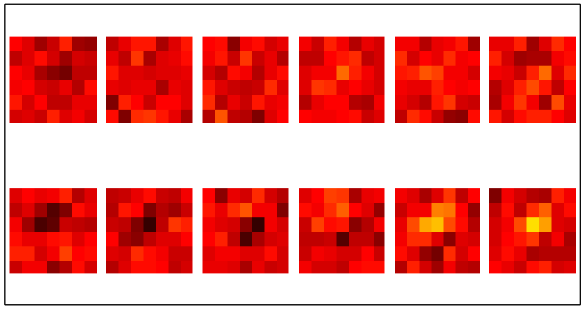
Predictive power

Excess predicted power

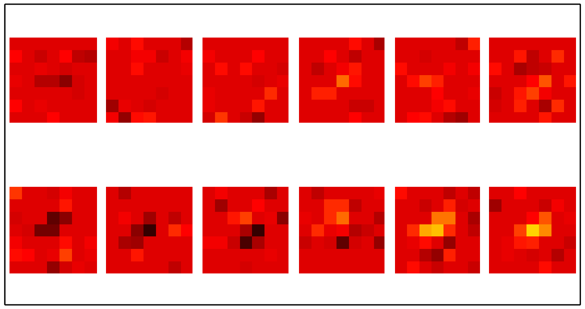
maximum likelihood (ML)



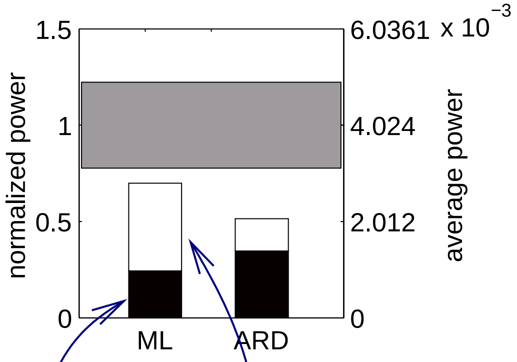
ridge regression (MAP)



automatic relevance detection (ARD)

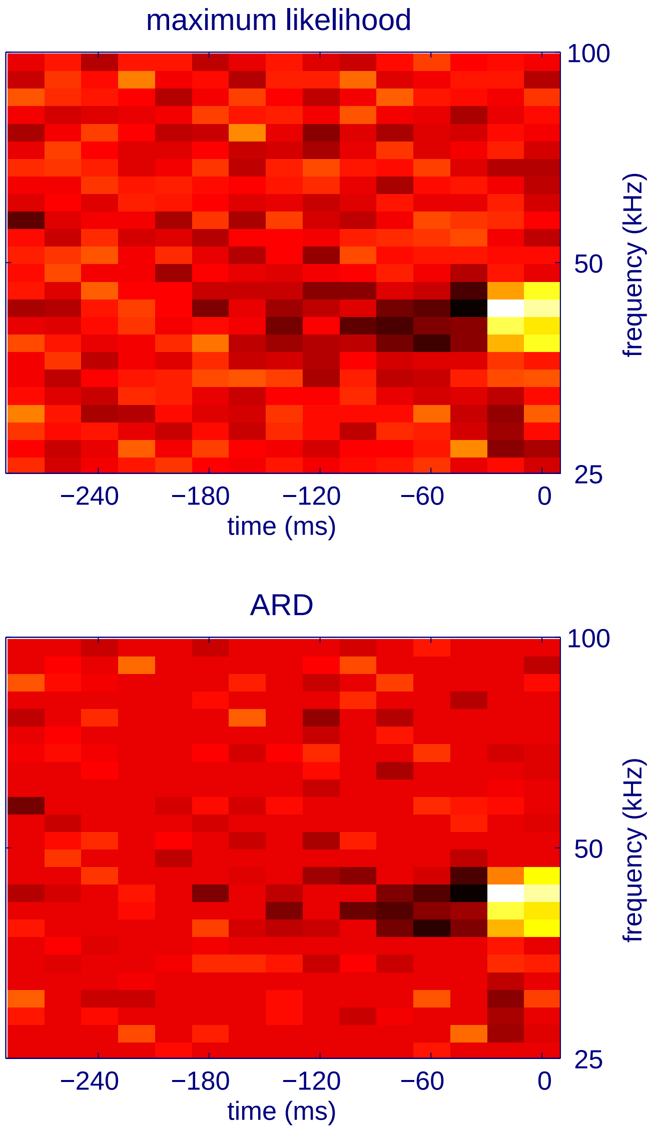


Rat auditory cortex cell



Predictive power

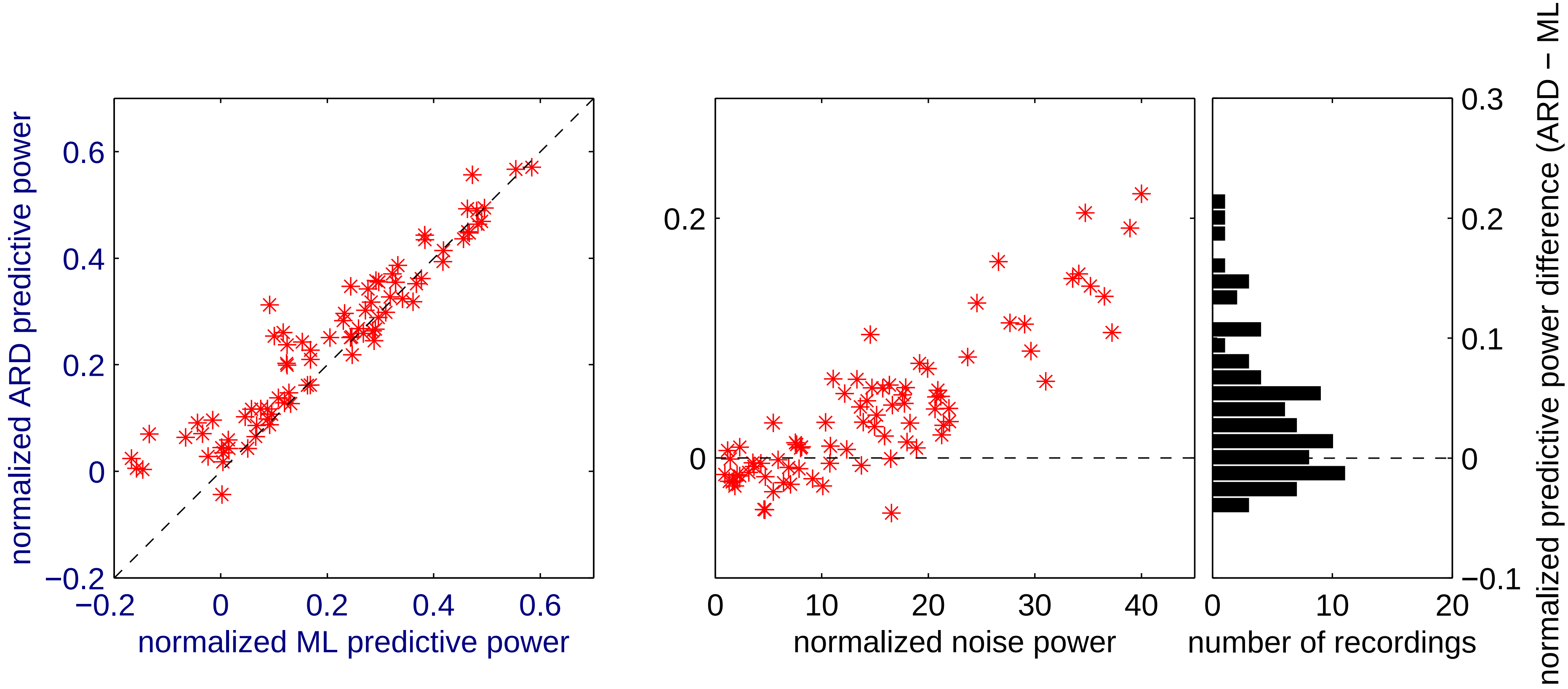
Excess predicted power



R2001061802G/20010911/pen12loc2poisshical008a

Population predictive performance

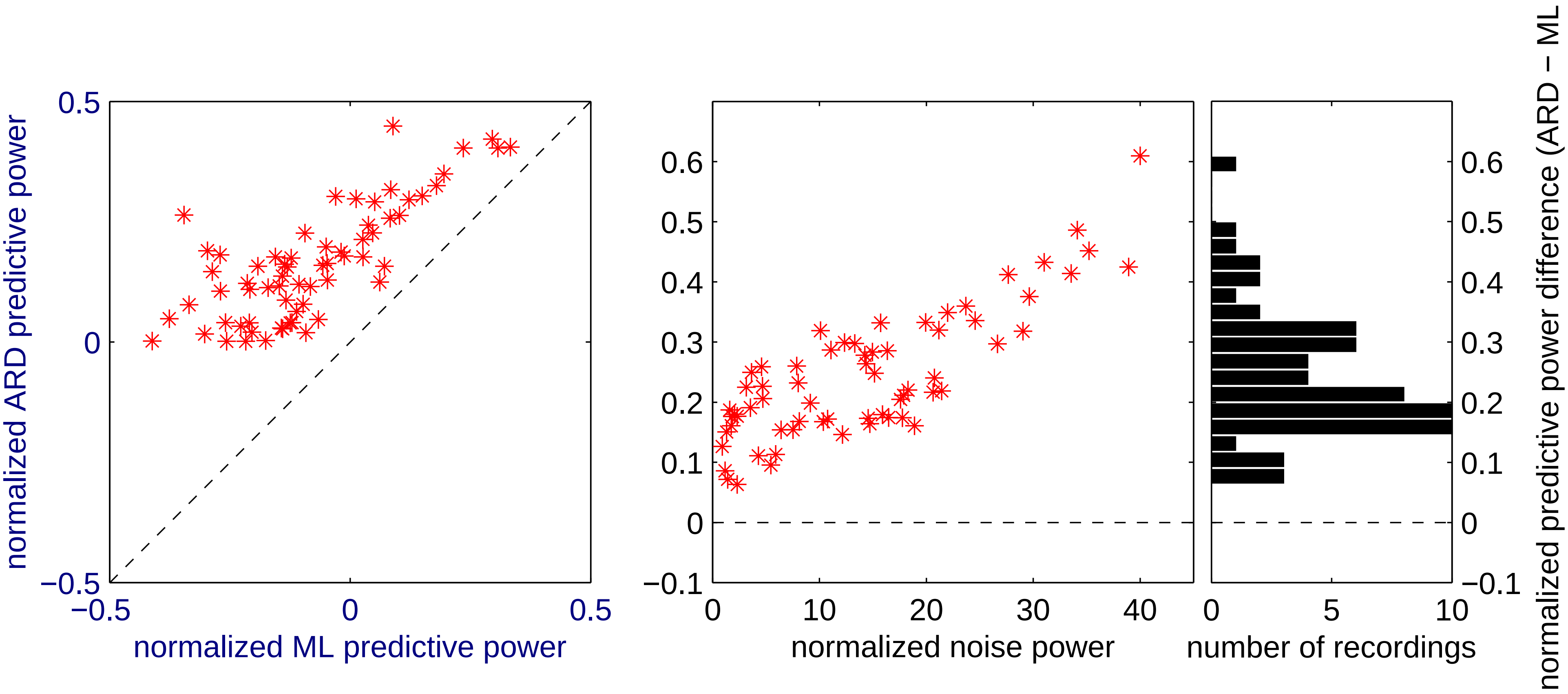
Tested on new responses to the same stimuli.



Parameters can still overfit to the particular stimulus used.

Population predictive performance

Tested on responses to new stimuli.



Evaluating constrained linear models by prediction

Separating signal power from noise

- The maximum possible reduction in variance is given by the power of the stimulus-locked signal.
- Repeated trials that use the same stimulus allow us to separate signal from noise.
- The estimator is unbiased, and has a computable variance, provided the noise distribution is not pathalogical.
- This is an estimated bound, but will not be practicably achievable because any model (even the right one!) will overfit given limited data.

$$y(t) = \underbrace{\mu(t)}_{\text{signal}} + \underbrace{\eta(t)}_{\text{noise}}$$

Taking powers: $\left(P(y(t)) = \left\langle (y(t) - \langle y \rangle)^2 \right\rangle\right)$

$$\underbrace{P(y)}_{\text{observed power}} \quad = \quad \underbrace{P(\mu)}_{\text{signal power}} \quad + \quad \underbrace{\langle \sigma^2 \rangle}_{\text{average noise varia}}$$

Averaging N trials

$$P(\overline{y}) = P(\mu) + \frac{\langle \sigma^2 \rangle}{N}$$

Thus

$$\widehat{P}(\mu) = \frac{N P(\overline{y}) - \overline{P(y)}}{N - 1}$$

Which is an unbiased estimator.

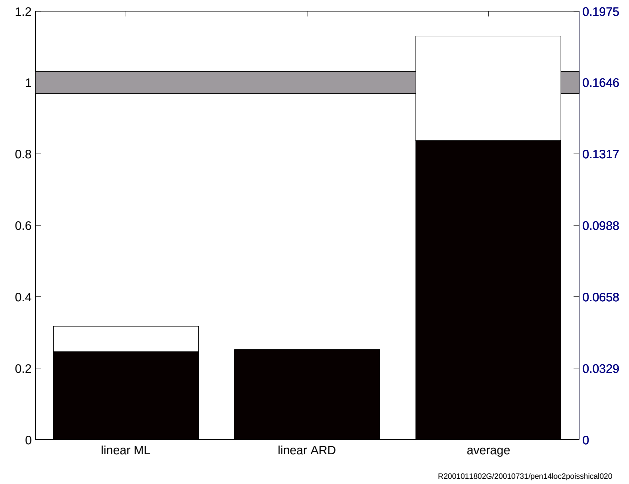
Its variance is given by

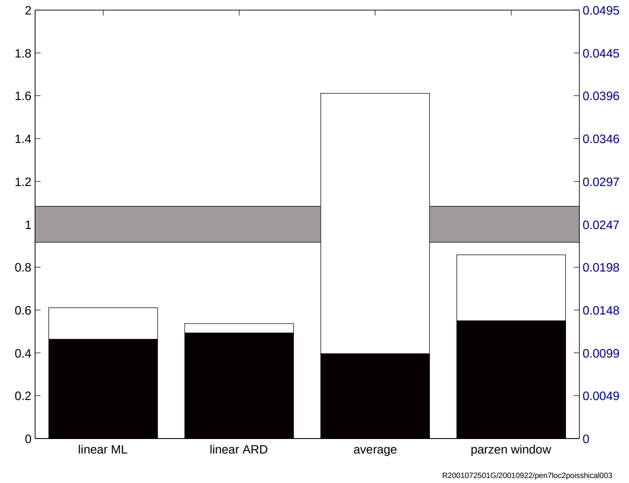
$$\mathcal{V}ar\left[\widehat{P}\right] = \frac{4}{N}\left(\frac{1}{T^2}\boldsymbol{\mu}'\Sigma\boldsymbol{\mu} - \frac{2}{T}\mu\boldsymbol{\sigma}'\boldsymbol{\mu} + \mu\sigma\mu\right) + \\ \frac{2}{N(N-1)}\left(\frac{1}{T^2}\mathrm{Tr}\left[\Sigma\Sigma\right] - \frac{2}{T}\boldsymbol{\sigma}'\boldsymbol{\sigma} + \sigma^2\right)$$

where Σ, $\boldsymbol{\sigma}$ and σ are statistics describing the (co)variance of the noise and must themselves be estimated from data.

Non-parametric predictability estimates

- Non-parametric prediction can provide an estimated lower bound on predictable power.
- The cell response function is not modelled explicitly. Instead, responses are predicted based on observations made on similar stimuli.
- The simplest approach is to average the training data over trials. However, this will overfit to the noise severely. Also, it does not allow prediction on new stimuli.
- An alternative is to smooth the average with reference to the stimulus. This is sometimes known as locally-weighted regression or Parzen-window regression.

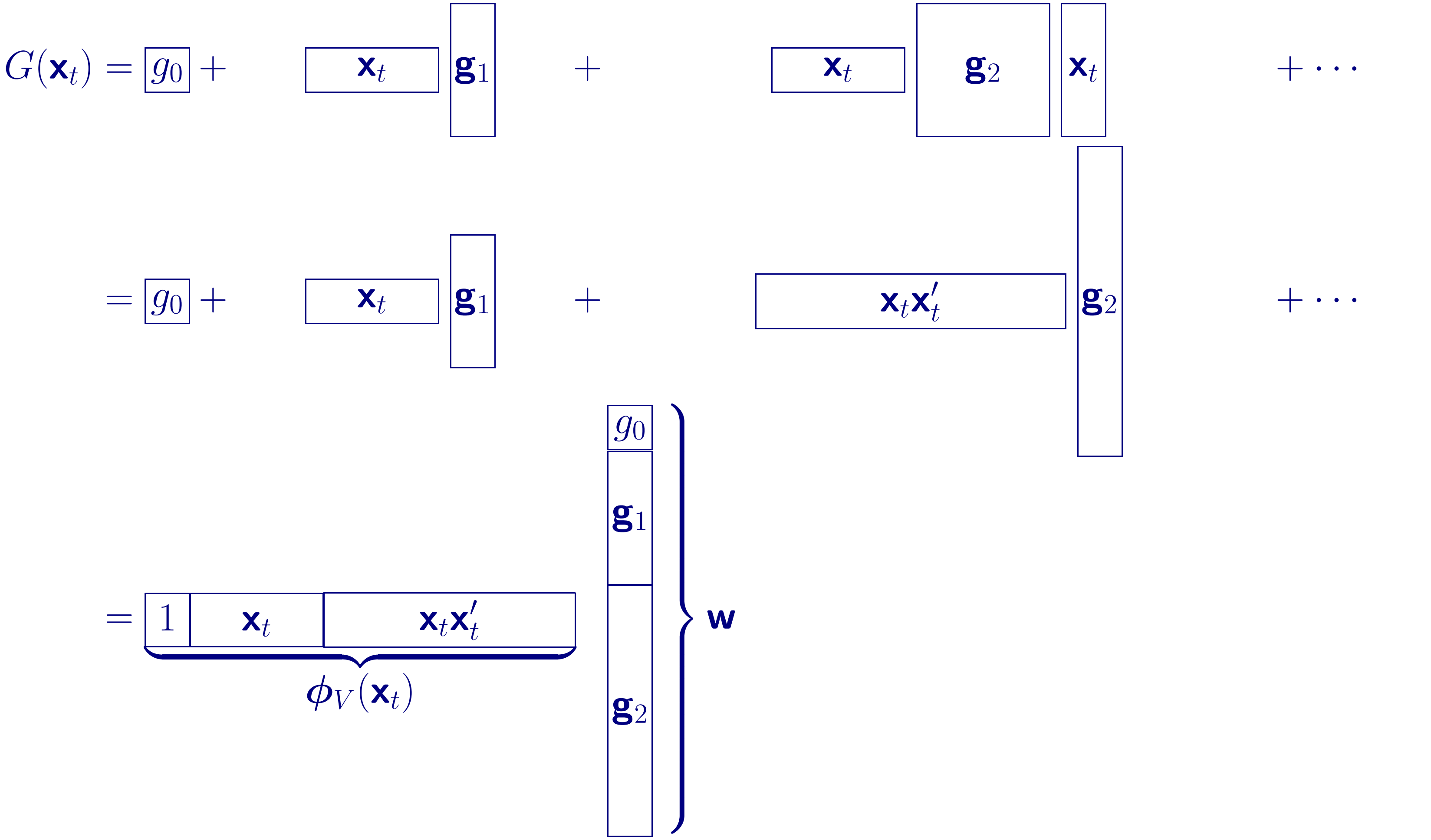




Nonlinearities – the Volterra Series

The usual extension to non-linear functionals involves a power series in $x(t)$.

$$\hat{y}(t) = G[x] = g_0 + \int d\tau_1\ g_1(\tau_1)x(t-\tau_1) + \iint d\tau_1 d\tau_2\ g_2(\tau_1,\tau_2)x(t-\tau_1)x(t-\tau_2) + \iiint \ldots$$



$$G(\mathbf{x}_t) = \boxed{g_0} + \boxed{\mathbf{x}_t}\,\boxed{\mathbf{g}_1} + \boxed{\mathbf{x}_t}\,\boxed{\mathbf{g}_2}\,\boxed{\mathbf{x}_t} + \cdots$$

$$= \boxed{g_0} + \boxed{\mathbf{x}_t}\,\boxed{\mathbf{g}_1} + \boxed{\mathbf{x}_t\mathbf{x}_t'}\,\boxed{\mathbf{g}_2} + \cdots$$

$$= \underbrace{\boxed{1}\boxed{\mathbf{x}_t}\boxed{\mathbf{x}_t\mathbf{x}_t'}}_{\phi_V(\mathbf{x}_t)}\ \left.\begin{array}{|c|}\hline g_0 \\ \hline \mathbf{g}_1 \\ \hline \mathbf{g}_2 \\ \hline \end{array}\right\}\mathbf{w}$$

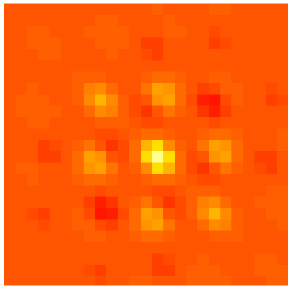
$$G(\mathbf{x}_t) = \phi_V(\mathbf{x}_t)\cdot\mathbf{w} = y_t$$

$$\Phi_V(X)W = Y$$

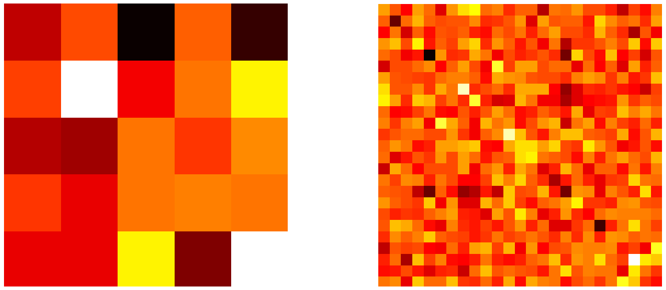
Thus, higher order terms can be estimated by linear regression in an augmented space.

Simulated complex cell

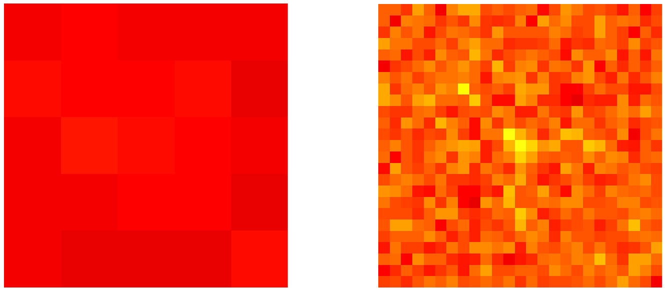
actual filter

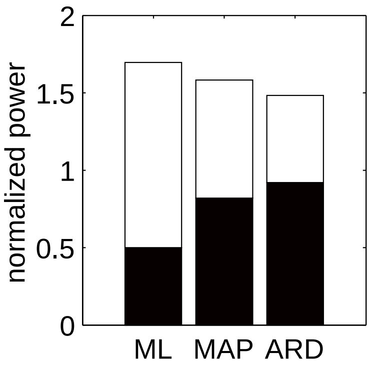


maximum likelihood (ML)

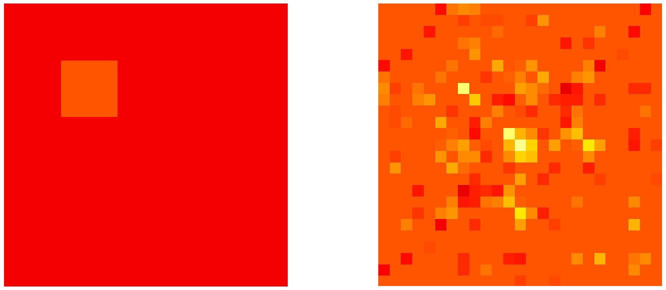


ridge regression (MAP)

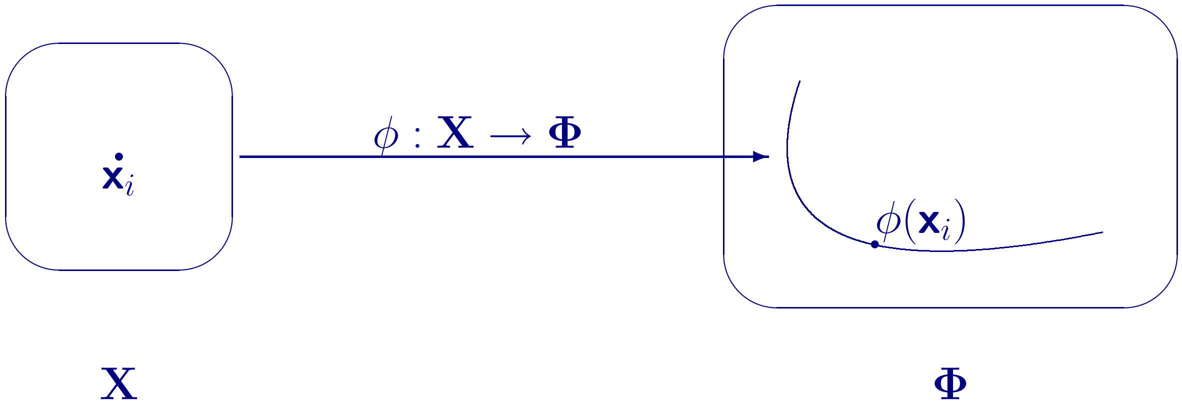




automatic relevance determination (ARD)



Kernel methods



For certain embeddings ϕ we can find a kernel function $K : \mathbf{X} \times \mathbf{X} \rightarrow \mathbb{R}$ such that

$$K(\mathbf{x}_1, \mathbf{x}_2) = \phi(\mathbf{x}_1) \cdot \phi(\mathbf{x}_2)$$

Thus linear operations in $\mathbf{\Phi}$ can be performed without explicit embedding. $\mathbf{\Phi}$ can be very high-dimensional (even ∞) without intractibility.

Only certain K have an associated ϕ and *vice versa* (Mercer).

Kernel regression

Linear regression in the augmented space requires finding a weight vector $\boldsymbol{\gamma}$ to solve

$$\Phi(X)'\boldsymbol{\gamma} = Y$$

If $\boldsymbol{\gamma} = \phi(\mathbf{x}_\gamma)$ we could write

$$K(\mathbf{x}_t, \mathbf{x}_\gamma) = y_t$$

but this is nonlinear in $\mathbf{x}_\gamma$ and the restriction $\boldsymbol{\gamma} \in \phi(\mathbf{X}) \subset \boldsymbol{\Phi}$ is too stringent.
Instead take

$$\boldsymbol{\gamma} = \sum_i w_i \phi(\mathbf{x}_i)$$

giving

$$\sum_i w_i K(\mathbf{x}_t, \mathbf{x}_i) = y_t \qquad \text{or} \qquad KW = Y$$

If $\{\mathbf{x}_i\} = \{\mathbf{x}_t\}$ this representation is complete because any additional part of $\boldsymbol{\gamma}$ would lie in the null space of Φ. In practice, we may choose $\{\mathbf{x}_i\} \subset \{\mathbf{x}_t\}$ randomly, or by a greedy process.

Volterra space and the polynomial kernels

The Volterra augmented space has a corresponding kernel function.

$$\text{Define}\qquad \begin{aligned} K_{p(1)}(\mathbf{x}_1,\mathbf{x}_2) &= \mathbf{x}_1\cdot\mathbf{x}_2 \\ K_{p(2)}(\mathbf{x}_1,\mathbf{x}_2) &= (\mathbf{x}_1\cdot\mathbf{x}_2)^2 \\ &= \left(\sum_i x_1^i x_2^i\right)^2 = \sum_{ij} x_1^i x_2^i x_1^j x_2^j = \sum_{ij}(x_1^i x_1^j)(x_2^i x_2^j) \\ &= (\mathbf{x}_1\mathbf{x}_1')(:)\cdot(\mathbf{x}_2\mathbf{x}_2')(:) \\ &\vdots \end{aligned}$$

then the kernel

$$K_V(\mathbf{x}_1,\mathbf{x}_2) = K_{p(1)}(\mathbf{x}_1,\mathbf{x}_2) + K_{p(2)}(\mathbf{x}_1,\mathbf{x}_2) + \ldots$$

corresponds to the Volterra augmentation

$$\phi_V(\mathbf{x}) = \begin{bmatrix} \mathbf{x}(:) \\ \mathbf{x}\mathbf{x}'(:) \\ \vdots \end{bmatrix}$$

(Also useful is the kernel $K_{\overline{p}(n)}(\mathbf{x}_1,\mathbf{x}_2) = (1+\mathbf{x}_1,\mathbf{x}_2)^n$)

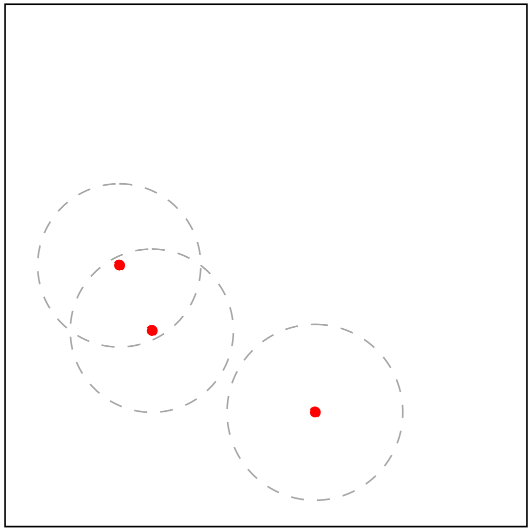
Other kernels

Volterra expansion ⇔ polynomial kernels

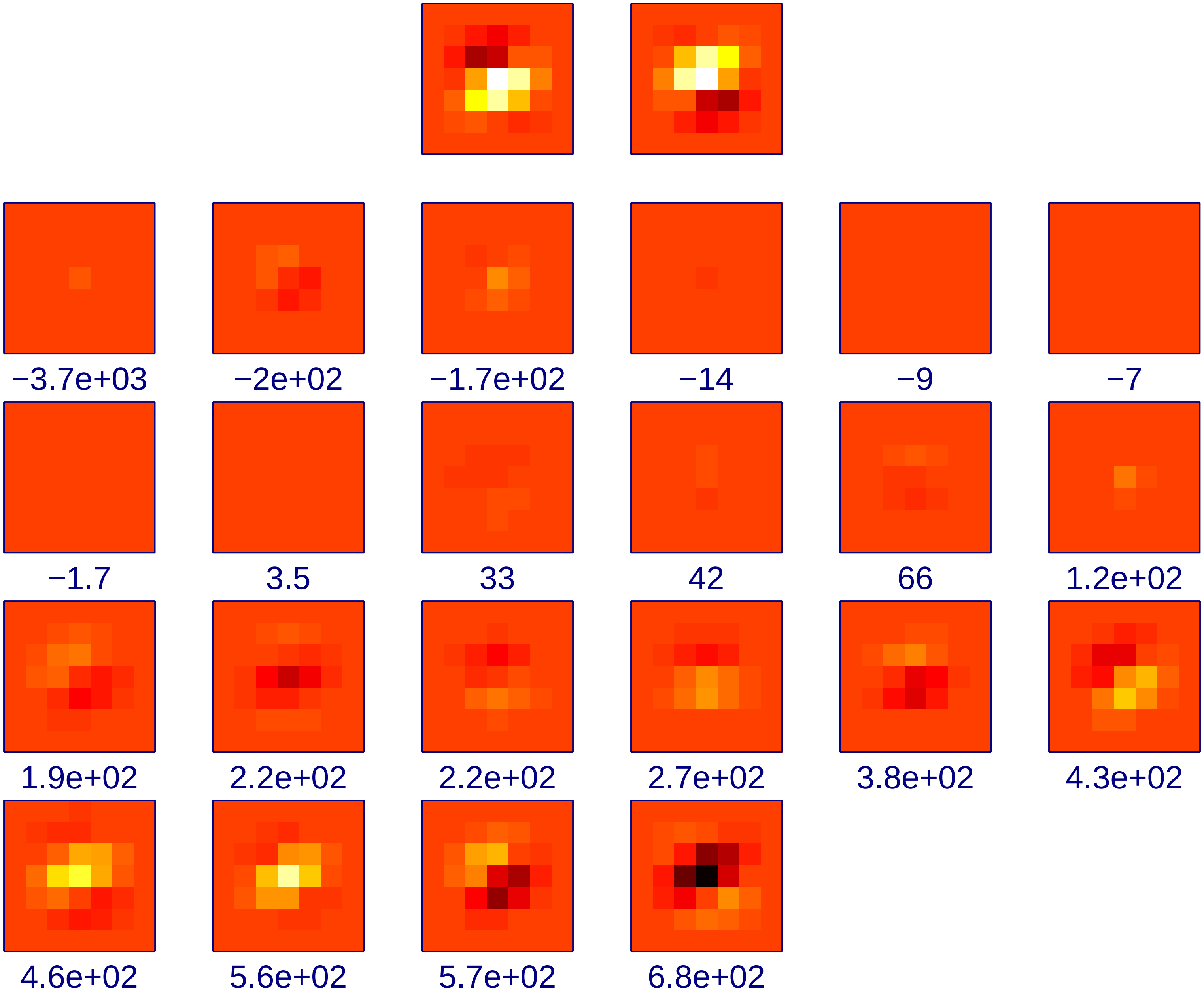
Other kernels can provide other non-linear expansions of the transform; some with useful intuitive interpretations. For example,

$$K_\sigma(\mathbf{x}_1, \mathbf{x}_2) = e^{-\|\mathbf{x}_1 - \mathbf{x}_2\|^2/2\sigma^2} \qquad \Rightarrow \qquad y_t = \sum_i w_i e^{-\|\mathbf{x}_t - \mathbf{x}_i\|^2/2\sigma^2}$$

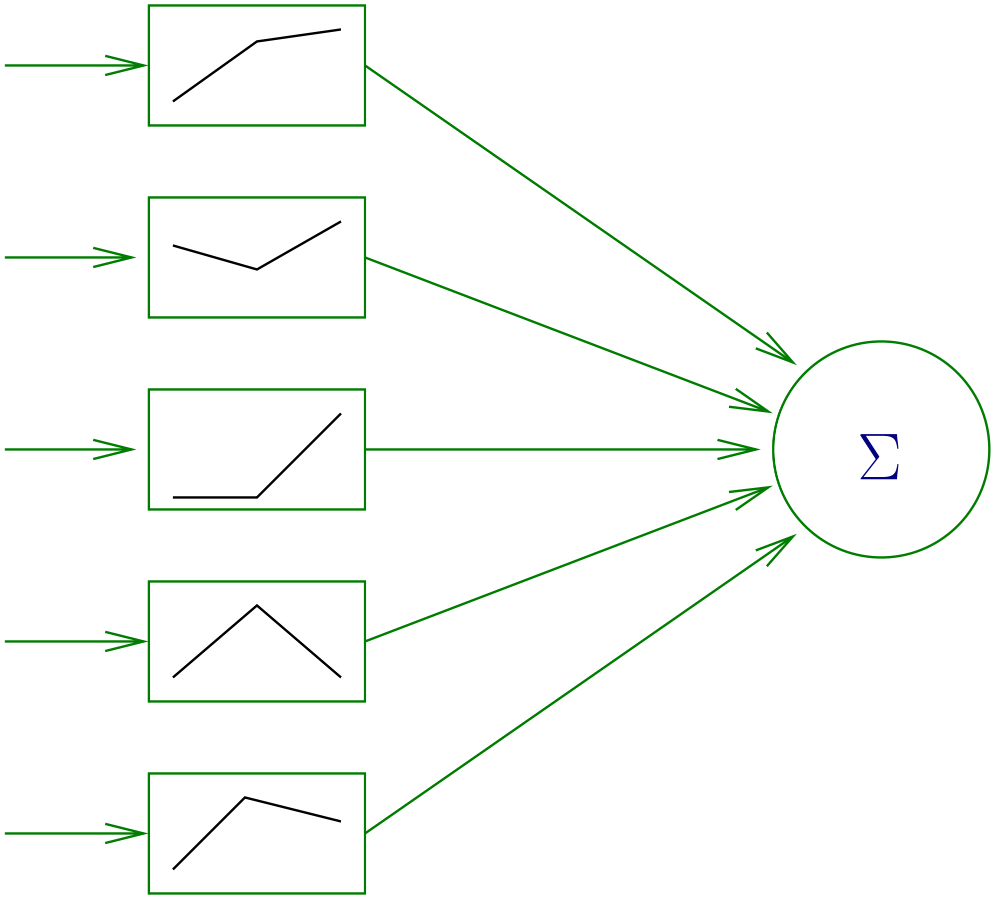
ARD will identify a small number of "characteristic" vectors $\mathbf{x}_i$, providing a "tiled" approximation to the nonlinearity.



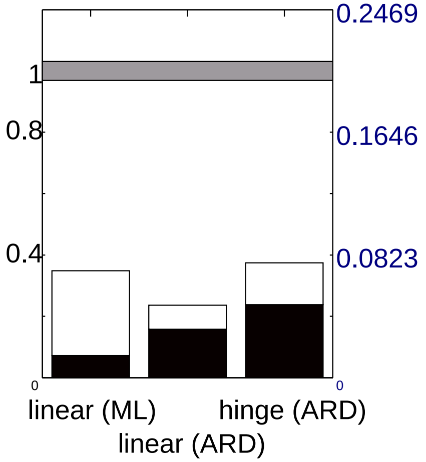
Simulated complex cell

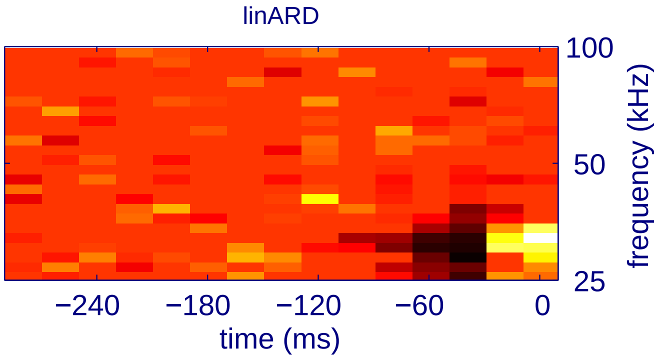


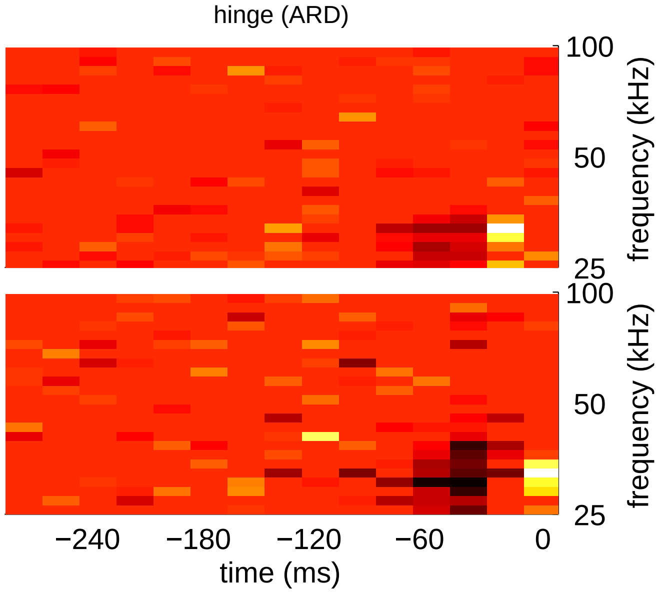
Input non-linearities



Hinge weights example 1

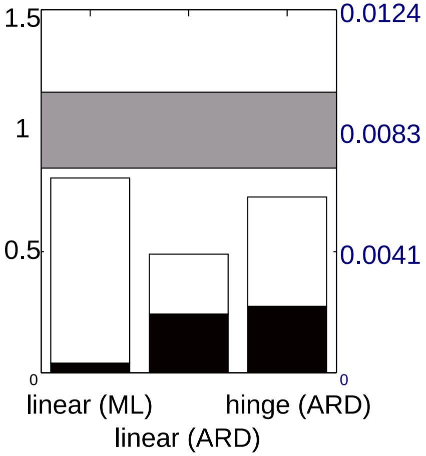


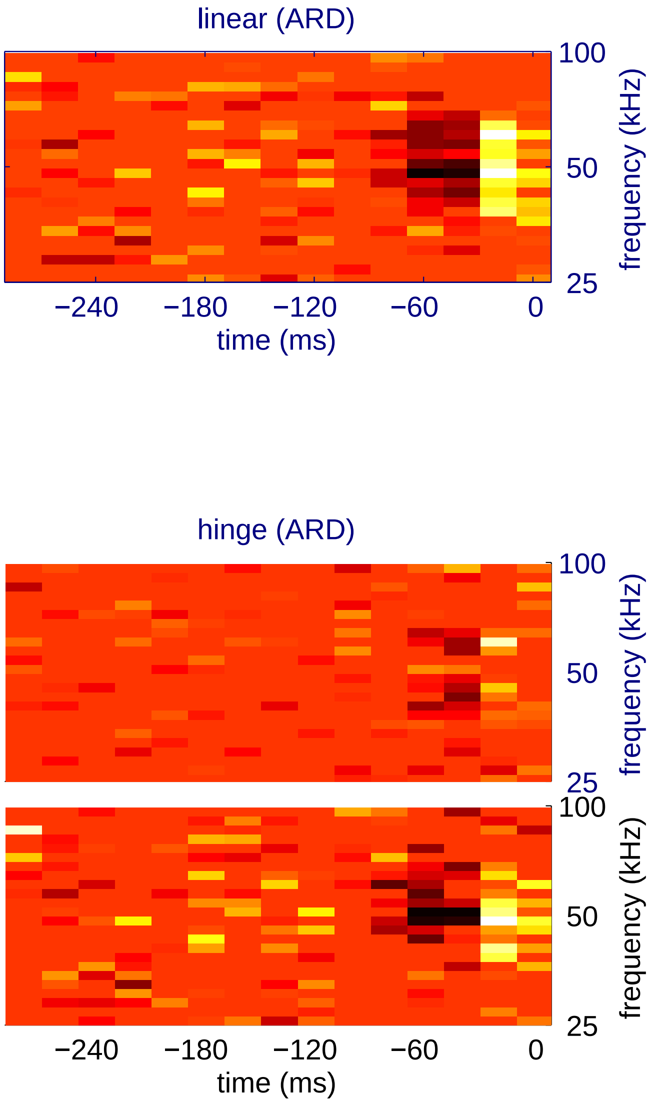




R2001011802G/20010731/pen14loc2poisshical020

Hinge weights example 2





J2001040902G/20010717/pen8loc3poisshical008

Conclusions

- The linear regression viewpoint for both linear and nonlinear finite-impulse systems identification permits the straightforward application of advanced regression techniques.
- In particular, ARD allows us to restrict the range of the filter to the relevant input subspace.
- The standard Volterra expansion for non-linear functions is a special case of the general kernel regression method.
- Other kernels can provide equally, if not more, interpretable representations of the nonlinearity, particularly when combined with ARD.