Statistical analysis of neural data:
Classification-based approaches: spike-triggered averaging, spike-triggered covariance, and the linear-nonlinear cascade model*

Liam Paninski
Department of Statistics and Center for Theoretical Neuroscience
Columbia University
http://www.stat.columbia.edu/∼liam

May 6, 2007

Contents

*Many of the figures shown in this chapter are adapted directly from (Simoncelli et al., 2004).

0.1 Introduction: estimating spiking probabilities one bin at a time

Before we attack the full neural coding problem of learning the full high-dimensional $p(\vec{n}|\vec{x})$, where $\vec{n}$ is a full spike train, or multivariate spike train, etc., and $\vec{x}$ is the observed signal with which we are trying to correlate $\vec{n}$ ($\vec{x}$ could be the stimulus, or observed motor behavior, etc.), it is conceptually easier to begin by trying to predict the scalar $p(n(t)|\vec{x})$, i.e., to predict the spike count in a single time bin t. From a statistical modeling point of view, we will therefore begin by discussing a simple first-order model for $p(\vec{n}|\vec{x})$:

$$p(\vec{n}|\vec{x}) = \prod_t p(n(t)|\vec{x}),$$

i.e., the responses $n(t)$ in each time bin are conditionally independent given the observed $\vec{x}$. (This model is typically wrong but it's a useful place to start; later we'll discuss a variety of ways to relax this conditional independence assumption (Paninski et al., 2004; Truccolo et al., 2005).)

Understanding $p(n(t)|\vec{x})$ is already a hard problem, due to the high dimensionality of $\vec{x}$, and the fact that, of course, we only get to observe a noisy version of this high-dimensional function of $\vec{x}$.

In the simplest case, we may take dt, the width of the time bin in which $n(t)$ is observed, to be small enough that only at most one spike is observed per time bin. Then estimating $p(n(t) = 1|\vec{x})$ is equivalent to estimating $E(n(t)|\vec{x})$.

0.2 Nonparametric smoothing of spike responses is useful in low-dimensional cases

We may begin by attempting to estimate this function $E(n|\vec{x})$ nonparametrically: this approach is attractive because it requires us to make fewer assumptions about the shape of $E(n|\vec{x})$ as a function of $\vec{x}$ (although as we will see, we still have to make some kind of assumption about how sharply $E(n|\vec{x})$ is allowed to vary as a function of $\vec{x}$). One simple method is based on kernel density estimation (Hastie et al., 2001; Devroye and Lugosi, 2001): we form the estimate

$$\hat{E}(n|\vec{x}) = \frac{\sum_t w(\vec{x}_t - \vec{x})n}{\sum_t w(\vec{x}_t - \vec{x})},$$

where $w(.)$ is a suitable smoothing kernel; typically, $w(.)$ is chosen to be positive, integrable with respect to $\vec{x}$, and elliptically symmetric[1] in $\vec{x}$ about $\vec{x} = 0$. See Fig. 1 for an illustration in the case that $\vec{x}$ is one-dimensional. A related approach is to simply form a histogram for $\vec{x}$, and set $\hat{E}(n|\vec{x})$ to be the mean of $n(t)$ for all time points t for which the corresponding $\vec{x}_t$ fall in the given histogram bin (Chichilnisky, 2001); see Fig. 2 for an example.

The wider $w(.)$ (or equivalently, the histogram bin) is chosen to be, the smoother the resulting estimate $\hat{E}(n|\vec{x})$ becomes; thus it is common to use an adaptive approach in the choice of $w(.)$, where $w(.)$ is chosen to be wider in regions of the $\vec{x}$-space where there are fewer samples $\vec{x}_t$ (where more smoothing is necessary) and narrower in regions where more data are available.

[1] A function $g(\vec{x})$ is "elliptically symmetric" if $g(\vec{x}) = q(||A\vec{x}||_2)$ for some scalar function $q(.)$, some symmetric matrix A, and the usual two-norm $||\vec{y}||_2 = (\sum_i y_i^2)^{1/2}$; that is, g is constant on the ellipses defined by fixing $||A\vec{x}||_2$; the contour curves of such an elliptically symmetric function are ellipses (hence the name), with the eigenvectors of A corresponding to the major axes of the ellipse. A function is "radially," or "spherically," symmetric if A is proportional to the identity, in which case the elliptic symmetries above become spherical.

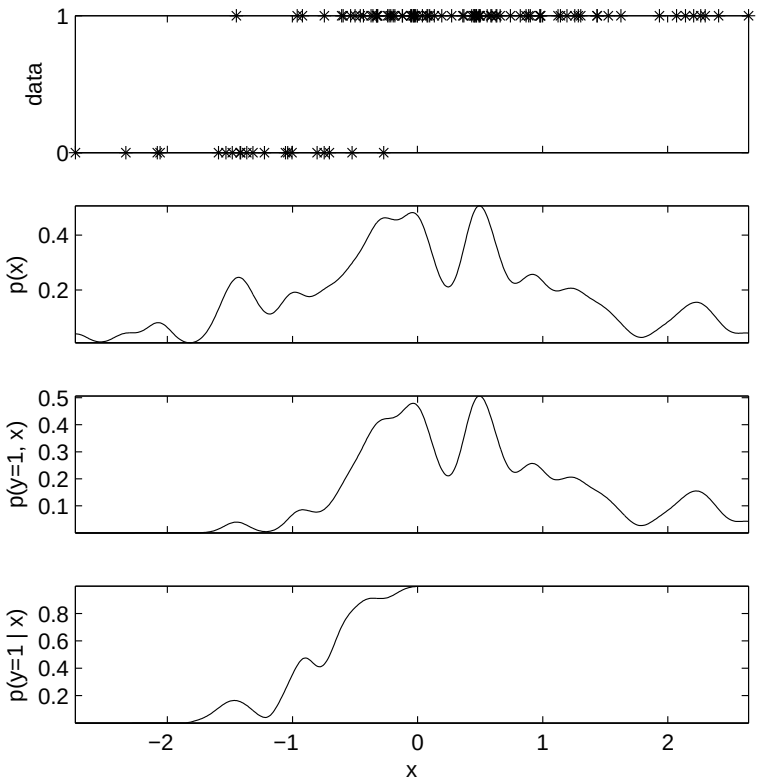

Figure 1: Illustration of the Gaussian smoothing kernel applied to simulated one-dimensional data x. **Top**: observed binary data. **Second panel**: Estimated density $\hat{p}(x) = \frac{1}{N}\sum_{i=1}^{N} w(x_i - x)$, with the smoother $w(.)$ chosen to be Gaussian with mean zero and standard deviation .1. **Third panel**: Estimated joint density $\hat{p}(x, y = 1) = \frac{1}{N}\sum_{i=1}^{N} 1(y_i = 1)w(x_i - x)$. **Bottom**: Estimated conditional density $\hat{p}(y = 1|x) = \hat{p}(y = 1, x)/\hat{p}(x)$.

This simple smoothing approach is quite effective in the case that $\dim(\vec{x}) \leq 2$, where it is possible to visualize the estimated function $\hat{E}(n|\vec{x})$ directly. However, for higher-dimensional $\vec{x}$ this approach becomes less useful, in effect because the number of samples needed to "fill in" a multidimensional histogram scales exponentially with $d = \dim(\vec{x})$: this is one example of the so-called "curse of dimensionality" (Duda and Hart, 1972; Hastie et al., 2001).

0.3 Multiple linear regression approaches

A great variety of more involved nonparametric approaches have ben developed in the statistics and machine learning community (Hastie et al., 2001). However, the approach emphasized here will be more model-based; this makes the results somewhat easier to interpret, and more importantly, allows us to build in more about what we know about biophysics, functional neuroanatomy, etc.

The simplest model-based approach is to employ classical linear multiple regression. We model $n(t)$ as

$$n(t) = \vec{k}^T \vec{x}_t + b + \epsilon_t,$$

where ϵ_t is taken to be an independent and identically distributed (i.i.d.) random variable with mean zero and variance $Var(\epsilon_t) = \sigma^2$. The solution to the problem of choosing the parameters $(\vec{k}, b)$ to minimize the mean-square error

$$\sum_t [\vec{k}^T \vec{x}_t + b - n(t)]^2$$

is well-known (Kutner et al., 2005): the best-fitting parameter vector $\hat{\theta}_{LS} = (\vec{k}^T \;\; b)^T_{LS}$ satisfies the "normal equations"

$$(X^T X)\hat{\theta}_{LS} = X^T \vec{n},$$

where the matrix X is defined as

$$X_t = (\vec{x}_t^T \;\; 1)$$

and

$$\vec{n} = \begin{pmatrix} n(1) & n(2) & \dots & n(t) \end{pmatrix}^T.$$

The normal equations are derived by simply writing the mean square error in matrix form,

$$\sum_t [\vec{k}^T \vec{x}_t + b - n(t)]^2 = ||X\theta - \vec{n}||_2^2 = \theta^T X^T X \theta - 2\theta^T X^T \vec{n} + \vec{n}^T \vec{n},$$

and setting the gradient with respect to the parameters $\theta = (\vec{k}^T \;\; b)^T$ equal to zero. In the case that the matrix $X^T X$ is invertible, we have the nice explicit solution

$$\hat{\theta}_{LS} = (X^T X)^{-1} X^T \vec{n};$$

more generally the solution to the normal equations is nonunique, and typically $(X^T X)^{-1}$ in the expression above is replaced by the Moore-Penrose pseudoinverse (i.e., $\hat{\theta}_{LS}$ is chosen to be the solution to the normal equations with the smallest squared 2-norm $||\theta||_2^2 = \sum_i \theta_i^2$; this makes the solution unique, since the squared 2-norm is strictly convex (Boyd and Vandenberghe, 2004) and the set of solutions to any linear equation is convex).

So the linear regression approach leads to a nice, computationally-tractable solution; moreover, the statistical properties of the estimated parameters $\hat{\theta}_{LS}$ are very well-understood: we can construct confidence intervals and do hypothesis testing using standard, well-defined techniques (again, see (Kutner et al., 2005) for all details).

Moreover, the components of the solution $(X^T X)^{-1} X^T \vec{n}$ turn out to have some useful, straightforward interpretations. For example,

$$X^T \vec{n} = \left(\sum_t \vec{x}_t^T n(t) \quad \sum_t n(t) \right)^T;$$

forming the quotient of the two terms on the right, $[\sum_t \vec{x}_t n(t)]/[\sum_t n(t)]$, gives us the spike-triggered average (de Boer and Kuyper, 1968) — the conditional mean $\vec{x}$ given a spike — about which we will have much more to say in a moment. Similarly, the matrix $X^T X$ contains all the information we need to compute the correlation matrix of the stimulus.

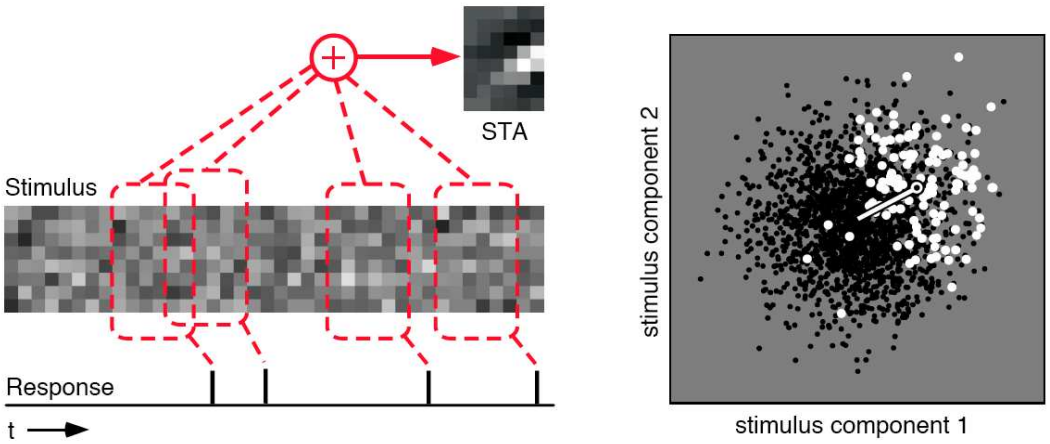

Figure 2: Two alternative illustrations of the reverse correlation procedure. **Left**: Discretized stimulus sequence and observed neural response (spike train). On each time step, the stimulus consists of an array of randomly chosen values (eight, for this example), corresponding to the intensities of a set of individual pixels, bars, or any other spatial patterns. The neural response at any particular moment in time is assumed to be completely determined by the stimulus segment that occurred during a pre-specified interval in the past. In this example, the segment covers six time steps, and lags three time steps behind the current time (to account for response latency). The "spike-triggered ensemble" (de Ruyter van Steveninck and Bialek, 1988) consists of the set of segments associated with spikes. The spike-triggered average (STA) is constructed by averaging these stimulus segments. **Right**: Geometric interpretation of the STA. Each stimulus segment corresponds to a point in a vector space space (in this example, the dimensionality of this stimulus space is $6 \times 8 = 48$) whose axes correspond to stimulus values (e.g., pixel intensities) during the interval. For illustration purposes, the scatter plot shows only two of the 48 axes. The spike-triggered stimulus segments (white points) constitute a subset of all stimulus segments presented (black points). The STA, indicated by the line in the diagram, corresponds to the difference between the mean (center of mass) of the spike-triggered ensemble, and the mean of the raw stimulus ensemble.

0.4 Nonlinear regression approaches

However, it is not clear that this linear regression model captures neural responses very well. Moreover, departures from the assumptions of the model might bias our estimates of the model parameters, or reduce the interpretability of the results.

A few such departures are obvious, even necessary; for example, the spike count $n(t)$, and therefore $E(n|\vec{x})$, must be nonnegative. More importantly, the function $E(n|\vec{x})$ may be quite nonlinear, reflecting saturation, rectification, adaptation effects, etc. It is straightforward to include nonlinear terms in the regression analysis (Sahani, 2000; Kutner et al., 2005), simply by redefining the matrix X appropriately: instead of letting the t-th row X_t contain just the elements of $\vec{x}$ and 1, we may also include arbitrary functionals $\mathcal{F}_i(\vec{x}_t)$:

$$X_t = \begin{pmatrix} \vec{x}^T & \mathcal{F}_1(\vec{x}_t) & \mathcal{F}_2(\vec{x}_t) & \ldots & \mathcal{F}_m(\vec{x}_t) & 1 \end{pmatrix}.$$

The resulting model of the response is now nonlinear:

$$n(t) = \vec{k}^T \vec{x} + \sum_{i=1}^{m} a_i \mathcal{F}_i(\vec{x}) + b + \epsilon_t,$$

with the least-squares parameters $(\vec{k}, \vec{a}, b)_{LS}$ determined by solving the normal equations (with the suitably redefined X) exactly as in the fully linear case.

We still need to make sure that the predicted firing rate $E(n|\vec{x})$ remains nonnegative. This nonnegativity constraint may be enforced with a collection of linear inequality constraints

$$\vec{k}^T \vec{x} + \sum_{i=1}^{m} a_i \mathcal{F}_i(\vec{x}) + b \geq 0 \quad \forall \vec{x},$$

(i.e., one constraint for each value of $\vec{x}$; note that each constraint is linear as a function of the parameters $(\vec{k}, \vec{a}, b)$, despite the nonlinearity in $\vec{x}$). This converts the original unconstrained quadratic regression problem into a quadratic program[2], which retains much of the tractability of the original problem (we will return to quadratic programs in much more depth soon (Huys et al., 2006; Paninski, 2006; Nikitchenko and Paninski, 2007)).

This nonlinear regression approach is useful in a number of contexts. One example involves the incorporation of known presynaptic nonlinearities: if we know that the neuron of interest receives input from presynaptic neurons which perform some well-defined nonlinear transformation on the stimulus $\vec{x}$, it is worth incorporating this knowledge into the model (Rust et al., 2006).

Another common application is a kind of polynomial expansion referred to as a "Volterra-Wiener" series (Marmarelis and Marmarelis, 1978). The N-th order Volterra expansion involves all polynomials in $\vec{x}$ up to the N-th order: thus the zero-th order model is

$$n(t) = b + \epsilon_t,$$

with a corresponding design matrix

$$X_t = (1);$$

the first order expansion is the linear model discussed above ($n(t) = b + \vec{k}^T \vec{x}_t + \epsilon_t$); the second-order model is

$$n(t) = b + \vec{k}^T \vec{x}_t + \sum_{ij} a_{ij} \vec{x}_t(i) \vec{x}_t(j) + \epsilon_t,$$

with

$$X_t = \left(1 \quad \vec{x}_t^T \quad \vec{x}_t(1)\vec{x}_t(1) \quad \vec{x}_t(2)\vec{x}_t(1) \quad \vec{x}_t(3)\vec{x}_t(1) \quad \dots \quad \vec{x}_t(2)\vec{x}_t(2) \quad \dots \quad \vec{x}_t(d)\vec{x}_t(d)\right),$$

while the third-order model includes all triplet terms $\vec{x}(i)\vec{x}(j)\vec{x}(l)$, and so on. The attraction of these expansion-based models is that, in principle, we may approximate an arbitrary smooth

[2] A quadratic program (QP) is a linearly-constrained quadratic optimization problem of the form

$$\max_{\theta} \frac{1}{2}\theta^T A \theta + a^T \theta, \ a_i^T \theta \geq c_i \quad \forall i,$$

for some negative semidefinite matrix A and some collection of vectors a and a_i and corresponding scalars c_i. Quadratic programs are do not in general have analytic solutions, but we may numerically solve a QP roughly as quickly as we may solve the unconstrained problem $\max_\theta \frac{1}{2}\theta^T A \theta + a^T \theta$, since we are maximizing a particularly simple concave function on a particularly simple convex space.

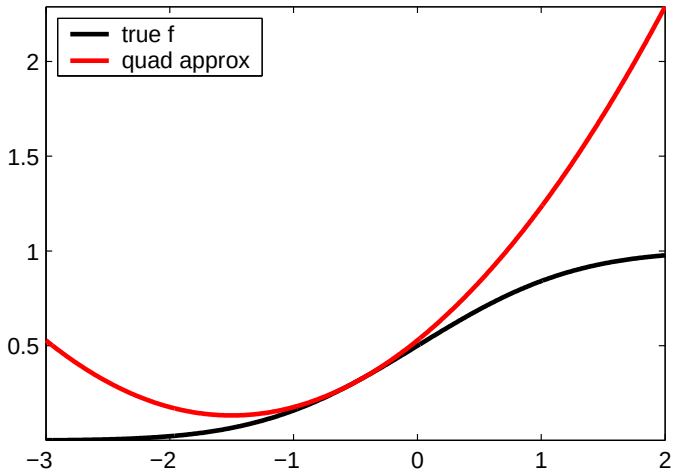

Figure 3: A simple toy example illustrating some flaws in the Volterra expansion approach. In this case we are approximating the true firing rate function $f(.)$ by its second-order Taylor series. The problem here is that the function $f(x)$ saturates for large values of x, while of course the x^2 term increases towards infinity, thus making a poor approximation.

function $E(n|\vec{x})$ by using a sufficiently large expansion order N, while the order N provides a natural, systematic index of the complexity of the model.

However, several problems are evident in this nonlinear regression approach. The key problem is that it is often difficult to determine *a priori* what nonlinearities $\mathcal{F}(\vec{x})$ to include in the analysis. In the Volterra-Wiener approach described above, for example, the polynomial expansion works poorly to approximate a saturating function $E(n|\vec{x})$, in the sense that a large N is required to obtain a reasonable degree of accuracy, and (more importantly) the resulting approximation is unstable, with delicately balanced oscillatory terms and unbounded behavior at the boundary of the $\vec{x}$ space (poor extrapolation). In general, moreover, the number of terms required in the expansion scales unfavorably with both the expansion order N and the dimension d of $\vec{x}$. A complementary problem is that the inclusion of many terms in any regression model will lead to overfitting effects (we will discuss this at much more length soon (Machens et al., 2003; Smyth et al., 2003; Paninski, 2004)): that is, poor generalization ability even in cases when the training error may be made small.

0.5 The linear-nonlinear cascade model

An alternate approach to enforcing the nonnegativity of the regression function $E(n|\vec{x})$ is to retain the initial linear structure of the model, but then to apply a nonnegative post-nonlinearity:

$$n(t) = f(\vec{k}^T\vec{x}_t + b) + \epsilon_t$$

(where the distribution of ϵ_t might in general also depend on $\vec{x}_t$). Models of this form are referred to as "cascade" models in the engineering literature, as a linear filtering stage is appended in a processing cascade to a nonlinear processing stage.

It has been argued (Chichilnisky, 2001; Simoncelli et al., 2004) that this simple linear-nonlinear (LN) cascade is a natural conceptual model for neural activity: the linear term

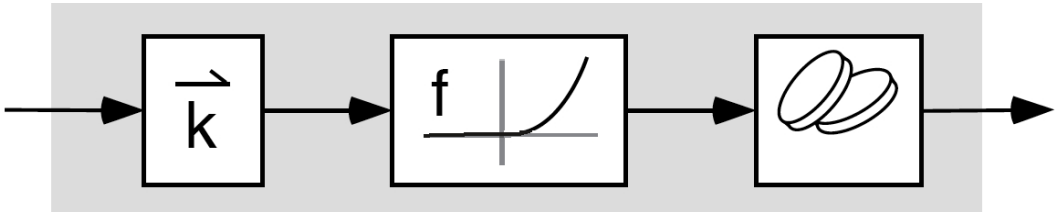

Figure 4: Block diagram of the linear-nonlinear cascade model. On each time step, the components of the stimulus vector are linearly combined using a weight vector, $\vec{k}$. The output of this linear filter is then passed through a nonlinear function $f(.)$, whose output determines the instantaneous firing rate of a Brenoulli ("coin-flip") spike generator, $E(n|\vec{x}) = f(\vec{k}^T\vec{x})$.

corresponds to dendritic filtering and the lumped effects of processing presynaptic to the neuron of interest, while the nonlinearity $f(.)$ corresponds to the translation of the subthreshold somatic voltage into a (nonnegative) firing rate. (Of course, more complex cascades, with additional nonlinearity and linear filtering stages, have also been explored (Hunter and Korenberg, 1986; Ahrens et al., 2006); we will discuss a few specific examples below.)

How do we estimate the parameters of this cascade model? First we must define the parameters somewhat more clearly: if the nonlinearity f and the distribution of the errors ϵ_t are both fully known, then we may apply standard maximum-likelihood approaches to obtain the filtering and offset parameters $\vec{k}$ and b. We will explore this approach in quite a bit of detail shortly.

However, first it is useful to discuss a different approach which is perhaps slightly more intuitive. Let's assume that

$$n(t) \sim Bernoulli[f(\vec{x}^T\vec{x}_t)],$$

i.e., at each time t we flip a coin with bias $f(\vec{x}^T\vec{x}_t)$ in order to determine whether $n(t) = 1$ or 0, where both the linear filter $\vec{k}$ and the scalar nonlinearity $f(.)$ are considered unknown parameters we need to fit to data. (We have eliminated the constant offset b for the moment, since this can be absorbed in the definition of the unknown function $f(.)$. In addition, for the same reason, we need only estimate the direction of $\vec{k}$, since the magnitude of $\vec{k}$ may be absorbed in the definition of $f(.)$.)

As emphasized above, if we know $f(.)$, then $\vec{k}$ is conceptually easy to estimate by maximum likelihood (though actually finding the ML solution — i.e., solving the likelihood optimization problem — may be computationally difficult). Conversely, if we know $\vec{k}$, then $f(.)$ may be estimated easily by some version of the nonparametric smoothing idea described above: since $\vec{k}^T\vec{x}$ is a one-dimensional variable, it is easy to estimate $E(n|\vec{k}^T\vec{x})$ using these methods (for a detailed description of a histogram-based version of this approach, see (Berry and Meister, 1998; Chichilnisky, 2001); see also Fig. 5). Estimating $\vec{k}$ (a finite-dimensional parameter) and $f(.)$ (an infinite-dimensional parameter) at once is what is known as a "semiparametric" problem in the statistics literature (Bickel et al., 1998).

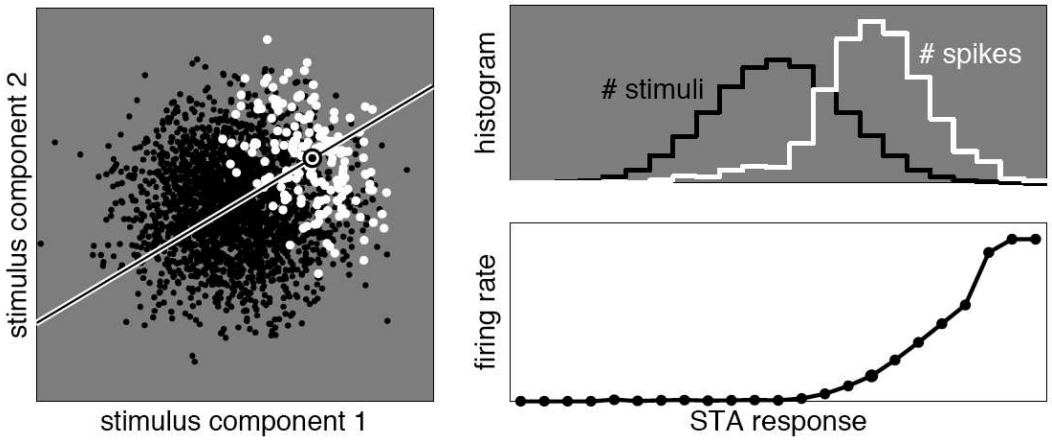

Figure 5: Simulated characterization of an LNP model using reverse correlation. The simulation is based on a sequence of $20,000$ stimuli, with 950 spikes. **Left**: The STA (black and white "target") provides an estimate of the linear weighting vector, $\vec{k}$ (see also Figure 2). The linear response to any particular stimulus corresponds to the position of that stimulus along the axis defined by $\vec{k}$ (line). **Right, top**: raw (black) and spike-triggered (white) histograms of the linear (STA) responses. **Right, bottom**: The quotient of the spike-triggered and raw histograms, $\hat{p}(n = 1|\vec{k}^T\vec{x}) = \hat{p}(n = 1, \vec{k}^T\vec{x})/\hat{p}(\vec{k}^T\vec{x})$, gives an estimate of the nonlinearity $f(.)$ that generates the firing rate.

0.6 The Fisher linear discriminant is a classical technique for separating two clouds of data points, and is closely related to the spike-triggered average

To begin, it is useful to think about this problem of estimating $\vec{k}$ as a classification problem: we want to classify the stimuli $\vec{x}$ which led to a spike versus those which did not lead to a spike. Conceptually, we have two clouds of data points in the $\vec{x}$ space: one cloud corresponds to the spike-triggered ensemble, and the other cloud corresponds to those stimuli which did not elicit a spike. Our goal is to find a projection $\vec{k}$ which separates these clouds as efficiently as possible, in some sense.

One classical way to find a good separating projection of two clouds of data points in a high-dimensional space is the so-called Fisher linear discriminant (Duda and Hart, 1972; Metzner et al., 1998). We begin by defining a simple measure of separation in terms of the mean and variance of the projected data:

$$J(\vec{k}) = \frac{[E(\vec{k}^T\vec{x}|n = 0) - E(\vec{k}^T\vec{x}|n = 1)]^2}{p(n = 0)Var(\vec{k}^T\vec{x}|n = 0) + p(n = 1)Var(\vec{k}^T\vec{x}|n = 1)}.$$

The numerator measures the separation of the projected means of each of the two classes ($n = 1$ indicates the spike-triggered ensemble, and $n = 0$ is the no-spike data cloud), while the denominator normalizes by the variance of the two projected distributions. (Note in particular that $J(\vec{k})$ is a function only of the direction of $\vec{k}$: $J(\vec{k}) = J(c\vec{k})$ for any nonzero constant c.)

Optimizing $J(\vec{k})$ as a function of $\vec{k}$ requires that we rewrite this objective function in a more linear-algebraic form, to make it easier to take the gradient with respect to $\vec{k}$. It is easy

to see that

$$J(\vec{k}) = \frac{\vec{k}^T S_B \vec{k}}{\vec{k}^T S_W \vec{k}},$$

where we have defined the "between-class" scatter matrix

$$S_B = [E(\vec{x}|n=1) - E(\vec{x}|n=0)][E(\vec{x}|n=1) - E(\vec{x}|n=0)]^T$$

and the "within-class" scatter marix

$$S_W = p(n=0)Cov(\vec{x}|n=0) + p(n=1)Cov(\vec{x}|n=1).$$

It is well-known that the solution to an optimization problem of this Rayleigh quotient form solves a generalized eigenvector problem: the optimal $\vec{k}$ must solve

$$S_B \vec{k} = \lambda S_W \vec{k}, \tag{1}$$

for λ equal to the top generalized eigenvalue[3]. If S_W is invertible (an acceptably weak assumption in this case), we can multiply both sides by S_W^{-1}, reducing the equation to a standard eigenvector problem in terms of the matrix $S_W^{-1} S_B$:

$$S_W^{-1} S_B \vec{k} = \lambda \vec{k}.$$

Finally, recalling that S_B has rank one, we may find the solution analytically:

$$\vec{k}_{FLD} = S_W^{-1}[E(\vec{x}|n=1) - E(\vec{x}|n=0)],$$

since

$$\begin{aligned}
S_W^{-1} S_B \vec{k}_{FLD} &= S_W^{-1} S_B S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0) \\
&= S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0)(\vec{\mu}_1 - \vec{\mu}_0)^T S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0) \\
&= \left[(\vec{\mu}_1 - \vec{\mu}_0)^T S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0)\right] S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0) \\
&= \left[(\vec{\mu}_1 - \vec{\mu}_0)^T S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0)\right] \vec{k}_{FLD} \\
&= \lambda \vec{k}_{FLD},
\end{aligned}$$

where we have abbreviated $\vec{\mu}_i = E(\vec{x}|s=i)$, and $\lambda = (\vec{\mu}_1 - \vec{\mu}_0)^T S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0)$.

Interestingly, this solution has a very similar form to the optimal solution we obtained in the linear regression context. In fact, in one important special case, the solutions are exactly the same. Assume that we have subtracted off the mean of X and $\vec{n}$: that is, $\sum_t X_t$ and $\sum_t n(t)$ are both zero (this entails no loss of generality). Then it is known (Kutner et al.,

[3]To see this, note that we can break up our optimization problem into two steps:

$$\max_{\vec{k}} J(\vec{k}) = \max_c \left[\frac{1}{c} \max_{\vec{k}:\vec{k}^T S_W \vec{k} = c} \vec{k}^T S_B \vec{k} \right].$$

The inner problem may be solved by the method of Lagrange multipliers: we must have

$$\arg \max_{\vec{k}:\vec{k}^T S_W \vec{k} = c} \vec{k}^T S_B \vec{k} = \arg \max_{\vec{k}} \vec{k}^T S_B \vec{k} - \lambda \vec{k}^T S_W \vec{k},$$

for some value of λ. By differentiating and setting the expression on the right to zero we obtain equation (1). We will encounter similar optimization problems in a few other contexts (Lewi et al., 2006).

2005) that the optimal value of the offset term b in the parameter vector $\theta = (\vec{k} \;\; b)$ is always zero, and so we can remove this term (and the corresponding column of ones in X) from the analysis. Then $X^T X$ is proportional to $S_W + aS_B$, for a suitable scalar a. Moreover, in this case the vector $\vec{\mu}_1 - \vec{\mu}_0$ is parallel to $X^T\vec{n}$. Thus the regression solution $(X^T X)^{-1}(X^T\vec{n})$ is proportional to the Fisher linear discriminant solution $S_W^{-1}[E(\vec{x}|n=1) - E(\vec{x}|n=0)]$ here, as can be seen by the Woodbury matrix lemma[4]:

$$\begin{aligned}
\vec{k}_{LS} &= (X^T X)^{-1}(X^T\vec{n}) \\
&\propto [S_W + a(\vec{\mu}_1 - \vec{\mu}_0)(\vec{\mu}_1 - \vec{\mu}_0)^T]^{-1}(\vec{\mu}_1 - \vec{\mu}_0) \\
&= \left[S_W^{-1} - S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0)\left(\frac{1}{a} + (\vec{\mu}_1 - \vec{\mu}_0)^T S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0)\right)(\vec{\mu}_1 - \vec{\mu}_0)^T S_W^{-1}\right](\vec{\mu}_1 - \vec{\mu}_0) \\
&\propto S_W^{-1}(\vec{\mu}_1 - \vec{\mu}_0) \\
&= \vec{k}_{FLD}.
\end{aligned}$$

Similar formulas arise in one more very familiar context: imagine we are asked to discriminate samples from one of two high-dimensional Gaussian distributions with equal covariance but different means (i.e., imagine that both the spike-triggered and non-spike-triggered ensembles consist of samples from two different Gaussian distributions with covariance C). In the language of simple hypothesis testing, the null hypothesis is

$$H_0 : \vec{x} \sim \mathcal{N}(\mu_0, C),$$

and the alternate hypothesis

$$H_1 : \vec{x} \sim \mathcal{N}(\mu_1, C).$$

Then, according to the Neyman-Pearson lemma (Schervish, 1995), the most powerful test is based on the likelihood ratio

$$\frac{p(\vec{x}|H_0)}{p(\vec{x}|H_1)} = \frac{(2\pi)^{-d/2}|C|^{-1/2}\exp\left(-\frac{1}{2}(\vec{x}-\mu_0)^T C^{-1}(\vec{x}-\mu_0)\right)}{(2\pi)^{-d/2}|C|^{-1/2}\exp\left(-\frac{1}{2}(\vec{x}-\mu_0)^T C^{-1}(\vec{x}-\mu_0)\right)} = c\exp[\vec{x}^T C^{-1}(\mu_0 - \mu_1)],$$

where c is a constant that does not depend on $\vec{x}$. Thus the optimal discrimination once again requires only the projection of $\vec{x}$ onto $C^{-1}(\mu_0 - \mu_1)$.

The Fisher discriminant objective function is based on just a few simple statistics (specifically, the first and second moments) of the relevant distribution $p(n, \vec{x})$. Clearly other objective functions are possible. For example, the "maximum margin" objective function

$$\min_{\vec{x}_0 \in \mathcal{X}_0, \vec{x}_1 \in \mathcal{X}_1} \vec{k}^T(\vec{x}_1 - \vec{x}_0); \;\; ||\vec{k}||_2 \leq 1$$

(where $\mathcal{X}_i$ is the set of $\vec{x}_t$ for which $n(t) = i$) is popular and very well-studied in the machine-learning literature (for example, the popular "support vector machine" algorithm for classification is based on this maximum margin framework (Scholkopf and Smola, 2002)). This

[4]The Woodbury matrix lemma is a handy trick for computing the inverse of a matrix of the form $A+UCV$, where A is a matrix whose inverse we already know and UCV is a matrix of low rank (here, if A is $d \times d$, then U is $d \times m$, C is $m \times m$, and V is $m \times d$, for some m). We have

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}.$$

While the right-hand side looks much more complicated, in fact if $m \ll d$ the right-hand side is much easier to compute (because we only need to compute a $m \times m$ inverse instead of an $d \times d$ inverse; the left-hand side is computed in $O(d^3)$ time, while the right requires $O(d^2 + m^3)$). See any text on matrix algebra for a proof.

margin-based objective function has certain advantages over the Fisher approach in machine learning settings (especially in cases when the distributions $p(\vec{x}|n)$ are poorly summarized by the first two moments, i.e., highly non-Gaussian $p(\vec{x}|n)$). In particular, the max-margin approach is competitive when the number of features is large comparable to the number of observed data points (i.e., the width of the matrix X, d, is large comparable to the height, N), due to the "representer theorem," which states that it is possible to restrict the search to a subspace of dimension N, instead of d (Scholkopf and Smola, 2002) (we will have more to say about this result later (Pillow and Paninski, 2007)). However, we will not explore this approach in depth here, because when $N >> d$ (as is typically the case for the neural examples of interest), moment-based approaches are more computationally tractable.

0.7 The spike-triggered average gives an unbiased estimate under certain symmetry conditions

So we have seen that the vector $C^{-1}(\mu_0 - \mu_1)$ arises in a variety of classification contexts. Interestingly, it turns out that this spike-triggered average-based solution turns out to provide a good estimator for $\vec{k}$ in the LN cascade model under certain conditions, even though it does not require estimation of the nonlinear function $f(.)$. For example, if the distribution of $p(\vec{x})$ is elliptically symmetric, then this estimator is unbiased for $\vec{k}$.[5] The proof of this fact is quite straightforward (Chichilnisky, 2001; Paninski, 2003; Schnitzer and Meister, 2003; Simoncelli et al., 2004), relying on the fact that the expectation

$$\int p(\vec{x}) g(\vec{k}^T \vec{x}) d\vec{x} \propto \vec{k}$$

whenever $p(\vec{x})$ is radially symmetric. This may be seen by a simple symmetry argument (Chichilnisky, 2001): since the function $g(\vec{k}^T\vec{x})p(\vec{x})$ is symmetric around the axis $\vec{k}$ (by the symmetry assumption on $p(\vec{x})$), the average $\int p(\vec{x})g(\vec{k}^T\vec{x})d\vec{x}$ of this function must lie on the axis spanned by $\vec{k}$[6].

Thus, if $p(\vec{x})$ satisfies the elliptical symmetry condition $p(\vec{x}) = q(||A\vec{x}||_2)$ for some nonsin-

[5] An estimator $\hat{\theta}$ for a parameter θ is unbiased if $E_\theta \hat{\theta} = \theta$ for all values of θ. In this case, $\vec{k}$ is considered as a one-dimensional subspace rather than as a vector: thus, we mean that $E(\hat{k})$ is proportional to $\vec{k}$ when the data are generated according to the LN model with parameter $\vec{k}$.

[6] A nice generalization to the multineuronal "multispike-triggered average" is described by (Schnitzer and Meister, 2003). Imagine we are observing two neurons simultaneously, and both neurons respond to the stimulus as independent linear-nonlinear encoders:

$$p(n_1 = 1, n_2 = 1|\vec{x}) = f_1(\vec{k_1}^T\vec{x}) f_2(\vec{k_2}^T\vec{x}),$$

where $\vec{k}_1$ and $\vec{k}_2$ denote the receptive fields of cell 1 and 2, respectively. Now if $p(\vec{x})$ is radially symmetric, then an identical argument establishes that the multi-spike triggered average $E(\vec{x}|n_1 = 1, n_2 = 1)$ must lie in the subspace spanned by $\vec{k}_1$ and $\vec{k}_2$, i.e.,

$$E(\vec{x}|n_1 = 1, n_2 = 1) = a_1\vec{k}_1 + a_2\vec{k}_2$$

for some scalars a_1 and a_2. Thus if we measure $E(\vec{x}|n_1 = 1, n_2 = 1)$ experimentally and find significant departures from the subspace spanned by $\vec{k}_1$ and $\vec{k}_2$, we can reject the hypothesis that both neurons respond to the stimulus as independent linear-nonlinear encoders. This argument may clearly be extended to more than two neurons.

gular symmetric matrix A and some scalar function $q(.)$, we may compute

$$\begin{aligned}
\frac{1}{N}E(X^T\vec{n}) &= \int p(\vec{x}|n=1)\vec{x}d\vec{x} \\
&= \int p(n=1|\vec{x})\frac{p(\vec{x})}{p(n=1)}\vec{x}d\vec{x} \\
&= \int f(\vec{k}^T\vec{x})\frac{p(\vec{x})}{p(n=1)}\vec{x}d\vec{x} \\
&= \int A^{-1}\vec{y}\frac{f(\vec{k}^TA^{-1}\vec{y})}{p(n=1)}p(A^{-1}\vec{y})|A|d\vec{y} \\
&= A^{-1}\int \vec{y}\frac{f(\vec{k}^TA^{-1}\vec{y})}{p(n=1)}p(A^{-1}\vec{y})|A|d\vec{y} \\
&\propto A^{-1}\left(A^{-1}\vec{k}\right) = A^{-2}\vec{k}.
\end{aligned}$$

The first three equalities are by definition and Bayes, the fourth a linear change of coordinates $y = Ax$, and the last by the symmetry argument above, since $\vec{k}^TA^{-1}\vec{y} = (A^{-1}\vec{k})^T\vec{y}$. Thus if we know A *a priori*, we may construct an unbiased estimator by simply taking

$$\hat{k} = A^2X^T\vec{n},$$

i.e., a simple linear transformation of the STA. Thus this linearly-transformed spike-triggered average estimator $A^2X^T\vec{n}$ may be considered a method-of-moments estimator (Schervish, 1995) for $\vec{k}$ (later we will discuss more efficient likelihood-based approaches (Paninski, 2004)). An application of the strong law of large numbers is enough to establish consistency for this estimator (we may moreover establish rate of convergence results and a central limit theorem for $\hat{k}$ easily; see (Paninski, 2003) for details). More generally we have to estimate A from data, but since we typically may collect or generate an arbitrarily large number of samples from $p(\vec{x})$, this is straightforward: if $p(\vec{x})$ has zero mean, as we may assume without loss of generality, then the sample second moment matrix $\hat{E}(\vec{x}\vec{x}^T)$ is a consistent estimator for A^{-2}, and therefore our familiar

$$\hat{k} = (X^TX)^{-1}X^T\vec{n}$$

is consistent for $\vec{k}$ — without exact knowledge of $f(.)$.

It is worth emphasizing that the elliptical symmetry condition on $p(\vec{x})$ is not only sufficient for consistency of the STA estimator, but also necessary (Paninski, 2003), in the sense that if $p(\vec{x})$ is not elliptically symmetric, then there exists an $f(.)$ for which the STA estimator has a nonnegligible bias (i.e., is inconsistent) for the $\vec{k}$; see Fig. 6 for an illustration.

0.8 Spike-triggered covariance methods can detect symmetric nonlinearities and identify higher-dimensional nonlinearities

The spike-triggered average technique fails in two cases, even in the case that the simulus distribution $p(\vec{x})$ is elliptically symmetric. We know in this case that the expectation of the covariance-corrected STA is guaranteed to be proportional to $\vec{k}$: however, if $E(\vec{k}^T\vec{x}|n=i)$ is independent of i, then it is easy to see that this proportionality constant is zero (Paninski, 2003). This occurs, for example, whenever the nonlinearity $f(.)$ is symmetric with respect to its argument, i.e., when the neuron is sensitive only to the magnitude of the stimulus,

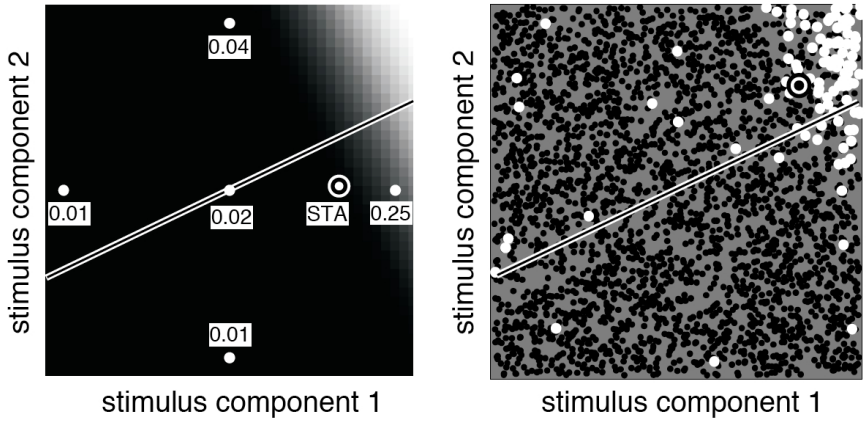

Figure 6: Simulations of an LNP model demonstrating bias in the STA for two different nonspherical stimulus distributions. The true $\vec{k}$ is indicated by the solid line in both panels, and the firing rate function $f(.)$ is a sigmoidal nonlinearity (corresponding to the intensity of the underlying grayscale image in the left panel). In both panels, the black and white "target" indicates the recovered STA. **Left**: Simulated response to "sparse" noise. The plot shows a two-dimensional subspace of a 10-dimensional stimulus space. Each stimulus vector contains a single element with a value of ± 1, while all other elements are zero. Numbers indicate the firing rate for each of the possible stimulus vectors. The STA is strongly biased toward the horizontal axis, pulled downwards by the asymmetry in $p(\vec{x})$. **Right**: Simulated response of the same model to uniformly distributed noise. The STA is now biased upwards toward the corner. Note that in both examples, the estimate will not converge to the correct answer, regardless of the amount of data collected, i.e., an asymptotic bias remains.

not the sign, as is the case for complex cells in the primary visual cortex (Simoncelli and Adelson, 1996), for example. In this case, the STA technique fails to recover any meaningful information at all (since the STA converges to zero).

We might also consider the following simple generalization of the LN model: $E(n|\vec{x}) = f(K^T\vec{x})$, where K is an m-by-d matrix and $f(.)$ is now an m-dimensional nonlinearity. Clearly in this case the STA fails to capture all of the information in K; even in the radially symmetric case, it is easy to see (by a direct generalization of our symmetry argument above) that the expectation of the STA estimate now falls within the subspace spanned by the columns of K. But is there a way to capture all the columns of K, instead of just a single linear combination?

The spike-triggered covariance (STC) technique was developed to address both of these problems (Brenner et al., 2001; Simoncelli et al., 2004; Rust et al., 2005). The basic idea is that, if the first moment of the conditional distribution $p(\vec{x}|n)$ is not different from that of the prior distribution $p(\vec{x})$, then perhaps an analysis of the second moment matrix will be more revealing. In the simplest case, assume that $\vec{x}$ has a multivariate Gaussian distribution and (after a standardizing transformation of $\vec{x} \rightarrow Cov(\vec{x})^{-1/2}[\vec{x} - E(\vec{x})]$, if necessary) that $Cov(\vec{x}) = I$. Then it is easy to see that the posterior second moment matrix $C_{post} \equiv E(\vec{x}\vec{x}^T|n = 1)$ can in general only differ from I in directions spanned by the columns of K. To see this, break $\vec{x}$ into two orthogonal components $\vec{x} = \vec{x}_K + \vec{x}_{\backslash K}$, where $\vec{x}_K$ lies

within the subspace spanned by K, and $\vec{x}_{\backslash K}$ in the orthogonal subspace, and note that

$$\begin{aligned} p(\vec{x}_{\backslash K}|\vec{x}_K, n) &= \frac{p(\vec{x}_{\backslash K}, \vec{x}_K, n)}{p(\vec{x}_K, n)} \\ &= \frac{p(\vec{x}_{\backslash K}, \vec{x}_K)p(n|\vec{x})}{p(\vec{x}_K, n)} \\ &= \frac{p(\vec{x}_{\backslash K}, \vec{x}_K)p(n|\vec{x}_K)}{p(\vec{x}_K, n)} \\ &= \frac{p(\vec{x}_{\backslash K})p(\vec{x}_K)p(n|\vec{x}_K)}{p(\vec{x}_K, n)} \\ &= \frac{p(\vec{x}_{\backslash K})p(\vec{x}_K, n)}{p(\vec{x}_K, n)} \\ &= p(\vec{x}_{\backslash K}); \end{aligned}$$

i.e., the conditional distribution in the orthogonal subspace is equal to the prior distribution, and therefore any conditional moments in this subspace must be equal to the prior moments. The key equality here — namely, $p(\vec{x}_{\backslash K}, \vec{x}_K) = p(\vec{x}_{\backslash K})p(\vec{x}_K)$ — is due to the Gaussian assumption on $p(\vec{x})$.

This suggests a straightforward principal components-based estimator for K: we compute the eigenvectors of the matrix formed by the difference $\hat{C}_{post} - I$, where $\hat{C}_{post}$ is a consistent estimator of C_{post} (typically the sample second moment matrix). Our estimator for the subspace spanned by K is now

$$\hat{K} = eig(\hat{C}_{post} - I),$$

where the operator $eig(A)$ extracts the eigenvectors of the matrix A which are significantly different from zero[7]. (Constructing a bona fide significance test in this setting is a slightly more difficult proposition; see, e.g., (Rust et al., 2005) for a bootstrap-based analysis. In general we prefer to consider this STC analysis an exploratory method, useful for identifying subspaces K in which the neuron is tuned, and therefore more qualitative definitions of significance — i.e., choose all eigenvectors corresponding to eigenvalues which appear qualitatively different in magnitude from the "bulk spectrum" (Johnstone, 2000), i.e., the remaining eigenvalues — are sufficient for our purposes.) It is straightforward to prove that this simple approach provides a consistent estimator for K in this case of a standard Gaussian $p(\vec{x})$ (Paninski, 2003).

Once K has been estimated, as before, we can estimate $E(n|K^T\vec{x})$ using either nonparametric smoothing techniques, or by fitting a parametric model to this function: the key fact is that $K^T\vec{x}$ is tractably low-dimensional compared to $\vec{x}$ ($\dim(K^T\vec{x}) = m$ instead of d).

In the case that $p(\vec{x})$ is Gaussian with nonwhite covariance C (as usual, we may assume the mean is zero without loss of generality), we once again have to correct for C. Computing

[7]Note that we can only hope to identify the column space of K, not K itself, since any nonsingular linear transformation of K may be absorbed in the definition of the nonlinear function $f(.)$, just as in the one-dimensional $\vec{k}$ case. It is also worth remembering that the eigenvectors of any symmetric matrix (e.g., $C_{post} - I$) must be orthogonal; thus the estimate $\hat{K}$ always has orthogonal columns.

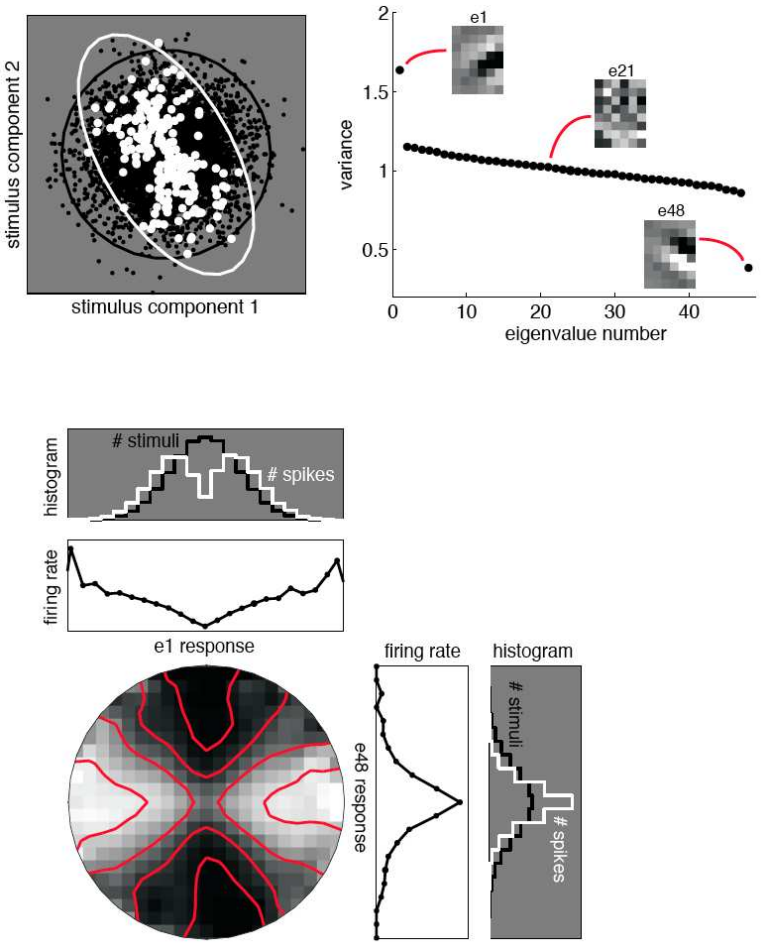

Figure 7: Simulated characterization of a two-dimensional linear-nonlinear cascade model via spike-triggered covariance (STC). In this model, the spike generator is driven by the squared response of one linear filter divided by the sum of squares of its own response and the response of another filter. As in Figure 2, both filters are 6×8, and thus live in a 48-dimensional space. The simulation is based on a sequence of $200,000$ raw stimuli, with $8,000$ spikes. **Top, left**: simulated raw and spike-triggered stimulus ensembles, viewed in the two-dimensional subspace spanned by the filters $\vec{k}_1$ and $\vec{k}_2$. The covariance of these ensembles within this two-dimensional space is represented geometrically by an ellipse that is three standard deviations from the origin in all directions. The raw stimulus ensemble has equal variance in all directions, as indicated by the black circle. The spike-triggered ensemble is elongated in one direction and compressed in another direction (white ellipse). **Top, right**: Eigenvalue analysis of the simulated data. The principal axes of the covariance ellipse correspond to the eigenvectors of the spike-triggered covariance matrix, and the associated eigenvalues indicate the variance of the spike-triggered stimulus ensemble along each of these axes. The plot shows the full set of 48 eigenvalues, sorted in descending order. Two of these are substantially different the others (one significantly larger and one significantly smaller), indicating the presence of two axes in the stimulus space along which the model is differentially responsive. Also shown are three example corresponding eigenvectors. **Bottom, one-dimensional plots**: Spike-triggered and raw histograms of responses along the two distinguished eigenvectors, along with the nonlinear firing rate functions estimated from their quotient (c.f. Fig. 2). **Bottom, two-dimensional plot**: the quotient of the two-dimensional spike-triggered and raw histograms provides an estimate of the two-dimensional nonlinear ring firate function $f(.)$. This is shown as a circular-cropped grayscale image, where intensity is proportional to firing rate. Superimposed contours (red) indicate four different response levels.

the eigenvectors of $C - C_{post}$ gives

$$\begin{aligned} C_{post} &= \int \vec{x}\vec{x}^T p(\vec{x}|n=1)d\vec{x} \\ &= \int \vec{x}\vec{x}^T \frac{1}{Z} p(\vec{x}) f(K^T\vec{x}) d\vec{x} \\ &= \int \vec{x}\vec{x}^T \frac{1}{Z} \exp\left(\frac{1}{2}\vec{x}^T C^{-1}\vec{x} + \log f(K^T\vec{x})\right) d\vec{x} \\ &= \left(C^{-1} + KBK^T\right)^{-1}, \end{aligned}$$

for some m-by-m matrix B and normalization factor Z. Now, by the Woodbury lemma, we have

$$C - C_{post} = C - \left[C - CK\left(B + K^T CK\right)^{-1} K^T C\right] = CK\left(B + K^T CK\right)^{-1} K^T C.$$

This matrix has rank at most m, with column space CK. Therefore a consistent estimator of K in this case may be constructed by computing the significant eigenvectors $eig(C - \hat{C}_{post})$ (with $eig(.)$ as defined above), and then computing

$$\hat{K} = C^{-1} eig(C - C_{post}),$$

to correct for the C term above.

The Gaussian condition on $p(\vec{x})$ is not only sufficient for consistency of the STC estimator, but also necessary (Paninski, 2003), in the sense (as in the STA case discussed above) that if $p(\vec{x})$ is non-Gaussian, then there exists an $f(.)$ for which the STC estimator has a nonnegligible bias (i.e., is inconsistent) for the column space of K. Nonetheless, for any given $f(.)$ the estimator may often have a suitably small bias, and the STC technique has been used profitably in the context of highly non-Gaussian $p(\vec{x})$ (e.g., natural visual scene stimuli (Touryan et al., 2002)).

Finally, it is worth noting that estimating the matrix C_{post} requires a fair amount of data. It is known that the usual empirical estimate of C_{post} (the sample second-order matrix) is unbiased, but if $d = \dim(\vec{x})$ is on the same order as N, the number of spikes, then the corresponding empirical eigenvalue spectrum is in fact strongly biased (Fig. 8). Dealing with this bias is an active research area in random matrix theory (Johnstone, 2000; Ledoit and Wolf, 2004; Schafer and Strimmer, 2005; El Karoui, 2007), though to date few of these more recent methods have been applied in the context of neural data.

0.9 Fully semiparametric estimators give correct estimates more generally than do the STA or STC estimators

It is natural to seek an estimator for K which is guaranteed to be consistent more generally (i.e., without the restrictive Gaussian assumptions of the STC technique, or the somewhat less restrictive elliptical symmetry assumption but more restrictive one-dimensional $\vec{k}$ assumption of the STA technique). Several such estimators have been constructed (Weisberg and Welsh, 1994; Paninski, 2003; Sharpee et al., 2004); however, the gains in generality are offset by the fact that the resulting estimators are much less computationally tractable than the STA or STC techniques, which require only simple linear algebraic operations.

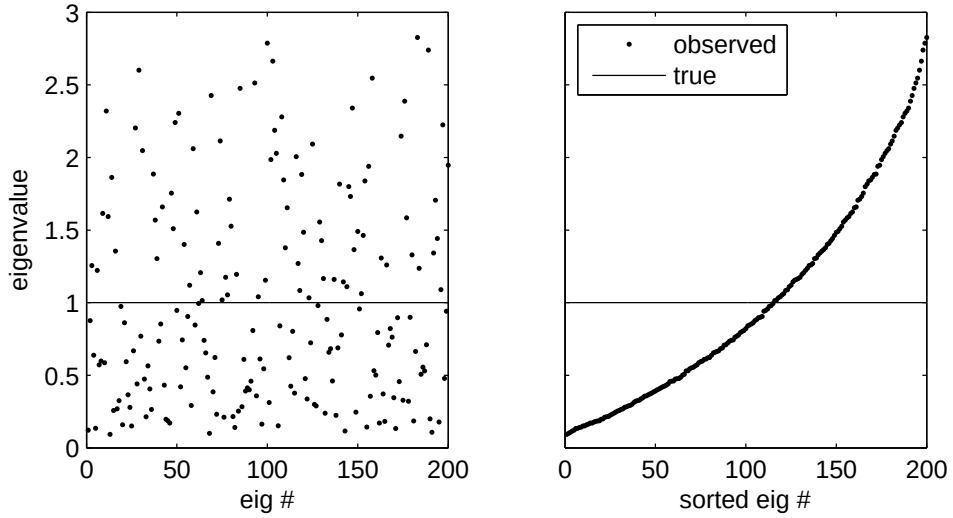

Figure 8: Illustration of the bias in a sparsely-sampled covariance spectrum (Johnstone, 2000). 400 samples were drawn from a 200-dimensional standard normal distribution (mean zero, identity covariance), and then the eigenvalues of the empirical covariance matrix were computed. These eigenvalues display a good deal of variability around their true value (identically 1 in this case), and this variability is converted into systematic bias when the eigenvalues are sorted; unsorted eigenvalues shown on the **left**, and sorted eigenvalues shown on the **right**.

Two approaches have been proposed. The first (Weisberg and Welsh, 1994) is to use maximum likelihood to simultaneously fit both K and the function $f(.)$, where $f(.)$ is represented in a finite dimensional basis which is allowed to become richer with increasing sample size N. Standard results for the consistency of maximum likelihood estimators may be adapted to establish the consistency of the resulting sequence of estimators for K; however, computationally solving this likelihood optimization problem is difficult (and becomes more difficult as N increases, as the dimensionality of the space in which we must search for an optimal $f(.)$ increases).

A second approach is to contruct an objective function, $M(V)$ (here V denotes a matrix of the same size as K), with the property that $M(V)$ obtains its optimum if and only if $V = K$. Then we construct our estimator by maximizing some empirical estimator $\hat{M}(V)$. The idea is that if $\hat{M}(V)$ is a good estimator for $M(V)$, then given enough data the maximizer of $\hat{M}(V)$ should be close to the maximizer of $M(V)$, i.e., K. See, e.g., (van der Vaart, 1998) for more details on this argument.

The objective function chosen is based on the so-called "data processing inequality" from information theory (Cover and Thomas, 1991). The basic idea is that $V^T\vec{x}$ is equivalent to $K^T\vec{x}$ plus some "noise" term that does not affect the spike process (more precisely, this noise term is conditionally independent of n given $K^T\vec{x}$); this noise term is obviously 0 for $V = K$. Thus if we optimize the objective function

$$M(V) \equiv I(V^T\vec{x}; n),$$

with $I(X;Y)$ denoting the mutual information between the random variables X and Y,

$$I(X;Y) = \int p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy,$$

the data processing inequality (Cover and Thomas, 1991) says that $M(V)$ obtains its maximum at the point $V = K$. Another way to put it is that $M(V)$ measures how strongly $V^T\vec{x}$ modulates the firing rate of the cell: for V near K, the conditional measures $p(n = i|V^T\vec{x})$ are on average very different from the prior measure $p(n = i)$, and $M(V)$ is designed to detect exactly these differences; conversely, for V orthogonal to K, the conditional distributions $p(n = i|V^T\vec{x})$ will appear relatively "unmodulated" (that is, $p(n = i|V^T\vec{x})$ will tend to be much nearer the average $p(n = i)$), and $M(V)$ will be comparatively small.

Under weak conditions (e.g., that the support of $p(\vec{x})$ covers every possible point $\vec{x}$ — i.e., all points are sampled with some probability, roughly speaking), this maximum of $M(V)$ at $V = K$ is unique (up to equivalence of column spaces), by the converse to the data processing inequality (Cover and Thomas, 1991). Thus if we choose some uniformly consistent estimator $\hat{M}(V)$ of $M(V)$ (Beirlant et al., 1997), it is easy to show that the estimator

$$\hat{K} = \arg\max_V \hat{M}(V)$$

is consistent for K under weak conditions (Paninski, 2003; Sharpee et al., 2004). However, as emphasized above, maximizing $\hat{M}(V)$ is often fairly computationally difficult, since $\hat{M}(V)$ is typically non-convex and many local maxima may occur in general.

0.10 Spike history effects introduce bias in simple classification-based estimators

As emphasized at the beginning of this chapter, the methods we have been discussing so far make sense when each element $n(t)$ of the neural response is treated as a conditionally independent random variable, given $\vec{x}$. Of course, this is at best an approximation, and in many cases this approximation is overly crude (Brillinger, 1992; Berry and Meister, 1998; Paninski et al., 2004; Truccolo et al., 2005): neurons have refractory effects; some neurons have bursty spike trains; firing rates typically adapt in some manner with time. How do such spiking history effects impact the analyses we have been discussing?

One fairly general model of these spike history effects is as follows:

$$p(spike|\vec{x}, s_-) = f(K^T\vec{x})g(T(s_-)). \tag{2}$$

Here T is some statistic of s_-, the spike train up to the present time (e.g., T could encode the time since the last spike (Miller and Mark, 1992; Berry and Meister, 1998; Kass and Ventura, 2001)); the "modulation function" g maps the range of T into $[0, \infty)$.

The key equality exploited in establishing the unbiasedness of the STA estimator in the LN model does not hold in general for model (2):

$$\begin{aligned} p(n = 1|\vec{x}) &= \int p(n = 1|\vec{x}, s_-)p(s_-|\vec{x})ds_- \\ &= \int f(K^T\vec{x})g(T(s_-))p(s_-|\vec{x})ds_- \\ &= f(K^T\vec{x}) \int g(T(s_-))p(s_-|\vec{x})ds_- \\ &= f(K^T\vec{x})h(\vec{x}). \end{aligned}$$

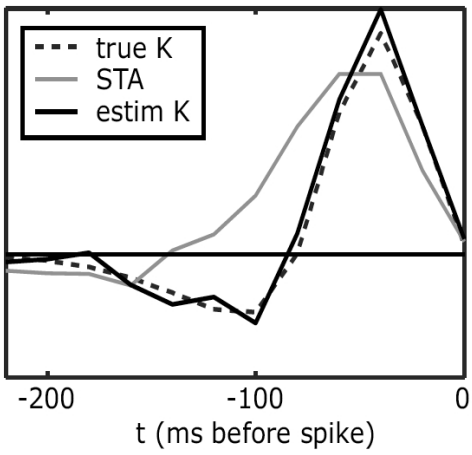

Figure 9: Illustration of the bias in the spike-triggered average when the true data displays spike-history effects. Simulated data were generated by sampling from an integrate-and-fire model driven by an input current formed by convolving a Gaussian white noise signal $x(t)$ with a temporal filter $\vec{k}$ (see (Paninski et al., 2004) for full details). Dashed line shows the true kernel $\vec{k}$ (a 12-sample function chosen to resemble the biphasic temporal impulse response of a macaque retinal ganglion cell (Chichilnisky, 2001)). While it is possible to estimate $\vec{k}$ directly (solid black trace) using likelihood techniques we will discuss later (Paninski et al., 2004), it is clear that the spike-triggered average differs significantly from the true $\vec{k}$ here, despite the radial symmetry of the stimulus $\vec{x}$.

The second equality is (2), and the last is by way of definition: h is an abbreviation for the conditional expectation of $g(T(s_-))$ given $\vec{x}$. If $g \equiv 1$ (as in the basic LN model), then $h(\vec{x}) \equiv 1$, and we recover the original unbiased result. However, in general, h is nonconstant in $\vec{x}$ (and is typically neither elliptically symmetric nor dependent only on the projection $\vec{k}$): h depends on $\vec{x}$ not only through its projection onto $\vec{K}$, but also through its projection on all time-shifted copies of K. Any K with any time-dependence will not, by definition, be time-translation invariant, and in this case the STA (and every other estimator we have discussed here) is biased for $\vec{k}$, even in the case of Gaussian $p(\vec{x})$.

Thus our next goal will be to develop methods that allow us to estimate both the stimulus-dependent effects ($\vec{k}$) and the spike-history effects simultaneously, in an unbiased fashion. First, though, we need to develop some point-process theory.

References

Ahrens, M., Paninski, L., Petersen, R., and Sahani, M. (2006). Input nonlinearity models of barrel cortex responses. *Computational Neuroscience Meeting, Edinburgh*.

Beirlant, J., Dudewicz, E., Gyorfi, L., and van der Meulen, E. (1997). Nonparametric entropy estimation: an overview. *International Journal of the Mathematical Statistics Sciences*, 6:17–39.

Berry, M. and Meister, M. (1998). Refractoriness and neural precision. *J. Neurosci.*, 18:2200–2211.

Bickel, P., Klaassen, C., Ritov, Y., and Wellner, J. (1998). *Efficient and Adaptive Estimation for Semiparametric Models*. Springer.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Oxford University Press.

Brenner, N., Bialek, W., and de Ruyter van Steveninck, R. (2001). Adaptive rescaling optimizes information transmission. *Neuron*, 26:695–702.

Brillinger, D. (1992). Nerve cell spike train data analysis: a progression of technique. *Journal of the American Statistical Association*, 87:260–271.

Chichilnisky, E. (2001). A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12:199–213.

Cover, T. and Thomas, J. (1991). *Elements of information theory*. Wiley, New York.

de Boer, E. and Kuyper, P. (1968). Triggered correlation. *IEEE Transactions on Biomedical Engineering*, 15:159–179.

de Ruyter van Steveninck, R. and Bialek, W. (1988). Real-time performance of a movement-senstivive neuron in the blowfly visual system: coding and information transmission in short spike sequences. *Proc. R. Soc. Lond. B*, 234:379–414.

Devroye, L. and Lugosi, G. (2001). *Combinatorial Methods in Density Estimation*. Springer-Verlag, New York.

Duda, R. and Hart, P. (1972). *Pattern classification and scene analysis*. Wiley, New York.

El Karoui, N. (2007). Spectrum estimation for large dimensional covariance matrices using random matrix theory. *arXiv:math/0609418v1*.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer.

Hunter, I. and Korenberg, M. (1986). The identification of nonlinear biological systems: Wiener and hammerstein cascade models. *Biological Cybernetics*, 55:135–144.

Huys, Q., Ahrens, M., and Paninski, L. (2006). Efficient estimation of detailed single-neuron models. *Journal of Neurophysiology*, 96:872–890.

Johnstone, I. (2000). On the distribution of the largest principal component. Technical Report 2000-27, Stanford.

Kass, R. and Ventura, V. (2001). A spike-train probability model. *Neural Comp.*, 13:1713–1720.

Kutner, M., Nachtsheim, C., Neter, J., and Li, W. (2005). *Applied Linear Statistical Models*. McGraw-Hill.

Ledoit, O. and Wolf, M. (2004). Honey, i shrunk the sample covariance matrix. *Journal of Portfolio Management*, 30:110–119.

Lewi, J., Butera, R., and Paninski, L. (2006). Real-time adaptive information-theoretic optimization of neurophysiological experiments. *NIPS*.

Machens, C., Wehr, M., and Zador, A. (2003). Spectro-temporal receptive fields of subthreshold responses in auditory cortex. *NIPS*.

Marmarelis, P. and Marmarelis, V. (1978). *Analysis of physiological systems: the white-noise approach*. Plenum Press, New York.

Metzner, W., Koch, C., Wessel, R., and Gabbiani, F. (1998). Feature extraction by burst-like spike patterns in multiple sensory maps. *Journal of Neuroscience*, 18:2283–2300.

Miller, M. and Mark, K. (1992). A statistical study of cochlear nerve discharge patterns in response to complex speech stimuli. *Journal of the Acoustical Society of America*, 92:202–209.

Nikitchenko, M. and Paninski, L. (2007). An expectation-maximization Fokker-Planck algorithm for the noisy integrate-and-fire model. *COSYNE*.

Paninski, L. (2003). Convergence properties of some spike-triggered analysis techniques. *Network: Computation in Neural Systems*, 14:437–464.

Paninski, L. (2004). Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15:243–262.

Paninski, L. (2006). The most likely voltage path and large deviations approximations for integrate-and-fire neurons. *Journal of Computational Neuroscience*, 21:71–87.

Paninski, L., Pillow, J., and Simoncelli, E. (2004). Maximum likelihood estimation of a stochastic integrate-and-fire neural model. *Neural Computation*, 16:2533–2561.

Pillow, J. and Paninski, L. (2007). Model-based decoding, information estimation, and change-point detection in multi-neuron spike trains. *Submitted*.

Rust, N., Mante, V., Simoncelli, E., and Movshon, J. (2006). How MT cells analyze the motion of visual patterns. *Nature Neuroscience*, 11:1421–1431.

Rust, N., Schwartz, O., Movshon, A., and Simoncelli, E. (2005). Spatiotemporal elements of macaque V1 receptive fields. *Neuron*, 46:945–956.

Sahani, M. (2000). Kernel regression for neural systems identification. Presented at NIPS00 workshop on Information and statistical structure in spike trains; abstract available at http://www-users.med.cornell.edu/∼jdvicto/nips2000speakers.html.

Schafer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, 4:32.

Schervish, M. (1995). *Theory of statistics*. Springer-Verlag, New York.

Schnitzer, M. and Meister, M. (2003). Multineuronal firing patterns in the signal from eye to brain. *Neuron*, 37:499–511.

Scholkopf, B. and Smola, A. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press.

Sharpee, T., Rust, N., and Bialek, W. (2004). Analyzing neural responses to natural signals: Maximally informative dimensions. *Neural Computation*, 16:223–250.

Simoncelli, E., Paninski, L., Pillow, J., and Schwartz, O. (2004). Characterization of neural responses with stochastic stimuli. In *The Cognitive Neurosciences*. MIT Press, 3rd edition.

Simoncelli, E. P. and Adelson, E. H. (1996). Noise removal via Bayesian wavelet coring. In *Third Int'l Conf on Image Proc*, volume I, pages 379–382, Lausanne. IEEE Sig Proc Society.

Smyth, D., Willmore, B., Baker, G., Thompson, I., and Tolhurst, D. (2003). The receptive-field organization of simple cells in primary visual cortex of ferrets under natural scene stimulation. *Journal of Neuroscience*, 23:4746–4759.

Touryan, J., Lau, B., and Dan, Y. (2002). Isolation of relevant visual features from random stimuli for cortical complex cells. *Journal of Neuroscience*, 22:10811–10818.

Truccolo, W., Eden, U., Fellows, M., Donoghue, J., and Brown, E. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble and extrinsic covariate effects. *Journal of Neurophysiology*, 93:1074–1089.

van der Vaart, A. (1998). *Asymptotic statistics*. Cambridge University Press, Cambridge.

Weisberg, S. and Welsh, A. (1994). Adapting for the missing link. *Annals of Statistics*, 22:1674–1700.