

A Maximum-Likelihood Interpretation for Slow Feature Analysis

Richard Turner

turner@gatsby.ucl.ac.uk

Maneesh Sahani

maneesh@gatsby.ucl.ac.uk

Gatsby Computational Neuroscience Unit, University College London, London, WC1N 3AR, U.K.

The brain extracts useful features from a maelstrom of sensory information, and a fundamental goal of theoretical neuroscience is to work out how it does so. One proposed feature extraction strategy is motivated by the observation that the meaning of sensory data, such as the identity of a moving visual object, is often more persistent than the activation of any single sensory receptor. This notion is embodied in the slow feature analysis (SFA) algorithm, which uses “slowness” as an heuristic by which to extract semantic information from multi-dimensional time-series. Here, we develop a probabilistic interpretation of this algorithm showing that inference and learning in the limiting case of a suitable probabilistic model yield exactly the results of SFA. Similar equivalences have proved useful in interpreting and extending comparable algorithms such as independent component analysis. For SFA, we use the equivalent probabilistic model as a conceptual spring-board, with which to motivate several novel extensions to the algorithm.

1 Introduction

The meaning of sensory information often varies more slowly than the activity of low-level sensory receptors. For example, the output of photoreceptors directed towards a swaying tree on a bright windy day may flicker; but the identity and percept of the tree remains invariant. Observations of this sort motivate temporal constancy, or “slowness”, as a useful learning principle to extract meaningful higher-level descriptions of sensory data (Hinton, 1989; Földiák, 1991; Mitchison, 1991; Stone, 1996).

The slowness learning principle is at the core of the slow feature analysis (SFA) algorithm (Wiskott & Sejnowski, 2002). SFA linearly extracts slowly-varying, uncorrelated projections of multi-dimensional time-series data, ordered by their slowness. When SFA is trained on a non-linear expansion of a video of natural scene patches, the filter outputs are found to resemble the receptive fields of complex cells (Berkes & Wiskott, 2005). Slowness and decorrelation of features (in SFA, and similar algorithms, for example: Kayser, Körding, & König, 2003; Körding, Kayser, Einhäuser, & König, 2004) thus provides an interesting alternative heuristic to sparseness and independence for recovering

receptive fields similar to those observed in the visual system (for example: Olshausen & Field, 1996; Bell & Sejnowski, 1997).

The purpose of this paper is to provide a probabilistic interpretation for SFA. Such a perspective has been useful for both principal component analysis (PCA; see Tipping & Bishop, 1999) and, in particular, independent component analysis (ICA; see MacKay, 1999; Pearlmutter & Parra, 1997), motivating many new learning algorithms (covariant, variational; see Miskin, 2000), and generalisations to the model (independent factor analysis (Attias, 1999) and Gaussian scale mixture priors (Karklin & Lewicki, 2005), etc.). More generally, probabilistic models have several desirable features. For instance they force the tacit assumptions of algorithms to be made explicit, allowing them to be criticised and improved more easily. Furthermore, a number of general tools have been developed for learning and inference in such models; for example, methods to handle missing data (Dempster, Laird, & Rubin, 1977).

The probabilistic framework is a powerful one. Fortunately it is intuitive too, and heuristics such as sparseness and slowness can naturally be translated into probabilistic priors. Indeed, previous work has illustrated how to combine the two approaches into a common probabilistic model (Hyvärinen, Hurri, & Väyrynen, 2003; Hurri & Hyvärinen, 2003). The advantage of the probabilistic approach is that it softens the heuristics and allows them to trade-off against one another. One of the contributions of this paper is to place SFA into a proper context within this common framework.

In the following, we first introduce SFA and provide a geometrical picture of the algorithm. Next, we develop intuition for a class of models in which maximum-likelihood learning has similar flavour to SFA, before proving exact equivalence under certain conditions. In the final section, we use this maximum-likelihood interpretation to motivate a number of interesting probabilistic extensions to the SFA algorithm.

2 Slow Feature Analysis

Formally, the SFA algorithm can be defined as follows: Given an M -dimensional time-series, $\mathbf{x}_t, t = 1 \dots T$, find M weight vectors $\{\mathbf{w}_m\}_{m=1}^M$, such that the average squared temporal difference of the output signals, $\hat{y}_{m,t} = \mathbf{w}_m^T \mathbf{x}_t$ are minimised:

$$\hat{\mathbf{w}}_m = \arg \min_{\mathbf{w}_m} \langle (\hat{y}_{m,t} - \hat{y}_{m,t-1})^2 \rangle, \quad (1)$$

given the constraint

$$\langle \hat{y}_{m,t} \hat{y}_{n,t} \rangle - \langle \hat{y}_{m,t} \rangle \langle \hat{y}_{n,t} \rangle = \delta_{mn}. \quad (2)$$

Without loss of generality we can assume the time-series has zero mean, hence $\langle \hat{y}_{m,t} \rangle = 0$. The constraint is important in that it is necessary to prevent the trivial solutions

$\hat{y}_{m,t} = 0$, and ensure that the output signals report different aspects of the stimulus: $\hat{y}_{m,t} \neq \hat{y}_{n,t}$. Furthermore, it also imposes an ordering on the solutions. We shall see below that the constraint plays as important a role in shaping the results of SFA as does the minimisation itself.

In the following we use $\Delta\hat{\mathbf{y}}$ to represent the time differences of the features $\hat{\mathbf{y}}_t - \hat{\mathbf{y}}_{t-1}$, and similarly $\Delta\mathbf{x}$ for the data time differences. With SFA defined in this way, it can be shown the optimal weights satisfy the generalised eigenvalue equation

$$\mathbf{W}\mathbf{A} = \bar{\mathbf{\Omega}}^2\mathbf{W}\mathbf{B}. \quad (3)$$

Where we have defined: $\hat{\mathbf{y}}_t = \mathbf{W}\mathbf{x}_t$ (that is, the matrix \mathbf{W} collects the weight vectors \mathbf{w}_m), $\mathbf{A}_{mn} = \langle \Delta x_m \Delta x_n \rangle$, $\mathbf{B}_{mn} = \langle x_m x_n \rangle$, and $\bar{\mathbf{\Omega}}_{mn}^2 = \bar{\omega}_m^2 \delta_{mn}$

The slow features are ordered from slowest to fastest by the eigenvalues $\bar{\omega}_m^2$. As there are efficient methods for solving the generalised eigenvalue equation, learning as well as inference is very fast.

3 A Geometric Perspective

SFA has a useful geometric interpretation, that makes clear connections to other algorithms such as principal component analysis. The constraint of eq. 2 requires that multiplication by the SFA weight matrix spatially whiten or “sphere” the data. In general, such a sphering matrix is given by the product of the square root of the covariance matrix with an arbitrary orthogonal transform:

$$\mathbf{W} = \mathbf{R}\mathbf{B}^{-1/2}. \quad (4)$$

Geometrically, as illustrated in Fig. 1, the rows of \mathbf{W} are constrained to lie on a hyper-ellipsoid, but we are free to choose their rotation \mathbf{R} (Särelä & Valpola, 2005). PCA makes one particular choice, $\mathbf{R} = \mathbf{I}$, and thus the rows of the principal loading matrix (analogous to \mathbf{W}) are parallel to the eigenvectors of \mathbf{B} . They may be ordered by the eigenvalues of \mathbf{B} , which give the variances of the data in each principal direction. Various ICA algorithms choose a non-trivial rotation \mathbf{R} , so as to maximise kurtosis or some other statistic of the projected data. In this case, a natural ordering is given by the value of the projected statistic.

[Figure 1 about here.]

By contrast, SFA chooses a rotation based on dynamical information, using the normalised eigenvectors of the covariance matrix of the temporal differences of the sphered data¹. This can be seen by substituting eq. 4 into eq. 3 leading to: $\mathbf{R}\mathbf{B}^{-1/2}\mathbf{A}\mathbf{B}^{-1/2} =$

¹However, some variants of ICA are closely related, see Blaschke, Berkes, & Wiskott, 2006

$\bar{\Omega}^2\mathbf{R}$. Geometrically, SFA corresponds to transforming into the sphered (PCA) space, and then rotating again to align the axes of the transformed data with the principal axes of the transformed time differences.

The natural ordering is then given by the eigenvalues of the temporal-difference covariance matrix. If each projection \hat{y}_t varies sinusoidally in time, the eigenvalues are the squares of the corresponding frequencies. More generally, the eigenvalues give the average squared frequency of the extracted features, where the average is taken over the normalised spectral power density. That is,

$$\bar{\omega}_n^2 = \sum_{\omega} \omega^2 \frac{|\tilde{y}_n(\omega)|^2}{\sum_{\omega'} |\tilde{y}_n(\omega')|^2}, \quad (5)$$

where $\tilde{y}_n(\omega)$ is the ω -component of the discrete Fourier transform of the n^{th} feature².

The geometric perspective makes clear one important difference between PCA and SFA: PCA is dependent on the relative scales of the measurements, but SFA is not. That is to say, rescaling a subset of dimensions of the observations $x_{n,1:T}$ alters the eigenvectors of the covariance matrix, and consequently the extracted time-series $\mathbf{y}_{1:T}$ that PCA recovers is also changed. Critically, SFA finds an interesting rotation *after* sphering the data and so the extracted time-series $\mathbf{y}_{1:T}$ that it recovers is insensitive to the rescaling. For this reason SFA can be used meaningfully with an expanded set of pseudo-observations (whose scale is arbitrary), whilst PCA cannot. This is a property that SFA shares with other algorithms such as factor analysis (Roweis & Ghahramani, 1999).

4 Towards a probabilistic interpretation

The very notion of extracting features based on temporal slowness is captured naturally in the generative modelling framework, and this suggests that SFA might be amenable to such an interpretation. To show the correspondence between the SFA algorithm and maximum likelihood learning of a particular probabilistic latent variable model (perhaps in a particular limit), we interpret SFA's weights as recognition weights derived from the statistical inverse of a generative model. The output signals then have a natural interpretation as the mean or mode of the posterior distribution over latent variables, given the data.

Our goal is to unpack the assumptions implicit in SFA's cost function and constraints, and intuitively build corresponding factors into a simple candidate form of generative model. This will then be formally analysed to re-derive the SFA algorithm under certain conditions.

²To prove this result note that eq. 3 gives $\mathbf{w}_n^T \mathbf{A} \mathbf{w}_n^T = \langle \Delta \hat{y}_n^2 \rangle = \bar{\omega}_n^2 \langle \hat{y}_n^2 \rangle$. By Parseval's theorem $\sum_t |\hat{y}_{n,t}|^2 = \sum_{\omega} |\tilde{y}_n(\omega)|^2$ and (using the Fourier transform of the derivative operator) $\sum_t |\Delta \hat{y}_{n,t}|^2 = \sum_{\omega} |\omega \tilde{y}_n(\omega)|^2$. Substituting these in for the two averages gives eq. 5.

One benefit of this new perspective relates to the hard constraints that had to be introduced to prevent SFA recovering trivial solutions. In a probabilistic model softened versions of these constraints might be expected to arise naturally. For example, in SFA the scale of the output signals must be set by hand to prevent them being shrunk away by the smoothness objective function. The temporal difference cost function depends only on the recovered signals, and there is no penalty for eliminating information about the measured data. In a probabilistic setting, a soft volume penalty arising from a combination of the normalising constant of the prior and the information-preserving likelihood term will prevent shrinking in an automatic manner. However, one of the consequences of such a soft constraint is that the scale of each of the latents may not be identical. Fortunately, this ambiguity does not effect the ordering of the solutions.

Similarly, the other constraint of SFA — that the output signals be decorrelated — emerges by choosing a decorrelated prior. Indeed, as with probabilistic PCA and ICA, we choose a fully factored prior distribution on the latent sources. Our ultimate choice will be Gaussian, so that decorrelation and independence are the same at each time-step; although the factored form also implies that the processes are decorrelated at different time steps.

Thus, consideration of the SFA constraints suggests a factored prior over the latent variables. The cost function, which penalises the sum of squared differences between adjacent time-steps, further specifies the form of the prior on each latent time-series. The squared cost is consistent with an underlying Gaussian distribution, while the penalty on adjacent differences suggests a Markov chain structure. A reasonable candidate is thus a one time-step linear-Gaussian dynamical system (or AR(1) auto-regressive model).

$$\begin{aligned}
 p(\mathbf{y}_t | \mathbf{y}_{t-1}, \lambda_{1:N}, \sigma_{1:N}^2) &= \prod_{n=1}^N p(y_{n,t} | y_{n,t-1}, \lambda_n, \sigma_n^2) \\
 p(y_{n,t} | y_{n,t-1}, \lambda_n, \sigma_n^2) &= \text{Norm}(\lambda_n y_{n,t-1}, \sigma_n^2) \\
 p(y_{n,1} | \sigma_{n,1}^2) &= \text{Norm}(0, \sigma_{n,1}^2).
 \end{aligned} \tag{6}$$

Intuitively the λ_n control the strength of the correlations between the latent variables at different time points, and therefore their slowness. If $\lambda_n = 0$ then successive variables are uncorrelated and the processes vary rapidly. As $\lambda_n \rightarrow 1$ the processes become progressively more correlated and hence slower. More generally, if $|\lambda_n| < 1$ then after a long time ($t \rightarrow \infty$), the prior distribution of each latent series settles into a stationary state with temporal covariance, or autocorrelation, given by

$$\langle y_{n,t} y_{n,t-\tau} \rangle \rightarrow \frac{\sigma_n^2}{1 - \lambda_n^2} \lambda_n^{|\tau|}. \tag{7}$$

The “effective memory” of this prior (defined as the time difference over which correlations fall by $1/e$) is thus $\tau_{\text{eff}} = -1/\log \lambda_n$.

At first glance, the choice of $|\lambda_n| < 1$ would seem to suggest that each latent process shrinks towards 0 with time; but this clearly is at odds with the stationary distribution derived above. This apparent conflict is resolved by noting that while the mean

of the one-step conditional distribution is indeed smaller than the preceding value, $\langle y_{n,t}|y_{n,t-1} \rangle = \lambda_n y_{n,t-1}$, the second moment includes the effects of the innovations process $\langle y_{n,t}^2|y_{n,t-1} \rangle = \lambda_n^2 y_{n,t-1}^2 + \sigma_n^2$. Thus, if $y_{n,t-1}^2 < \frac{\sigma_n^2}{1-\lambda_n^2}$, the conditional expected square is *larger* than $y_{n,t-1}^2$. Indeed, it is clear that the only way to achieve a stationary AR(1) process with non-zero innovations noise is to choose $|\lambda_n| < 1$.

Two useful corollaries follow from eq. 7. First, it is possible to express the prior expectation that each latent process is stationary from the outset (that is, without a initial transient) by choosing its initial distribution to match the long-run variance:

$$\sigma_{n,1}^2 = \frac{\sigma_n^2}{1 - \lambda_n^2}. \quad (8)$$

This form will be assumed in the following, but it is not essential to the derivation. Second, by choosing $\sigma_n^2 = 1 - \lambda_n^2$, we can set the stationary variance of the prior to one, a fact which we will make use of later.

It is important to note here that we have so far discussed the properties of the prior $p(\mathbf{y}_{1:T})$, and that these may generally be quite different from the properties of the posterior $p(\mathbf{y}_{1:T}|\mathbf{x}_{1:T})$.

In addition to the prior on the latent variables, the complete specification of a generative model requires a (potentially probabilistic) mapping from the latents to the observations. This is constrained by noting that inference in SFA is instantaneous: a feature at time t is derived through a linear combination of only the current observations, without reference to earlier or later observed data. In the general Markov dynamical model, accurate inference requires information from multiple time-points. For example, adding a linear-Gaussian output mapping to the latent model (eq. 6), leads to inference through the well-known Kalman smoothing recursions in time. The only way for instantaneous linear recognition to be optimal is for the mapping from latents to observations to be deterministic and linear; the matrix of generative weights is then the inverse of the recognition matrix \mathbf{W} . It is useful to view this deterministic process as the limit of a probabilistic mapping (Tipping & Bishop, 1999), and a natural choice is:

$$p(\mathbf{x}_t|\mathbf{y}_t, \mathbf{W}, \sigma_x) = \text{Norm}(\mathbf{W}^{-1}\mathbf{y}_t, \sigma_x^2\mathbf{I}), \quad (9)$$

where the deterministic mapping is recovered as $\sigma_x^2 \rightarrow 0$.

This completes the specification of the model which is a member of the linear Gaussian state space models (Roweis & Ghahramani, 1999), in the limit of zero observation noise. We have described why it might have the same flavour as SFA, but it is certainly not clear *a priori* under what restrictions this equivalence will hold. For instance, we might worry that peculiar settings of the transition dynamics and state noise might be required. Fortunately these conditions can be derived analytically and are found to be surprisingly un-restrictive. Denoting the parameters $\theta = [\sigma_{1:N}^2, \lambda_{1:N}, \sigma_x^2, \mathbf{W}]$ we first form the likelihood function:

$$\begin{aligned}
p(\mathbf{x}_{1:T}|\theta) &= \int d\mathbf{y}_{1:T} \left[\prod_{t=1}^T p(\mathbf{x}_t|\mathbf{y}_t, \mathbf{W}, \sigma_x^2) \right] \\
&\times \left[\prod_{n=1}^N p(y_{n,1}|\sigma_n^2) \prod_{t=2}^T p(y_{n,t}|y_{n,t-1}, \lambda_n, \sigma_n^2) \right] \quad (10)
\end{aligned}$$

$$\begin{aligned}
&= \int d\mathbf{y}_{1:T} \left[\prod_{t=1}^T \delta(\mathbf{x}_t - \mathbf{W}^{-1}\mathbf{y}_t) \right] \\
&\times \frac{1}{Z} \exp \left[- \sum_{n=1}^N \left(\frac{1}{2\sigma_{n,1}^2} y_{n,1}^2 + \frac{1}{2\sigma_n^2} \sum_{t=2}^T [y_{n,t} - \lambda_n y_{n,t-1}]^2 \right) \right], \quad (11)
\end{aligned}$$

where we have taken the limit $\sigma_x^2 \rightarrow 0$ in the likelihood term.

Completing the integrals, the log-likelihood is

$$L(\theta) = \log p(\mathbf{x}_{1:T}|\theta) \quad (12)$$

$$\begin{aligned}
&= c + T \log |\det \mathbf{W}| - \sum_{n=1}^N \left[\frac{1}{2\sigma_{n,1}^2} (\mathbf{w}_n^T \mathbf{x}_1)^2 \right. \\
&\quad \left. + \frac{1}{2\sigma_n^2} \sum_{t=2}^T (\mathbf{w}_n^T \mathbf{x}_t - \lambda_n \mathbf{w}_n^T \mathbf{x}_{t-1})^2 \right]. \quad (13)
\end{aligned}$$

Where the constant c is independent of the weights. Expanding the square yields

$$\begin{aligned}
L(\theta) &= c + T \log |\det \mathbf{W}| - \sum_{n=1}^N \frac{1}{2\sigma_n^2} \left[\frac{\sigma_n^2}{\sigma_{n,1}^2} \mathbf{w}_n^T \mathbf{x}_1 \mathbf{x}_1^T \mathbf{w}_n + \mathbf{w}_n^T \left(\sum_{t=2}^T \mathbf{x}_t \mathbf{x}_t^T \right) \mathbf{w}_n \right. \\
&\quad \left. + \lambda_n^2 \mathbf{w}_n^T \left(\sum_{t=1}^{T-1} \mathbf{x}_t \mathbf{x}_t^T \right) \mathbf{w}_n - \lambda_n \mathbf{w}_n^T \left(\sum_{t=2}^T \mathbf{x}_t \mathbf{x}_{t-1}^T + \sum_{t=2}^T \mathbf{x}_{t-1} \mathbf{x}_t^T \right) \mathbf{w}_n \right]. \quad (14)
\end{aligned}$$

The following identities are then useful:

$$\sum_{t=2}^T \mathbf{x}_t \mathbf{x}_t^T = T\mathbf{B} - \mathbf{x}_1 \mathbf{x}_1^T \quad (15)$$

$$\sum_{t=1}^{T-1} \mathbf{x}_t \mathbf{x}_t^T = T\mathbf{B} - \mathbf{x}_T \mathbf{x}_T^T \quad (16)$$

$$\sum_{t=2}^T \mathbf{x}_t \mathbf{x}_{t-1}^T + \sum_{t=2}^T \mathbf{x}_{t-1} \mathbf{x}_t^T = 2T\mathbf{B} - (T-1)\mathbf{A} - \mathbf{x}_1 \mathbf{x}_1^T - \mathbf{x}_T \mathbf{x}_T^T \quad (17)$$

Where we remind the reader

$$\mathbf{A}_{mn} = \langle \Delta x_m \Delta x_n \rangle = \frac{1}{T-1} \sum_{t=1}^{T-1} (x_{m,t+1} - x_{m,t})(x_{n,t+1} - x_{n,t})$$

$$\mathbf{B}_{mn} = \langle x_m x_n \rangle = \frac{1}{T} \sum_{t=1}^T x_{m,t} x_{n,t}.$$

Substituting and collecting terms this yields

$$L(\theta) = c + T \log |\det \mathbf{W}| - \sum_{n=1}^N \frac{T}{2\sigma_n^2} \left(\mathbf{w}_n^T \left[\mathbf{B}(1 - \lambda_n)^2 + \frac{(T-1)}{T} \mathbf{A} \lambda_n + \frac{\lambda_n(1 - \lambda_n)}{T} (\mathbf{x}_1 \mathbf{x}_1^T + \mathbf{x}_T \mathbf{x}_T^T) \right] \mathbf{w}_n \right). \quad (18)$$

As the number of observations increases, $T \rightarrow \infty$, the relative contribution from edge effects reduces, $\frac{\lambda_n(1-\lambda_n)}{T} (\mathbf{x}_1 \mathbf{x}_1^T + \mathbf{x}_T \mathbf{x}_T^T) \rightarrow 0$ and $\frac{T-1}{T} \rightarrow 1$. Therefore, assuming a large number of data points:

$$L(\theta) \approx c + T \log |\det \mathbf{W}| - \sum_n \frac{T}{2\sigma_n^2} \mathbf{w}_n^T [\mathbf{B}(1 - \lambda_n)^2 + \mathbf{A} \lambda_n] \mathbf{w}_n \quad (19)$$

$$= c + T \log |\det \mathbf{W}| - \frac{T}{2} \text{tr} \left[\mathbf{W} \mathbf{B} \mathbf{W}^T \mathbf{\Lambda}^{(2)} + \mathbf{W} \mathbf{A} \mathbf{W}^T \mathbf{\Lambda}^{(1)} \right], \quad (20)$$

where $\mathbf{\Lambda}_{mn}^{(1)} = \delta_{mn} \frac{\lambda_n}{\sigma_n^2}$, and $\mathbf{\Lambda}_{mn}^{(2)} = \delta_{mn} \frac{(1-\lambda_n)^2}{\sigma_n^2}$. Differentiating this expression with respect to the recognition weights (assuming the determinant is not zero), recovers the following condition:

$$\frac{dL(\theta)}{d\mathbf{W}} \propto \mathbf{W}^{-T} - \left[\mathbf{\Lambda}^{(2)} \mathbf{W} \mathbf{B} + \mathbf{\Lambda}^{(1)} \mathbf{W} \mathbf{A} \right]. \quad (21)$$

Setting this to 0 and rearranging, we obtain a condition on the maximum:

$$\langle \hat{y}_{m,t} \hat{y}_{n,t} \rangle = \frac{\sigma_n^2}{(1 - \lambda_n)^2} \delta_{mn} - \frac{\lambda_n}{(1 - \lambda_n)^2} \langle \Delta \hat{y}_{m,t} \Delta \hat{y}_{n,t} \rangle, \quad (22)$$

where, as defined earlier; $\hat{y}_{m,t} = \mathbf{w}_m^T \mathbf{x}_t$.

Now, as the first term on the right hand side of eq. 22 is diagonal, symmetry of the left hand side requires that, for off-diagonal terms,

$$\frac{\lambda_n}{(1 - \lambda_n)^2} \langle \Delta \hat{y}_{m,t} \Delta \hat{y}_{n,t} \rangle = \frac{\lambda_m}{(1 - \lambda_m)^2} \langle \Delta \hat{y}_{n,t} \Delta \hat{y}_{m,t} \rangle. \quad (23)$$

But, if we choose $\lambda_m \neq \lambda_n$ for $m \neq n$, this condition can only be met when the covariance matrix of the latent temporal differences is diagonal, which in turn makes the covariance of the latents themselves diagonal by eq. 22. This implies that we have recovered the SFA directions, but not necessarily the correct scaling. In other words, the most probable output signals are decorrelated but not of equal power; further rescaling is necessary to achieve sphering. This is a consequence of the soft volume constraints, and might be a desirable result for reasons discussed earlier.

Surprisingly, the maximum-likelihood weights do not depend on the exact setting of the λ_n , so long as they are all different. If $0 < \lambda_n < 1 \quad \forall n$, then larger values of λ_n correspond to slower latents. This corresponds directly to the ordering of the solutions from SFA.

To recover exact equivalence to SFA, another limit is required that corrects the scales. There are several choices, but a natural one is to let $\sigma_n^2 = 1 - \lambda_n^2$ (which fixes the prior covariance of the latent chains to be one, as discussed earlier) and then take the limit $\lambda_1 \leq \lambda_2 \leq \dots \lambda_M \rightarrow 0$, which implies

$$\langle \hat{y}_{m,t} \hat{y}_{n,t} \rangle \rightarrow (1 + 2\lambda_n) \delta_{mn} - \lambda_n \langle \Delta \hat{y}_{m,t} \Delta \hat{y}_{n,t} \rangle \rightarrow \delta_{mn}. \quad (24)$$

Using the geometric intuitions of section 3, at the limit the weights sphere the inputs and therefore lie somewhere on the shell of the hyper-ellipsoid defined by the data covariance. However, as the limit is taken the weights must be parallel to those of SFA (as we have argued above). Provided the behaviour is smooth, as the perturbation vanishes the weights must drop onto the hyper-ellipsoid at precisely the point specified by SFA.

The analytical results presented here have been verified using computer simulations, for instance, using the expectation-maximisation (EM; Dempster et al., 1977) algorithm and annealing the output noise and transition matrices appropriately so as to take the approximate limit (see Fig. 2, top). However, learning by EM is very slow when the variance of output noise σ_x^2 is small; this is alleviated somewhat by the annealing process, but other likelihood optimisation schemes might be more suitable.

[Figure 2 about here.]

In the above analysis we thought of the inputs $\mathbf{x}_{1:T}$ as observations from some temporal process. However, the argument we have presented is agnostic to the actual source of the inputs. This means, for example, that the recognition model and learning as presented above is formally equivalent to SFA even when the inputs $\mathbf{x}_{1:T}$ themselves are formed from non-linear combinations of observations (as is the usually the case in applications of the SFA algorithm). However, the generative model is not likely to correspond to a particularly sensible selection of assumptions in this case, as will be discussed in the next section.

5 Extensions

The probabilistic modelling perspective suggests we could use a linear Gaussian state space model with a factorial prior in place of SFA, and learn the state and observation noise, *i.e.* the dynamics, as well as the weights (for which SFA would provide an intelligent initialisation). This probabilistic version improves over standard SFA when there is substantial observation noise and if there is missing data, even providing a natural measure of confidence under these conditions (see Fig. 2). More radical variants are also interesting.

5.1 Bilinear extensions

Loosely speaking, there are at least two broad types of slow features in images: object identity (what, or content information) and object location and pose (where, or style information), and there is evidence that there are parallel pathways for the processing of each sort in the brain. Bilinear learning algorithms have recently been developed in which the effects of the two are segregated (Tenenbaum & Freeman, 2000; Grimes & Rao, 2005). SFA, by contrast, confounds these two types of signal because it extracts them both into one type of latent variable. A natural extension to the SFA model might therefore augment the continuous “where” latent variables ($y_{n,t}$) with a new set of binary “what” latent variables ($s_{n,t}$), forming independent discrete Markov chains with transition matrices $\mathbf{T}^{(n)}$, and combining the two bilinearly:

$$p(\mathbf{s}_t | \mathbf{s}_{t-1}, \{\mathbf{T}^{(n)}\}_{n=1}^N) = \prod_{n=1}^N p(s_{n,t} | s_{n,t-1}, \mathbf{T}^{(n)}) \quad (25)$$

$$p(s_{n,t} = a | s_{n,t-1} = b, \mathbf{T}^{(n)}) = \mathbf{T}_{ab}^{(n)} \quad (26)$$

$$p(\mathbf{x}_t | \mathbf{y}_t, \mathbf{s}_t, \{\mathbf{g}_{mn}\}_{n=1, m=1}^{N, M}, \sigma_x) = \text{Norm} \left(\sum_{mn} \mathbf{g}_{mn} y_{n,t} s_{m,t}, \sigma_x^2 \mathbf{I} \right). \quad (27)$$

Exact inference in such a model would be intractable, but the maximum likelihood parameters could be approximated, for example by using variational EM (Neal & Hinton, 1998; Jordan, Ghahramani, Jaakkola, & Saul, 1999).

5.2 Generalised bilinear extensions

Traditionally, SFA has been made into a non-linear algorithm by expanding the observations through a large set of non-linearities and using these as new inputs. Probabilistic models are not well suited to such an approach for two different reasons: (1) They are much slower to train when the dimensionality of the inputs is large, and (2) The expansion introduces complex dependencies between the new observations that a standard probabilistic model cannot capture. We consider the second of these points in more detail next, first noting that an interesting alternative which side-steps these two issues

is to learn a non-linear mapping from the latents to the observations directly. One straightforward approach would be to use a generalised linear (or “neural-network”) mapping

$$p(\mathbf{x}_t | \mathbf{y}_t, \mathbf{s}_t, \{\mathbf{g}_{mn}\}_{n=1, m=1}^{N, M}, \sigma_x) = \text{Norm} \left(\Phi \left[\sum_{mn} \mathbf{g}_{mn} y_{n,t} s_{m,t} \right], \sigma_x^2 \mathbf{I} \right). \quad (28)$$

for a nonlinear link function Φ . This form would also easily accommodate types of observation for which a normal output model would be inappropriate; binary data, for instance. Again, approximations would be required for learning.

5.3 Nonlinear extensions

The above is a fairly conventional extension to time-series models (e.g. Ghahramani & Roweis, 1999). Further consideration of the second point above motivates an alternative that is more in the spirit of traditional SFA. In the expanded observation space, points corresponding to realisable data lie on low dimensional manifolds (see Fig. 3 for an example). Therefore, a good generative model should only assign probability to these manifolds and not “waste” it on the unrealisable volumes. The probabilistic model developed thus far is free to generate points anywhere in the expanded observation space; it cannot capture the structure introduced by the expansion. Fortunately, taking inspiration from product of expert models (Hinton, 2002; Osindero, Welling, & Hinton, 2006), we can use our old probabilistic model to produce a new, more sensible model. The idea is to treat the old generative distribution as a global expert that models temporal correlations in the data, and then form a new distribution by multiplying its density by K local experts, which constrain the expanded observations to lie on the realisable manifold, and renormalising. It is important to note that, unlike in the usual product of experts model, the local experts will not correspond to normalisable distributions in their own right.

[Figure 3 about here.]

Generally speaking, product models can be succinctly parameterised via an energy such that $P(w) = \frac{1}{Z} \exp(-E(w))$. A sensible energy function can be devised by introducing two new types of latent variables: $\tilde{x}_{m,t}$, which can usefully be thought of as perturbations from the observations, and $\phi_{k,t}$ which are perturbations in the expanded observation space:

$$E = -\frac{1}{2} \sum_t \left[\frac{1}{\sigma_x^2} \sum_m (x_{m,t} - \tilde{x}_{m,t})^2 + \beta \sum_k [\phi_{k,t} - \phi_k(\tilde{\mathbf{x}}_t)]^2 \right] - \log P(\mathbf{Y}, \Phi). \quad (29)$$

The first two terms in this energy correspond to the local experts, and the final term is the global, dynamical model, expert (alternatively, the old generative model). Intuitively speaking, the global expert assigns lower energy to solutions with slow latent variables Y , and therefore favours them. The local experts are more complicated, but essentially they constrain the observations of the old generative model, Φ , to lie on the realisable manifold as described below (see also Fig. 3).

The $\tilde{x}_{m,t}$ tend to be close to the observations $x_{m,t}$ (as this lowers the energy, by an amount depending on the value of σ_x^2) and can therefore be thought of as perturbations. The $\phi_k(\tilde{\mathbf{x}}_t)$ correspond to the K non-linearities applied in the expansion (e.g. polynomials). Due to the perturbations, they are constrained to lie on a localised region of the manifold centred on $\phi_k(\mathbf{x}_t)$ in the expanded observation space. The latents $\phi_{k,t}$ tend to be close to the $\phi_k(\tilde{\mathbf{x}}_t)$ and therefore lie around the manifold (with a fuzziness determined by β , different from the output noise of the dynamical model). Taken together, these terms act as a local expert, vetoing any output from the global expert that does not lie near to the realisable manifold. The parameters of this model can be learned using contrastive divergence (Hinton, 2002). What is more, the function $\phi_k(\tilde{\mathbf{x}}_t)$ can be parameterised, by a neural network for example, and the parameters can also be learned in the same way.

6 Conclusion

We have shown the equivalence between the SFA algorithm and maximum-likelihood learning in a linear Gaussian state-space model, with an independent Markovian prior. This perspective inspired a number of variants on the traditional slow feature analysis algorithm. First of all, a fully probabilistic version of slow features analysis was shown to be capable of dealing with both output noise and missing data naturally, using the Bayesian framework. Secondly we suggested more speculative probabilistic extensions to further improve this new approach. For example, it is known that the traditional SFA algorithm makes no distinction between the “what information” in natural scenes and the “where information”. We suggest augmenting the continuous latent space with binary gating variables to produce a model capable of representing these two types of information distinctly. Finally we noted that probabilistic models are not compatible with the standard kernel method of expanding the observations through a large family of non-linearities. However an interesting alternative is to learn the non-linearities in the model directly. We suggest both a traditional method using neural networks to parameterise the mapping from latent variable to observation, and a non-traditional method that learns the inverse mapping and is therefore more in the spirit of SFA. The power of these new models will be explored in future work.

7 Acknowledgements

We thank Pietro Berkes for helpful discussions and for helping us to clarify the presentation. This work was supported by the Gatsby Charitable Foundation.

8 References

- Attias, H. (1999). Independent factor analysis. *Neural Comp*, 11(4), 803–851.
- Bell, A. J., & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision Research*, 37(23), 3327–3338.
- Berkes, P., & Wiskott, L. (2005). Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of Vision*, 5(6), 579–602.
- Blaschke, A. J., Berkes, P., & Wiskott, L. (2006). What is the relation between slow feature analysis and independent component analysis? *Neural Computation*, 18(10), In Press.
- Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society Series B (Methodological)*, 39, 1–38.
- Földiák, P. (1991). Learning invariance from transformation sequences. *Neural Computation*, 3(2), 194–200.
- Ghahramani, Z., & Roweis, S. T. (1999). Learning nonlinear dynamical systems using an EM algorithm. In M. S. Kearns, S. A. Solla, & D. A. Cohn (Eds.), *Advances in Neural Information Processing Systems*, Vol. 11 (pp. 599–605). Cambridge, MA: MIT Press.
- Grimes, D. B., & Rao, R. P. N. (2005). Bilinear sparse coding for invariant vision. *Neural Comput*, 17(1), 47–73.
- Hinton, G. E. (1989). Connectionist learning procedures. *Artificial Intelligence*, 40(1-3), 185–234.
- Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8), 1771–800.
- Hurri, J., & Hyvärinen, A. (2003). Temporal and spatiotemporal coherence in simple-cell responses: a generative model of natural image sequences. *Network*, 14(3), 527–51.
- Hyvärinen, A., Hurri, J., & Väyrynen, J. (2003). Bubbles: a unifying framework for low-level statistical properties of natural image sequences. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 20(7), 1237–52.
- Jordan, M. I., Ghahramani, Z., Jaakkola, T., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2), 183–233.

- Karklin, Y., & Lewicki, M. S. (2005). A hierarchical bayesian model for learning non-linear statistical regularities in nonstationary natural signals. *Neural Computation*, *17*(2), 397–423.
- Kayser, C., Körding, K. P., & König, P. (2003). Learning the nonlinearity of neurons from natural visual stimuli. *Neural Computation*, *15*(8), 1751–9.
- Körding, K. P., Kayser, C., Einhäuser, W., & König, P. (2004). How are complex cell properties adapted to the statistics of natural stimuli? *Journal of Neurophysiology*, *91*(1), 206–12.
- MacKay, D. J. C. (1999). *Maximum likelihood and covariant algorithms for independent component analysis*. Unpublished manuscript available from <ftp://wol.ra.phy.cam.ac.uk/pub/mackay/ica.ps.gz>.
- Miskin, J. (2000). *Ensemble learning for independent components analysis*. PhD thesis, University of Cambridge, Cambridge.
- Mitchison, G. (1991). Removing time variation with the anti-hebbian differential synapse. *Neural Computation*, *3*, 312–20.
- Neal, R. M., & Hinton, G. E. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan (Ed.), *Learning in graphical models* (pp. 355–370). Kluwer Academic Press.
- Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*(6583), 607–9.
- Osindero, S., Welling, M., & Hinton, G. E. (2006). Topographic product models applied to natural scene statistics. *Neural Computation*, *18*(2), 381–414.
- Pearlmutter, B., & Parra, L. (1997). Maximum likelihood blind source separation: A context-sensitive generalization of ICA. In M. C. Mozer, M. I. Jordan, & T. Petche (Eds.), *Advances in Neural Information Processing Systems*, Vol. 9 (pp. 613–9). Cambridge, MA: MIT Press.
- Roweis, S., & Ghahramani, Z. (1999). A unifying review of linear gaussian models. *Neural Computation*, *11*(2), 305–45.
- Särelä, J., & Valpola, H. (2005). Denoising source separation. *J. Machine Learning Res.*, *6*, 233–72.
- Stone, J. V. (1996). Learning perceptually salient visual parameters using spatiotemporal smoothness constraints. *Neural Computation*, *8*(7), 1463–92.
- Tenenbaum, J. B., & Freeman, W. T. (2000). Separating style and content with bilinear models. *Neural Computation*, *12*(6), 1247–83.
- Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal components analysis. *Journal of the Royal Statistical Society Series B (Methodological)*, *61*(3), 611–22.
- Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: unsupervised learning of invariances. *Neural Computation*, *14*(4), 715–70.

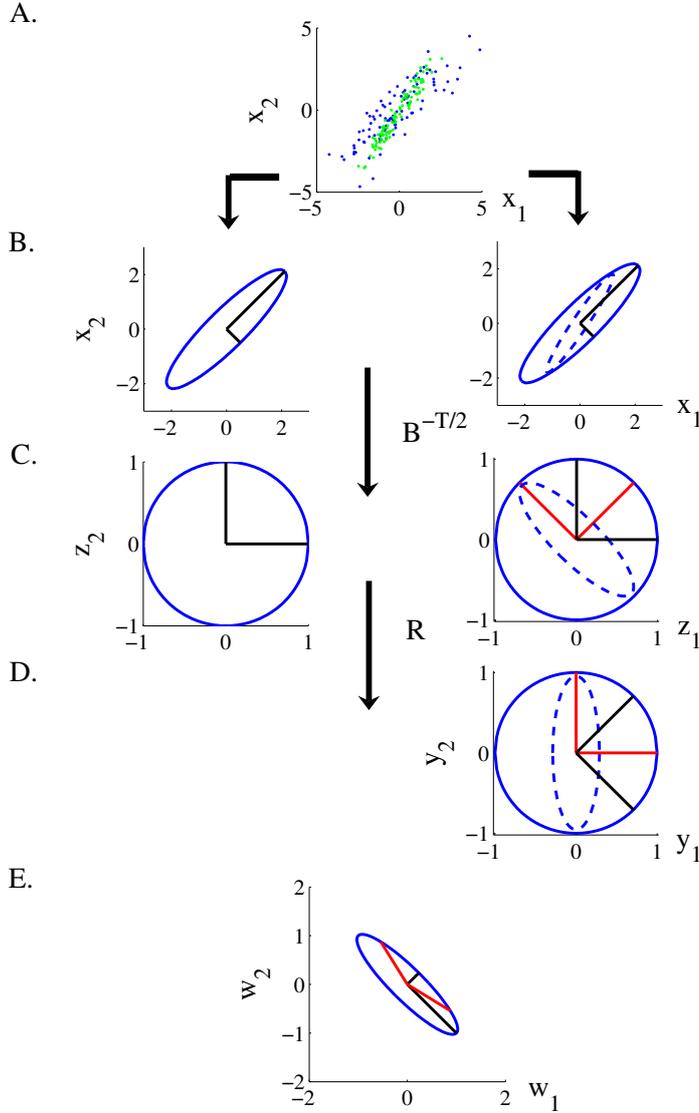


Figure 1: The geometrical interpretation of PCA and SFA in two dimensions. The left column illustrates PCA whilst the right column illustrates SFA. A) The data $x_{1,t}$, $x_{2,t}$ (blue dots), and the temporal differences $\Delta x_{1,t}$, $\Delta x_{2,t}$ (green dots). B) Data space, Left panel: The covariance of the data illustrated schematically by a blue ellipse: $x^T \mathbf{B} x = 1$, the principal axes are shown in black. Right: the covariance of the data, and the covariance of the time derivatives (blue dotted ellipse) $x^T \mathbf{A} x = 1$. C) Sphered (PCA) space resulting from the linear transform $\mathbf{B}^{-T/2}$ (a rotation determined by the eigenvectors of \mathbf{B} and a rescaling by the square-root of the corresponding eigenvalues). Left panel: The sphered covariance matrix (blue circle), the principal axes of which (black) are now of equal length. Right panel: The sphered covariance and the transformed covariance of the time derivatives (dotted ellipse), the principal axes of which (red lines) are not aligned with the axes of the sphered space. D) SFA space: An additional rotation \mathbf{R} is made to align the axes of the space with the transformed time difference covariance. E) PCA and SFA weights \mathbf{W} shown in weight space. Other choices for the rotation matrix correspond to points on the ellipsoid chosen to be orthogonal in the metric of the ellipse.

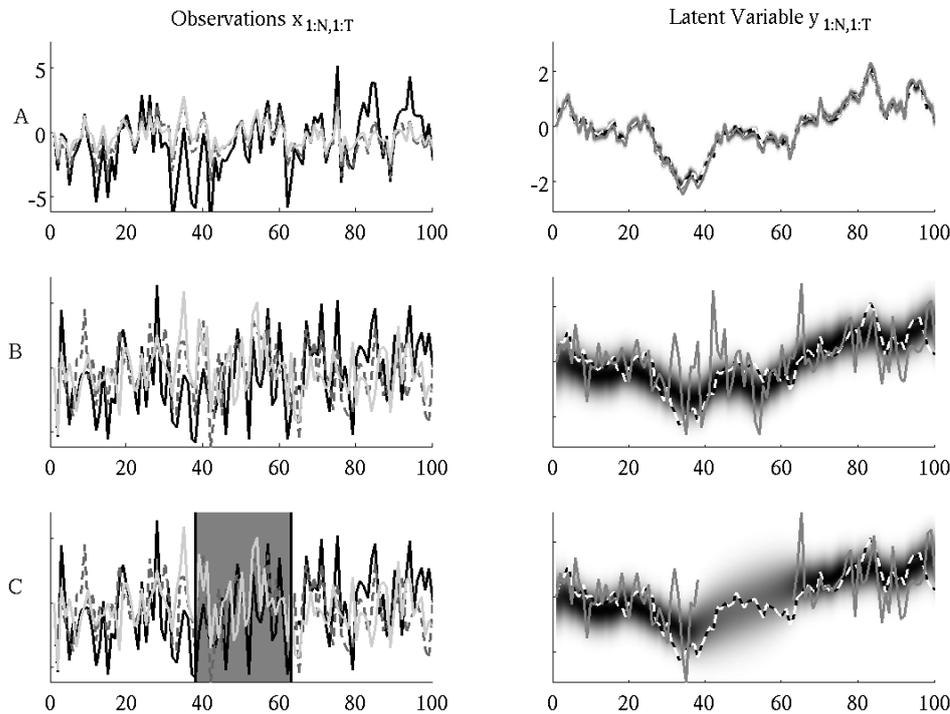


Figure 2: Comparisons between slow feature analysis and probabilistic SFA on three different data sets drawn from the forward model. Left hand column: the three dimensional observations. Right column: the slowest latent variable/extracted feature. The dashed line is the actual value of the slowest latent, the gray line is the slowest feature extracted by slow feature analysis, and the grey scale indicates the posterior distribution from the probabilistic model. Row A) Low observation noise: both SFA and probabilistic SFA recover the true latents accurately. For each algorithm we calculate the root-mean-square (RMS) error between the reconstructed and true value of the latent process. The difference between these errors, divided by the RMS amplitude of the latent process gives the proportional difference in errors, D . In this case, $D = 0.03$, indicating that the methods perform similarly at low observation noise levels. Row B) Higher observation noise: SFA is adversely effected, but the probabilistic algorithm makes reasonable predictions. The difference in the RMS errors is now comparable to the amplitude of the latent signal ($D = 1.10$). Row C) As for B), but with missing data: the probabilistic model can predict the intermediate latents in a principled manner, but SFA cannot. Using an ad hoc straight line interpolation of the normal SFA solution, the RMS error over the region of missing data is substantially greater ($D = 1.4$), indicating the superiority of the probabilistic interpolation.

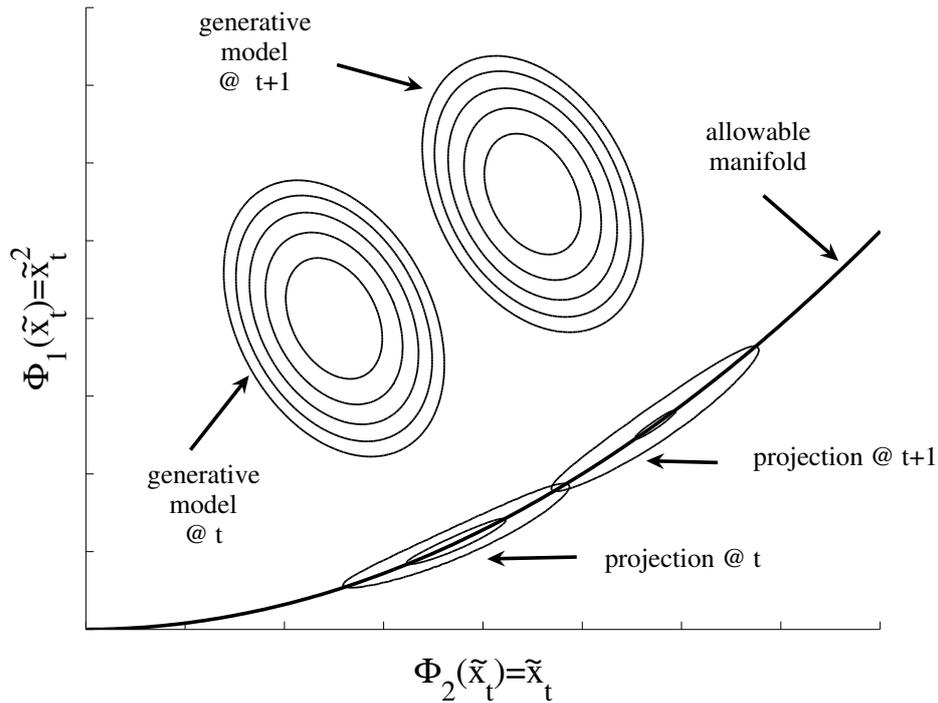


Figure 3: A schematic of the product of experts model in expanded observation space: The observations are one dimensional, and are expanded to two dimensions: $\phi_1(\tilde{x}_t) = \tilde{x}_t$ and $\phi_2(\tilde{x}_t) = \tilde{x}_t^2$. The generative model assigns non-zero probability off the manifold as shown by its energy contours (shown at two successive time steps to indicate a slow direction and a fast direction). The local experts have high energy only around a localised section of the realisable manifold. The product of the two types of expert leads to a projected process with slow latents that generates around the manifold. The width around the manifold is shown for illustration purposes: in the limit $\beta \rightarrow 0$ the projected process lies precisely on the manifold.