University of Pennsylvania
ScholarlyCommons

Departmental Papers (CIS) | Department of Computer & Information Science

9-1-2003

Statistical signal processing with nonnegativity constraints

Lawrence K. Saul
University of Pennsylvania, lsaul@cis.upenn.edu

Fei Sha
University of Pennsylvania, feisha@seas.upenn.edu

Daniel D. Lee
University of Pennsylvania, ddlee@seas.upenn.edu

Follow this and additional works at: http://repository.upenn.edu/cis_papers

Part of the Computer Engineering Commons, and the Computer Sciences Commons

Recommended Citation

Lawrence K. Saul, Fei Sha, and Daniel D. Lee, "Statistical signal processing with nonnegativity constraints", . September 2003.

Proceedings of the Eighth European Conference on Speech Communication and Technology, volume 2, pages 1001-1004, held 1-4 September 2003 in Geneva, Switzerland.

This paper is posted at ScholarlyCommons. http://repository.upenn.edu/cis_papers/13
For more information, please contact repository@pobox.upenn.edu.

Statistical signal processing with nonnegativity constraints

Abstract
Nonnegativity constraints arise frequently in statistical learning and pattern recognition. Multiplicative updates provide natural solutions to optimizations involving these constraints. One well known set of multiplicative updates is given by the Expectation-Maximization algorithm for hidden Markov models, as used in automatic speech recognition. Recently, we have derived similar algorithms for nonnegative deconvolution and nonnegative quadratic programming. These algorithms have applications to low-level problems in voice processing, such as fundamental frequency estimation, as well as high-level problems, such as the training of large margin classifiers. In this paper, we describe these algorithms and the ideas that connect them.

Disciplines
Computer Engineering | Computer Sciences

Comments
Proceedings of the Eighth European Conference on Speech Communication and Technology, volume 2, pages 1001-1004, held 1-4 September 2003 in Geneva, Switzerland.

This conference paper is available at ScholarlyCommons: http://repository.upenn.edu/cis_papers/13

Statistical Signal Processing with Nonnegativity Constraints

Lawrence K. Saul[1], *Fei Sha*[1], *Daniel D. Lee*[2]

[1]Department of Computer and Information Science
[2]Department of Electrical and System Engineering
University of Pennsylvania, Philadephia, PA
{lsaul,feisha,ddlee}@seas.upenn.edu

Abstract

Nonnegativity constraints arise frequently in statistical learning and pattern recognition. Multiplicative updates provide natural solutions to optimizations involving these constraints. One well known set of multiplicative updates is given by the Expectation-Maximization algorithm for hidden Markov models, as used in automatic speech recognition. Recently, we have derived similar algorithms for nonnegative deconvolution and nonnegative quadratic programming. These algorithms have applications to low-level problems in voice processing, such as fundamental frequency estimation, as well as high-level problems, such as the training of large margin classifiers. In this paper, we describe these algorithms and the ideas that connect them.

1. Introduction

Many problems in statistical learning and pattern recognition involve optimizations that cannot be solved in closed form. For these problems, iterative updates are required that converge in the limit to actual solutions. Quite often, objective functions have structure that can be exploited in their optimizations; for example, they give rise to auxiliary functions—lower or upper bounds—whose optimizations are themselves tractable and guaranteed to improve the objective function at each iteration[1, 8, 13]. The use of auxiliary functions to derive iterative updates has emerged as a powerful alternative to gradient-based methods.

Particularly useful updates have been derived in this way for a large number of problems involving nonnegativity constraints. In this paper, we examine three such problems in statistical learning and pattern recognition. An interesting trend in these problems has been the emergence of *multiplicative* updates. Our goals in this paper are to describe these multiplicative updates in simple terms, to develop their shared intuitions, and to sketch their applications to problems in voice processing.

2. Maximum likelihood estimation

We begin by reviewing multiplicative updates and nonnegativity constraints in a familiar context: maximum likelihood (ML) estimation in discrete hidden Markov models (HMMs)[1]. Consider an HMM with n hidden states $s \in \{1, 2, .., n\}$ and m observations $o \in \{1, 2, ..., m\}$. The parameters of the HMM are the transition matrix $a_{ii'} = P(s_{t+1} = i' | s_t = i)$, the emission matrix $b_{ij} = P(o_t = j | s_t = i)$, and the initial distribution $\pi_k = P(s_1 = k)$. These parameters obey simplex constraints: they are nonnegative, and the distributions they represent must be properly normalized. The goal of ML estimation is to maximize the log-likelihood $\mathcal{L} = \log P(o_1, o_2, .., o_T)$ of one or more observation sequences.

2.1. Multiplicative updates

The Expectation-Maximization (EM) algorithm prescribes a set of iterative updates for ML estimation. The update for $a_{ii'}$ can be written in terms of the gradient of the log-likelihood as:

$$a_{ii'} \leftarrow a_{ii'} \left[\frac{\partial \mathcal{L} / \partial a_{ii'}}{\sum_k a_{ik} (\partial \mathcal{L} / \partial a_{ik})} \right]. \tag{1}$$

The updates for the parameters b_{ij} and π_i have a similar form. Notably, these update rules are guaranteed to increase the log-likelihood $\mathcal{L}$ at each iteration. They are derived by constructing an auxiliary function which provides a lower bound on $\mathcal{L}$.

We can view the EM algorithm as a set of multiplicative updates. The multiplicative form of eq. (1) is apparent from the factor that appears in square brackets. Note how the multiplicative update hinges on the nonnegativity of the gradient; in particular, if it were not true that $\partial \mathcal{L} / \partial a_{ii'} \geq 0$ then the update in eq. (1) would violate the nonnegativity constraints on $a_{ii'}$.

2.2. Discrete Bayesian networks

Discrete HMMs are a special case of discrete Bayesian networks—directed acyclic graphs whose nodes represent discrete random variables and whose edges represent conditional dependencies. Many such extensions of HMMs are being investigated for automatic speech recognition[10, 19]. The EM updates in discrete Bayesian networks have the same multiplicative form as eq. (1), with parameters rescaled by partial derivatives of the log-likelihood, then renormalized to sum to one. The nonnegativity of these derivatives makes possible the simple multiplicative form of the updates. The updates in sections 3 and 4 will exhibit interesting variations on this theme.

3. Nonnegative deconvolution

The problem of linear deconvolution is to estimate an unobserved signal $\mathbf{x}$ from an observed signal $\mathbf{y} = \mathbf{W}\mathbf{x}$. The matrix $\mathbf{W}$ is assumed to be known. The problem of nonnegative deconvolution[7] arises when the matrix $\mathbf{W}$ and the vectors $\mathbf{x}$ and $\mathbf{y}$ are constrained to be nonnegative. A generalization of the Kullback-Leibler (KL) divergence provides a natural measure of distance between nonnegative vectors. In this framework, the distance between the vectors $\mathbf{y}$ and $\mathbf{W}\mathbf{x}$ is computed as:

$$G(\mathbf{x}) = \sum_i \left[y_i \log \frac{y_i}{(\mathbf{W}\mathbf{x})_i} - y_i + (\mathbf{W}\mathbf{x})_i \right]. \tag{2}$$

Given $\mathbf{y}$ and $\mathbf{W}$, we can estimate $\mathbf{x}$ by minimizing eq. (2); the function is convex and lower bounded by zero, with $G(\mathbf{x}) = 0$ if and only if $\mathbf{y} = \mathbf{W}\mathbf{x}$. The minimum of eq. (2), however, cannot be computed in closed form, and an iterative solution is needed.

3.1. Multiplicative updates

Multiplicative updates for this optimization are derived by writing the cost function as $G(\mathbf{x}) = G^+(\mathbf{x}) - G^-(\mathbf{x})$, where

$$G^+(\mathbf{x}) = \sum_i \left[(\mathbf{W}\mathbf{x})_i - y_i\right], \quad (3)$$

$$G^-(\mathbf{x}) = \sum_i y_i \left[\log(\mathbf{W}\mathbf{x})_i - \log y_i\right]. \quad (4)$$

The gradient of $G(\mathbf{x})$ can similarly be decomposed in terms of contributions from these two pieces:

$$\partial G^+/\partial x_j = \sum_i W_{ij}, \quad (5)$$

$$\partial G^-/\partial x_j = \sum_i y_i W_{ij}/(\mathbf{W}\mathbf{x})_i. \quad (6)$$

Note that these partial derivatives are themselves nonnegative. The multiplicative updates for nonnegative deconvolution hinge on the nonnegativity of these derivatives just as the EM updates in section 2 hinge on the nonnegativity of derivatives of the log-likelihood. In particular, the updates for minimizing eq. (2) take the form:

$$x_j \leftarrow x_j \left[\frac{\partial G^-/\partial x_j}{\partial G^+/\partial x_j}\right]. \quad (7)$$

The fixed points of these updates have a simple intuition. One fixed point occurs at $x_j = 0$; the other occurs when the numerator and denominator in eq. (7) are perfectly balanced, implying that $\partial G^+/\partial x_j = \partial G^-/\partial x_j$, or $\partial G/\partial x_j = 0$. It can be shown that the function $G(\mathbf{x})$ decreases monotonically to the value of its global minimum under these updates[8]. The proof of convergence relies on the construction of an auxiliary function which provides an upper bound on $G(\mathbf{x})$. The algorithm is similar in this respect to the EM algorithm of section 2. The updates in eq. (7) have been successfully applied to a number of problems in science and engineering[7].

3.2. Fundamental frequency estimation

The fundamental frequency f_0 of a periodic signal is equal to the reciprocal of the minimum period at which it repeats itself. Given a mixture of periodic signals (or sources), a classic problem is to deduce the number of sources and the value of f_0 for each source. The problem is of great interest because a sound's fundamental frequency corresponds (in most cases) to its perceived pitch as registered by the human auditory system[6].

It is helpful to visualize this problem in the frequency domain. Fig. 1 shows the time domain waveforms and magnitude spectrum of four periodic signals with $f_0 = 100$ Hz. Fig. 2 shows the same for four mixtures of periodic signals with $f_0 = 100$ Hz and $f_0 = 173$ Hz. How can we deduce the number of periodic sources from observations of this form,[1] as well as their fundamental frequencies?

We will formulate this problem as a nonnegative deconvolution. Let the elements of the observed vector $\mathbf{y}$ store a signal's magnitude spectrum, as shown in Figs. 1 and 2, with the exception that indices of these elements correspond to frequencies that are equally spaced on a log scale. Thus, the nonzero elements of $\mathbf{y}$ in this setting correspond to nonzero frequency components (or partials) of the individual sound sources, assumed to be periodic.

[1] Pitch tracking of speech and music involves additional complications. Signals are not perfectly periodic: they are nonstationary, corrupted by noise, and only approximately periodic over short time scales. Nevertheless, it remains instructive to consider the idealized problem sketched above.

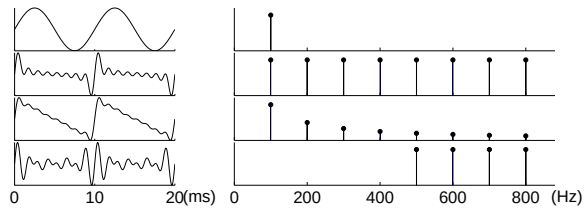

Figure 1: Time domain waveforms and magnitude spectra of different periodic signals with the same fundamental frequency, $f_0 = 100$ Hz. From top to bottom: sinusoid, impulse train, sawtooth wave, and highpass filtered impulse train with missing fundamental.

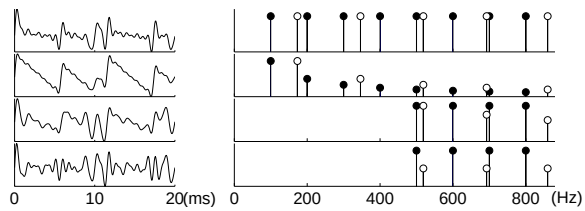

Figure 2: Time domain waveforms and magnitude spectra for mixtures of two periodic signals, one with $f_0 = 100$ Hz, the other with $f_0 = 173$ Hz. Shaded nodes denote harmonics of the $f_0 = 100$ Hz signal.

Likewise, let the unobserved vector $\mathbf{x}$ encode the number of periodic sources, with precisely one nonzero element per source, and with indices that correspond to possible fundamental frequencies (also spaced on a log scale). Thus, for example, the "target" vector $\mathbf{x}$ for the signals in Fig. 1 has precisely one nonzero element at the index representing $f_0 = 100$ Hz, while the "target" vector $\mathbf{x}$ for the signals in Fig. 2 has precisely two nonzero elements, one at the index representing $f_0 = 100$ Hz, the other at the index representing $f_0 = 173$ Hz.

We obtain $\mathbf{x}$ from the nonnegative deconvolution of $\mathbf{y}$ by minimizing eq. (2). This requires specifying the matrix $\mathbf{W}$. Let each column of $\mathbf{W}$ store a basis function, or template, for the magnitude spectra of a periodic source whose fundamental frequency matches the corresponding index in the vector $\mathbf{x}$. In practice, we set the matrix $\mathbf{W}$ to be a discretized, truncated, and smoothed approximation to the continuum kernel $W(f, f_0)$ given by:

$$W(f, f_0) = \sum_n h_n \, \delta(f - n f_0), \quad (8)$$

where the sum is over all positive integers n. Intuitively, eq. (8) describes each basis function as a weighted harmonic stack. Suitable values for the coefficients h_n were found by experimentation; in practice, we used $h_n = 0.7 + 0.3/n$. The basis function for $f_0 = 100$ Hz is plotted in Figure 3. Note that basis functions for different values of f_0 are related by simple translations on the log-frequency axis.

Estimating f_0 of overlapping sources by nonnegative deconvolution is based on the familiar idea of harmonic template matching[4]. We imagine that observed partials (such as the ones in Figs. 1 and 2) are generated by a weighted nonnegative combination of harmonic stacks (as shown in Fig. 3). The nonzero elements of $\mathbf{x}$ encode the weights in this combination, as expressed by the convolution $\mathbf{y} = \mathbf{W}\mathbf{x}$. Note that the cost function in eq. (2) diverges if $(\mathbf{W}\mathbf{x})_i = 0$ when y_i is nonzero; this useful property ensures that minima of eq. (2) explain each

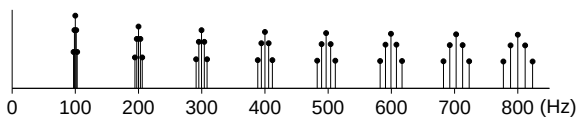

Figure 3: The discretized basis function for $f_0 = 100$ Hz stored by one column of the matrix $\mathbf{W}$, obtained from eq. (8). The basis functions are smoothed by assigning small weights to neighboring log-frequency bins around each harmonic. There are 48 bins per octave.

observed partial by its attribution to one or more sources. Nonnegative deconvolution with the basis functions in Fig. 3 gives the correct solution for all the examples in Figs. 1 and 2, as well as many other signals (isolated and mixed) that we have tested.

There exists a large body of related work on fundamental frequency estimation of overlapping sources[2, 5, 15, 17, 18]. Nonnegative deconvolution is similar to EM algorithms[5] for harmonic template matching, but it does not impose normalization constraints on spectral peaks as if they represented a probability distribution. Our approach differs from subtractive methods[2] that rely on extracting a predominant source, cancelling out its frequency components, and analyzing the residual signal. In particular, given a mixture of sources, the minimization of eq. (2) solves simultaneously for all component weights and fundamental frequencies. (It does not, however, assume as prior knowledge the number of sources; this is obtained simply from the final number of nonzero elements in $\mathbf{x}$.) Our approach also does not rely on heuristics for discovering harmonic ratios, such as approximating fractions by ratios of small integers and computing greatest common denominators.

We are currently investigating nonnegative deconvolution as a back end for fundamental frequency estimation in speech and music. These signals do not have the idealized spectra shown in Figs. 1 and 2, but we can use an auditory filterbank to resolve their lower-order harmonics[6], followed by simple sinusoid detectors to estimate and track the frequencies of observed partials[12]. The windowed output of such a front end can then be analyzed by nonnegative deconvolution.

4. Nonnegative quadratic programming

Lastly, we consider the problem of nonnegative quadratic programming. Here, the goal is to minimize the objective function:

$$F(\mathbf{v}) = \frac{1}{2}\mathbf{v}^{\mathrm{T}}\mathbf{A}\mathbf{v} + \mathbf{b}^{\mathrm{T}}\mathbf{v}, \tag{9}$$

subject to constraints $v_i \geq 0$ for all i. In what follows, we assume that the matrix $\mathbf{A}$ is symmetric and semipositive definite. Thus, the objective function $F(\mathbf{v})$ is bounded below, and its optimization is convex. Due to the nonnegativity constraints, however, there does not exist an analytical solution for the global minimum (or minima), and an iterative solution is needed.

4.1. Multiplicative updates

We have derived multiplicative updates for the optimization[13] of eq. (9), expressed in terms of the positive and negative components of the matrix $\mathbf{A}$. Let $\mathbf{A}^+$ and $\mathbf{A}^-$ denote the *nonnegative* matrices with elements:

$$A^+_{ij} = \begin{cases} A_{ij} & \text{if } A_{ij} > 0, \\ 0 & \text{otherwise,} \end{cases} \qquad A^-_{ij} = \begin{cases} |A_{ij}| & \text{if } A_{ij} < 0, \\ 0 & \text{otherwise.} \end{cases}$$

It follows that $\mathbf{A} = \mathbf{A}^+ - \mathbf{A}^-$. In terms of these nonnegative matrices, the objective function can be decomposed as $F(\mathbf{v}) = F_a(\mathbf{v}) + F_b(\mathbf{v}) - F_c(\mathbf{v})$, where we use the first and third of these terms to "split" the quadratic piece of $F(\mathbf{v})$, and the second term to capture the linear piece:

$$F_a(\mathbf{v}) = \frac{1}{2}\mathbf{v}^{\mathrm{T}}\mathbf{A}^+\mathbf{v}, \tag{10}$$

$$F_b(\mathbf{v}) = \mathbf{b}^T\mathbf{v}, \tag{11}$$

$$F_c(\mathbf{v}) = \frac{1}{2}\mathbf{v}^{\mathrm{T}}\mathbf{A}^-\mathbf{v}. \tag{12}$$

The gradient of $F(\mathbf{v})$ can be similarly decomposed in terms of contributions from these three pieces. We have chosen our notation in eqs. (9) and (11) so that $b_i = \partial F_b/\partial v_i$; for the quadratic terms in the objective function, we define the corresponding derivatives:

$$a_i = \partial F_a/\partial v_i = (\mathbf{A}^+\mathbf{v})_i, \tag{13}$$

$$c_i = \partial F_c/\partial v_i = (\mathbf{A}^-\mathbf{v})_i. \tag{14}$$

Note that these partial derivatives are themselves nonnegative: that is, $a_i \geq 0$ and $c_i \geq 0$. Our multiplicative updates for nonnegative quadratic programming hinge on the nonnegativity of these derivatives, just as the updates in previous sections. In particular, they take the form:

$$v_i \longleftarrow v_i \left[\frac{-b_i + \sqrt{b_i^2 + 4a_i c_i}}{2a_i} \right]. \tag{15}$$

These updates are meant to be applied in parallel to all the elements of $\mathbf{v}$. They are remarkably simple to implement, neither involving a learning rate nor other heuristic criteria that must be tuned to ensure convergence. Previously, we have shown that the function $F(\mathbf{v})$ in eq. (9) decreases monotonically to the value of its global minimum under these updates[13]. As in sections 2 and 3, the proof of convergence relies on the construction of an auxiliary function.

The reader will recognize the factor multiplying v_i on the right hand side of eq. (15) as the quadratic formula for the positive root of the polynomial $a_i z^2 + b_i z - c_i$. This factor is guaranteed to be nonnegative, as we observed earlier that $a_i \geq 0$ and $c_i \geq 0$. The updates thus naturally enforce the nonnegativity constraints on v_i. An intuition for these multiplicative updates can be gained by examining their fixed points. One fixed point for eq. (15) occurs at $v_i^* = 0$; the other occurs when the positive root of the polynomial $a_i z^2 + b_i z - c_i = 0$ is located at $z = 1$, since in this case the multiplicative factor in eq. (15) is equal to unity. The latter condition, together with the definitions in eqs. (13–14), implies that $(\partial F/\partial v_i)|_{\mathbf{v}^*} = a_i + b_i - c_i = 0$. Thus the two criteria for fixed points are either (i) $v_i^* = 0$, or (ii) $(\partial F/\partial v_i)|_{\mathbf{v}^*} = 0$.

Further intuition is gained by considering the effects of the multiplicative update away from its fixed points. Although the partial derivative $\partial F/\partial v_i$ does not appear explicitly in eq. (15), there is a close link between the sign of this derivative and the effect of the update on v_i. In particular, using the fact that $\partial F/\partial v_i = a_i + b_i - c_i$, it is easy to show that the update decreases v_i if $\partial F/\partial v_i > 0$ and increases v_i if $\partial F/\partial v_i < 0$. Thus, the update in eq. (15) moves each element v_i in the same direction as gradient descent (though not by the same amount).

4.2. Support vector machines

Various problems in nonnegative quadratic programming arise in the training of large margin classifiers, such as support vector

machines (SVMs)[16]. In SVMs, kernel methods are used to map inputs into a higher, potentially infinite, dimensional feature space; the decision boundary between classes is then identified as the maximum margin hyperplane in the feature space. SVMs currently provide state-of-the-art solutions to many problems in statistical learning. There have also been promising applications of SVMs to automatic speech recognition[9, 14].

We briefly review the problem of computing the maximum margin hyperplane in SVMs[16]. Let $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ denote labeled examples with binary class labels $y_i = \pm 1$, and let $K(\mathbf{x}_i, \mathbf{x}_j)$ denote the kernel dot product between inputs. For brevity, we consider only the simple case where in the high dimensional feature space, the classes are linearly separable and the hyperplane is required to pass through the origin. In this case, the maximum margin hyperplane is obtained by minimizing the loss function:

$$L(\alpha) \;=\; -\sum_i \alpha_i + \frac{1}{2}\sum_{ij} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j), \qquad (16)$$

subject to the nonnegativity constraints $\alpha_i \geq 0$. Let α^* denote the minimum of eq. (16). The maximal margin hyperplane has normal vector $\mathbf{w} = \sum_i \alpha_i^* y_i \mathbf{x}_i$ and satisfies the margin constraints $y_i K(\mathbf{w}, \mathbf{x_i}) \geq 1$ for all examples in the training set.

The loss function in eq. (16) is a special case of eq. (9) with $A_{ij} = y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$ and $b_i = -1$. Thus, the multiplicative updates in eq. (15) are easily adapted to SVMs. This algorithm for training SVMs is known as Multiplicative Margin Maximization (M^3). The algorithm can be generalized[13] to data that is not linearly separable and to separating hyperplanes that do not pass through the origin.

Many iterative algorithms have been developed for nonnegative quadratic programming in general and for SVMs as a special case. Benchmarking experiments have shown that M^3 is a feasible algorithm for small to moderately sized data sets. On the other hand, it does not converge as fast as leading subset methods[3, 11] for large data sets. Nevertheless, the extreme simplicity and convergence guarantees of M^3 make it a useful starting point for experimenting with SVMs.

5. Discussion

The multiplicative updates in this paper exhibit an interesting progression. The updates for ML estimation hinge on the nonnegativity of the gradient of the log-likelihood. The updates for nonnegative deconvolution hinge on the nonnegativity of partial derivatives obtained from the *two-way* decomposition in eqs. (3–4). Finally, the updates for nonnegative quadratic programming hinge on the nonnegativity of partial derivatives obtained from the *three-way* decomposition in eqs. (10–12). Future work will continue to develop the theoretical foundations and practical applications of these multiplicative updates.

6. References

[1] L. Baum. An inequality and associated maximization technique in statistical estimation of probabilistic functions of Markov processes. *Inequalities*, 3:1–8, 1972.

[2] A. de Cheveigne and H. Kawahara. Multiple period estimation and pitch perception model. *Speech Communication*, 27:175–185, 1999.

[3] T. Friess, N. Cristianini, and C. Campbell. The Kernel Adatron algorithm: a fast and simple learning procedure for support vector machines. In *Proceedings of the Fifteenth International Conference on Machine Learning*. Morgan Kaufman, 1998.

[4] J. Goldstein. An optimum processor theory for the central formation of the pitch of complex tones. *Journal of the Acoustical Society of America*, 54:1496–1516, 1973.

[5] M. Goto. A robust predominant-F0 estimation method for real-time detection of melody and bass lines in CD recordings. In *Proceedings of ICASSP-2000*, pages 757–760, 2000.

[6] W. M. Hartmann. Pitch, periodicity, and auditory organization. *Journal of the Acoustical Society of America*, 100(6):3491–3502, 1996.

[7] D. D. Lee and H. S. Seung. Learning the parts of objects with nonnegative matrix factorization. *Nature*, 401:788–791, 1999.

[8] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural and Information Processing Systems*, volume 13, Cambridge, MA, 2001. MIT Press.

[9] P. Niyogi, C. Burges, and P. Ramesh. Distinctive feature detection using support vector machines. In *Proceedings of ICASSP-99*, pages 425–428, 1999.

[10] H. Nock and S. Young. Modelling asynchrony in automatic speech recognition using loosely coupled HMMs. *Cognitive Science*, 26(3):283–301, 2002.

[11] J. Platt. Fast training of support vector machines using sequential minimal optimization. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods — Support Vector Learning*, pages 185–208, Cambridge, MA, 1999. MIT Press.

[12] L. K. Saul, D. D. Lee, C. L. Isbell, and Y. LeCun. Real time voice processing with audiovisual feedback: toward autonomous agents with perfect pitch. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*. MIT Press, 2003.

[13] F. Sha, L. K. Saul, and D. D. Lee. Multiplicative updates for large margin classifiers. Technical Report MS-CIS-03-12, Department of Computer and Information Science, University of Pennsylvania, 2003.

[14] N. Smith and M. Gales. Speech recognition using SVMs. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural and Information Processing Systems*, volume 14, Cambridge, MA, 2002. MIT Press.

[15] T. Tolonen and M. Karjalainen. A computationally efficient multipitch analysis model. *IEEE Transactions on Speech and Audio Processing*, 8(6):708–716, 2000.

[16] V. Vapnik. *Statistical Learning Theory*. Wiley, N.Y., 1998.

[17] T. Virtanen and A. Klapuri. Separation of harmonic sounds using multipitch analysis and iterative parameter estimation. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2001.

[18] M. Wu, D. Wang, and G. J. Brown. A multipitch tracking algorithm for noisy speech. *IEEE Transactions on Speech and Audio Processing*, 2003 (in press).

[19] G. Zweig and S. Russell. Probabilistic modeling with Bayesian networks for automatic speech recognition. *Australian Journal of Intelligent Information Processing Systems*, 5(4):253–60, 1999.