Generalized Low-Rank Approximations

Nathan Srebro **Tommi Jaakkola**
Department of Electircal Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139
nati@mit.edu,tommi@ai.mit.edu

Abstract

We study the frequent problem of approximating a target matrix with a matrix of lower rank. We provide a simple and efficient (EM) algorithm for solving *weighted* low rank approximation problems, which, unlike simple matrix factorization problems, do not admit a closed form solution in general. We analyze, in addition, the nature of locally optimal solutions that arise in this context, demonstrate the utility of accommodating the weights in reconstructing the underlying low rank representation, and extend the formulation to non-Gaussian noise models such as classification (collaborative filtering).

1 Introduction

Low-rank matrix approximation with respect to the squared or Frobenius norm has wide applicability in estimation and can be easily solved with singular value decomposition. For many application, however, the deviation between the observed matrix and the low-rank approximation has to be measured relative to a weighted-norm. While the extension to the weighted norm case is conceptually straightforward, standard algorithms (such as SVD) for solving the unweighted case do not carry over to the weighted case. Only the special case of a rank-one weight matrix (where the weights can be decomposed into row weights and column weights) can be solved directly, analogously to SVD [1]. Perhaps surprisingly, the weighted extension has attracted relatively little attention.

Weighted-norms can arise in several situations. A zero/one weighted-norm, for example, arises when some of the entries in the matrix are not observed. External estimates of the noise variance associated with each measurement may be available (e.g. gene expression analysis) and using weights inversely proportional to the noise variance can lead to better reconstruction of the underlying structure. In other applications, entries in the target matrix represent aggregates of many samples. When using *un*weighted low-rank approximations (e.g. for separating style and content [2]), we assume a uniform number of samples for each entry. By incorporating weights, we can account for varying numbers of samples in such situations.

Shpak [3] and Lu *et al.* [4] studied weighted-norm low-rank approximations for the design of two-dimensional digital filters where the weights arise from constraints of varying importance. Shpak studies gradient-based methods while Lu *et al.* suggested alternating-optimization methods. In both cases, rank-k approximations are greedily combined from

k rank-one approximations (unlike for the unweighted case, such a greedy procedure is sub-optimal).

We suggest optimization methods that are significantly more computationally efficient and simpler to implement (Section 2). We also consider other measures of deviation, beyond weighted-Frobenius norms. Such measures arise, for example, when the noise model associated with matrix elements is known, but is not Gaussian. Classification, rather than regression, also gives rise to different measures of deviation. Classification tasks over matrices arise, for example, in the context of collaborative filtering. To predict the unobserved entries, one can fit a partially observed binary matrix using a logistic model with an underlying low-rank representation (input matrix). In sections 3 and 4 we show how weighted-norm approximations can be applied as a subroutine for solving these more general low-rank problems.

We note that low-rank approximation can be viewed as an unconstrained matrix factorization problem. Lee and Seung [5] studied generalizations that impose (non-negative) constraints on the factorization and considered different measures of deviation, including versions of the KL-divergence appropriate for non-negative matrices.

2 Weighted Low-Rank Approximations

Given a target matrix $A \in \Re^{n\times d}$, a corresponding non-negative weight matrix $W \in \Re_+^{n\times d}$ and a desired (integer) rank k, we would like to find a matrix $X \in \Re^{n\times d}$ of rank (at most) k, that minimizes the weighted Frobenius distance $J(X) = \sum_{i,a} W_{i,a} \left(X_{i,a} - A_{i,a}\right)^2$.

2.1 A Matrix-Factorization View

It will be useful to consider the decomposition $X = UV'$ where $U \in \Re^{n\times k}$ and $V \in \Re^{d\times k}$. Since any rank-k matrix can be decomposed in such a way, and any pair of such matrices yields a rank-k matrix, we can think of the problem as an unconstrained minimization problem over pairs of matrices (U, V) with the minimization objective $J(U, V) = \sum_{i,a} W_{i,a} \left(X_{i,a} - (UV')_{i,a}\right)^2 = \sum_{i,a} W_{i,a} \left(X_{i,a} - \sum_\alpha U_{i,\alpha} V_{a,\alpha}\right)^2$.

This decomposition is not unique. For any invertible $R \in \Re^{k\times k}$, the pair (UR, VR^{-1}) provides a factorization equivalent to (U, V), and $J(U, V) = J(UR, VR^{-1})$, resulting in a k^2-dimensional manifold of equivalent solutions (an equivalence class of solutions consists of a collection such manifolds, asymptotically tangent to one another). In particular, any (non-degenerate) solution (U, V) can be orthogonalized to a (non-unique) equivalent orthogonal solution $\bar{U} = UR, \bar{V} = VR^{-1}$ such that $\bar{U}'\bar{U} = I$ and $\bar{V}'\bar{V}$ is a diagonal matrix.[1] Instead of limiting our attention only to orthogonal decompositions, it is simpler to allow any matrix pair (U, V), resulting in an unconstrained optimization problem (but remembering that we can always focus on an orthogonal representative).

We first revisit the well-studied case where all of the weights are equal to one. In this case, the partial derivatives of the objective J with respect to U, V are $\frac{\partial J}{\partial U} = 2(UV' - A)V$, $\frac{\partial J}{\partial V} = 2(VU' - A')U$. Solving $\frac{\partial J}{\partial U} = 0$ for U yields $U = AV(V'V)^{-1}$ and focusing on an orthogonal solution where $V'V = I$ and $U'U = \Lambda$ is diagonal, yields $U = AV$. Substituting back into $\frac{\partial J}{\partial V} = 0$, we have $0 = VU'U - A'U = V\Lambda - A'AV$. The columns of V are mapped by $A'A$ to multiples of themselves, i.e. they are eigenvectors of $A'A$. Thus, the gradient $\frac{\partial J}{\partial (U,V)}$ vanishes at an orthogonal (U, V) if and only if the columns of V are eigenvectors of $A'A$ and the columns of U are corresponding eigenvectors of AA',

[1] We slightly abuse the standard linear-algebra notion of "orthogonal" since we cannot always have both $\bar{U}'\bar{U} = I$ and $\bar{V}'\bar{V} = I$.

scaled by the square root of their eigenvalues. More generally, the gradient vanishes at any (U, V) if and only if the columns of U are spanned by eigenvectors of AA' and the columns of V are correspondingly spanned by eigenvectors of $A'A$. In terms of the singular value decomposition $A = U_0 S V_0'$, the gradient vanishes at (U, V) if and only if there exist matrices $Q_U' Q_V = I \in \Re^{k \times k}$ (or more generally, a zero/one diagonal matrix rather than I) such that $U = U_0 S Q_U$, $V = V_0 Q_V$.

The global minimum can be identified by investigating the value of the objective function at these critical points. Let $\sigma_1 \geq \cdots \geq \sigma_m$ be the eigenvalues of $A'A$. For critical (U, V) that are spanned by eigenvectors corresponding to eigenvalues $\{\sigma_q | q \in Q\}$, the error of $J(U, V)$ is given by the sum of the eigenvalues *not* in Q ($\sum_{q \notin Q} \sigma_q$), and so the global minimum is attained when the eigenvectors corresponding to the highest eigenvalues are taken. As long as there are no repeated eigenvalues, all (U, V) global minima correspond to the same low-rank matrix $X = UV'$, and belong to the same equivalence class (a collection of k^2-dimensional asymptotically tangent manifolds). If there are repeated eigenvalues, the global minima correspond to a polytope of low-rank approximations in X space (and in U, V space, form a collection of higher-dimensional asymptotically tangent manifolds).

What is the nature of the remaining critical points? For a critical point (U, V) spanned by eigenvectors corresponding to eigenvalues as above (assuming no repeated eigenvalues), the Hessian has exactly $\sum_{q \in Q} q - \binom{k}{2}$ negative eigenvalues: We can replace any eigencomponent with eigenvalue σ with an alternate eigencomponent not already in (U, V) with eigenvalue $\sigma' > \sigma$, decreasing the objective function. The change can be done gradually, replacing the component with a convex combination of the original and improved components. This results in a line between the two critical points which is a monotonic improvement path. Since there are $\sum_{q \in Q} q - \binom{k}{2}$ such pairs of eigencomponents, there are at least this many directions of improvements. Other than these directions of improvements, and the k^2 directions along the equivalence manifold corresponding to k^2 zero eigenvalues of the Hessian, all other eigenvalues of the Hessian are positive (except for very degenerate A, for which they might be zero).

Hence, in the unweighted case, all critical points that are not global minima are saddle points. Despite $J(U, V)$ not being a convex function, all of its local minima are global.

When weights are introduced, the critical point structure changes significantly. The partial derivatives become (with $\otimes$ denoting element-wise multiplication):

$$\frac{\partial J}{\partial U} = 2(W \otimes (UV' - A))V \qquad \frac{\partial J}{\partial V} = 2(W \otimes (VU' - A'))U \tag{1}$$

The equation $\frac{\partial J}{\partial U} = 0$ is still a linear system in U, and for a fixed V, it can be solved, recovering $U_V^* = \arg\min_U J(U, V)$ (since $J(U, V)$ is convex in U). However, the solution cannot be written using a single pseudo-inverse $V(V'V)$. Instead, a separate pseudo-inverse is required for each row $(U_V^*)_i$ of U_V^*:

$$(U_V^*)_i = (V' \underline{W_i} V)^{-1} V' \underline{W_i} A_i = \text{pinv}(\sqrt{\underline{W_i}} V)(\sqrt{\underline{W_i}} A_i) \tag{2}$$

where $\underline{W_i} \in \Re^{k \times k}$ is a diagonal matrix with the weights from the ith row of W on the diagonal, and A_i is the ith row of the target matrix[2]. In order to proceed as in the unweighted case, we would have liked to choose V such that $V' \underline{W_i} V = I$ (or is at least diagonal). Although we can do this for a single i, we cannot, in general, achieve this concurrently for all rows. The critical points of the weighted low-rank approximation problem, therefore, lack the eigenvector structure of the unweighted case.[3] Another implication of this is that the

[2]Here and throughout the paper, rows of matrices, such as A_i and $(U_V^*)_i$, are treated in equations as *column* vectors.

[3]When W is of rank one, concurrent diagonalization is possible, allowing an eigenvector-based solution to the weighted low-rank approximation problem [1].

incremental structure of unweighted low-rank approximations is lost: An optimal rank-k factorization cannot necessarily be extended to an optimal rank-$(k+1)$ factorization.

Lacking an analytic solution, we revert to numerical optimization methods to minimize $J(U,V)$. But instead of optimizing $J(U,V)$ by numerically searching over (U,V) pairs, we can take advantage of the fact that for a fixed V, we can calculate U_V^*, and therefore also the projected objective $J^*(V) = \min_U J(U,V) = J(U_V^*, V)$. The parameter space of $J^*(V)$ is of course much smaller than that of $J(U,V)$, making optimization of $J^*(V)$ more tractable. This is especially true in many typical applications where the the dimensions of A are highly skewed, with one dimension several orders of magnitude larger than the other (e.g. in gene expression analysis one often deals with thousands of genes, but only a few dozen experiments).

Recovering U_V^* using (2) requires n inversions of $k \times k$ matrices. The dominating factor is actually the matrix multiplications: Each calculation of $V'\underline{W_i}V$ requires $O(dk^2)$ operations, for a total of $O(ndk^2)$ operations. Although more involved than the unweighted case, this is still significantly less than the prohibitive $O(n^3k^3)$ required for each iteration in Lu *et al.* [4], or for Hessian methods on (U,V) [3], and is only a factor of k larger than the $O(ndk)$ required just to compute the prediction UV'.

After recovering U_V^*, we can easily compute not only the value of the projected objective, but also its gradient. Since $\frac{\partial J(V,U)}{\partial U}\Big|_{U=U_V^*} = 0$, we have

$$\frac{\partial J^*(V)}{\partial V} = \frac{\partial J(V,U)}{\partial V}\Big|_{U=U_V^*} = 2(W \otimes (VU_V^{*\prime} - A'))U_V^*. \tag{3}$$

The computation requires only $O(ndk)$ operations, and is therefore "free" after U_V^* has been recovered.

The Hessian $\frac{\partial^2 J^*(V)}{\partial V^2}$ is also of interest for optimization. The mixed second derivatives with respect to a pair of rows V_a and V_b of V is (where δ_{ab} is the Kronecker delta):

$$\Re^{k\times k} \ni \frac{\partial^2 J^*(V)}{\partial V_a \partial V_b} = 2\sum_i \left(W_{ia}\delta_{ab}(U_V^*)_i(U_V^*)_i' - G_{ia}'(V'\underline{W_i}V)^{-1}G_{ja}(V_a) \right), \tag{4}$$

$$\text{where:} \quad G_{ia}(V_a) \stackrel{\text{def}}{=} W_{ia}(V_a(U_V^*)_i' + ((U_V^*)_i'V_a - A_{ia})I) \in \Re^{k\times k}. \tag{5}$$

By associating the matrix multiplications efficiently, the Hessian can be calculated with $O(nd^2k)$ operations, significantly more than the $O(ndk^2)$ operations required for recovering U_V^*, but still manageable when d is small enough.

Equipped with the above calculations, we can use standard gradient-descent techniques to optimize $J^*(V)$. Unfortunately, though, unlike in the unweighted case, $J(U,V)$, and $J^*(V)$, might have local minima that are not global. Figure 1 shows the emergence of a non-global local minimum of $J^*(V)$ for a rank-one approximation of $A = \left(\begin{smallmatrix} 1 & 1.1 \\ 1 & -1 \end{smallmatrix}\right)$. The matrix V is a two-dimensional vector. But since $J^*(V)$ is invariant under invertible scalings, V can be specified as an angle θ on a semi-circle. We plot the value of $J^*([\cos\theta, \sin\theta])$ for each θ, and for varying weight matrices of the form $W = \left(\begin{smallmatrix} 1+\alpha & 1 \\ 1 & 1+\alpha \end{smallmatrix}\right)$. At the front of the plot, the weight matrix is uniform and indeed there is only a single local minimum, but at the back of the plot, where the weight matrix emphasizes the diagonal, a non-global local minimum emerges.

The function $J^*(V)$ also has many saddle points, their number far surpassing the number of local minima. In most regions, the function is not convex. Therefore, Newton-Raphson methods are generally inapplicable except very close to a local minimum.

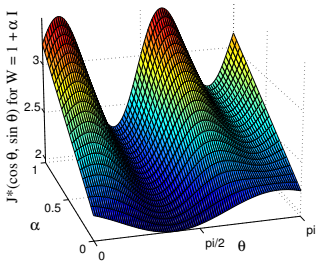

Figure 1: Emergence of local minima when the weights become non-uniform.

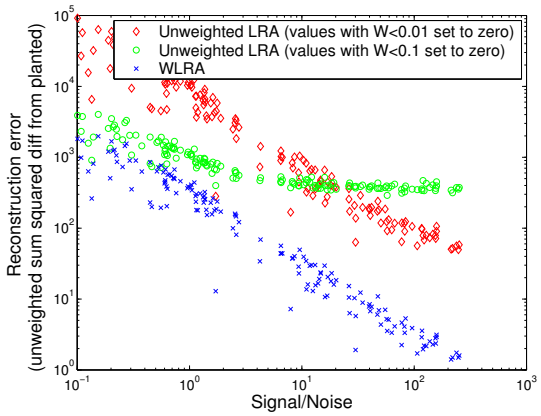

Figure 2: Reconstruction of a 1000×30 rank-three matrix.

2.2 A missing-values view and an EM procedure

The weighted low-rank approximation problem can also be viewed as a maximum likelihood problem with missing values. Consider first systems with only zero/one weights, where only some of the elements of the target matrix A are observed (those with weight one), while others are missing (those with weight zero). Referring to a probabilistic model parameterized by a low-rank matrix X, where $A = X + Z$ and Z is white Gaussian noise, the weighted cost of X is equivalent to the log-likelihood of the observed variables.

This suggests an expectation-maximization procedure. In each **EM** update we would like to find a new parameter matrix maximizing the expected log-likelihood of a filled-in A, where missing values are filled in according to the distribution imposed by a current estimate of X. This maximum-likelihood parameter matrix is the (unweighted) low-rank approximation of the mean filled-in A, which is A with missing values filled in from X. To summarize: In the **E**expectation step values from the current estimate of X are filled in for the missing values in A, and in the **M**maximization step X is reestimated as a low-rank approximation of the filled-in A.

In order to extend this approach to a general weight matrix, consider a probabilistic system with several target matrices, $A_{(1)}, A_{(2)}, \ldots, A_{(N)}$, but a single low-rank parameter matrix X, where $A_{(r)} = X + Z_{(r)}$ and the random matrices $Z_{(r)}$ are independent white Gaussian noise, with fixed variance. When all target matrices are fully observed, the maximum likelihood setting for X is the low-rank approximation of the their average. Now, if some of the entries of some of the target matrices are not observed, we can use a similar **EM** procedure, where at the expectation step values from the current estimate of X are filled in for all missing entries in the target matrices, and in the maximization step X is updated to be a low-rank approximation of the mean of the filled-in target matrices.

To see how to use the above procedure to solve weighted low-rank approximation problems, consider systems with weights limited to $W_{ia} = \frac{w_{ia}}{N}$ with integer $w_{ia} \in \{0, 1, \ldots, N\}$. Such a low-rank approximation problem can be transformed to a missing value problem in the form above by "observing" the value A_{ia} in w_{ia} of the target matrices (for each entry i, a), and leaving the entry as missing in the rest of the target matrices. The **EM** update then becomes:

$$X^{(t+1)} = \text{unweighted-low-rank-approx}\left(W \otimes A + (\mathbf{1} - W) \otimes X^{(t)}\right) \qquad (6)$$

Note that this procedure is independent of N. For any weight matrix (scaled to weights

between zero and one) the procedure in equation (6) can thus be seen as an expectation-maximization procedure. This provides for a very simple method for finding weighted low-rank approximations.

2.3 Reconstruction experiments

Since the unweighted or simple low rank approximation problem permits a closed form solution, one might be tempted to use such a solution even in the presence of non-uniform weights (i.e., ignore the weights). We demonstrate here that this procedure would accompany a substantial loss of reconstruction accuracy as compared to the EM algorithm designed for the weighted problem.

To this end, we generated 1000×30 low rank matrices combined with Gaussian noise models to yield the observed (target) matrices. For each matrix entry, the noise variance σ_{ia}^2 was chosen uniformly between zero and some maximal noise level. The planted matrix was subsequently reconstructed using weighted low-rank approximation (EM with weights $W_{ia} = 1/\sigma 2_{ia}$), and unweighted low-rank approximation (SVD). The quality of reconstruction was assessed by an unweighted squared distance from the "planted" matrix. SVD reconstruction is heavily affected by matrix entries with high variance, orders of magnitude larger than most entries in the matrix. To further aid the SVD reconstruction, target values associated with very small weights (very high noise variance) were set to zero.

Figure 2 shows the quality of reconstruction attained by the two approaches as a function of the signal (variance of planted low-rank matrix) to noise (overall variance of the error) ratio. The performance of the EM algorithm incorporating the weights is clearly superior albeit comes at a cost of guaranteeing only a locally optimal solution.

The performance of the EM algorithm is tied to initialization. When initialized to $X = 0$, the EM algorithm typically converged after about a dozen iterations, always to what seemed to be the global minimum (lower weighted-distance to the data than the planted solution or any of the "zeroed" unweighted solutions, and the same minimum to which all gradient-based optimizations converged). However, when initialized to other starting points (e.g. to the unweighted low-rank approximation), in many cases EM converged to a much worse local minimum.

3 Low-rank logistic regression

In certain situations we might like to capture a binary data matrix $y \in \{-1, +1\}^{n \times d}$ with a low-rank model. A natural choice in this case is a logistic model parameterized by a low-rank matrix $X \in \Re^{n \times d}$, such that $\Pr(Y_{ia} = +1 | X_{ia}) = g(X_{ia})$ independently for each i, a, where g is the logistic function $g(x) = \frac{1}{1+e^{-x}}$. One then seeks a low-rank matrix X maximizing the likelihood $\Pr(Y = y|X)$.

Using a weighted low-rank approximation, we can fit a low-rank matrix X minimizing a quadratic loss from the target. In order to fit a non-quadratic loss such as a logistic loss, $\text{Loss}(y_{ia}, X_{ia}) = \log g(y_{ia} X_{ia})$, we use a quadratic approximation to the loss.

Consider the second-order Taylor expansion of $\log g(yx)$ about $\tilde{x}$:

$$\begin{aligned} \log g(yx) &\approx \log g(y\tilde{x}) + yg(-y\tilde{x})(x - \tilde{x}) - \tfrac{g(y\tilde{x})g(-y\tilde{x})}{2}(x - \tilde{x})^2 \\ &\approx -\tfrac{g(y\tilde{x})g(-y\tilde{x})}{2}\left(x - \left(\tilde{x} + \tfrac{y}{g(y\tilde{x})}\right)\right)^2 + \log g(y\tilde{x}) + \tfrac{g(-y\tilde{x})}{2g(y\tilde{x})} \end{aligned} \quad (7)$$

The log-likelihood of a low-rank parameter matrix X can then be approximated as:

$$\log \Pr(y|X) \approx -\sum_{ia} \tfrac{g(y_{ia}\tilde{X}_{ia})g(-y_{ia}\tilde{X}_{ia})}{2}\left(X_{ia} - \left(\tilde{X}_{ia} + \tfrac{y_{ia}}{g(y_{ia}\tilde{X}_{ia})}\right)\right)^2 + \text{Const} \quad (8)$$

Maximizing (8) is a weighted low-rank approximation problem. Note that for each entry (i, a), we use a second-order expansion about a *different* point $\tilde{X}_{ia}$. The closer the origin $\tilde{X}_{ia}$ is to X_{ia}, the better the approximation. This suggests an iterative approach, where in each iteration we find a parameter matrix X using an approximation of the log-likelihood about the parameter matrix found in the previous iteration.

For the Taylor expansion, the improvement of the approximation is not always monotonic. This might cause the method outlined above not to converge. In order to provide for a more robust method, we use the following variational bound on the logistic [6]:

$$\begin{aligned}\log g(yx) &\geq \log g(y\tilde{x}) + \tfrac{yx-y\tilde{x}}{2} - \tfrac{\tanh(\tilde{x}/2)}{4\tilde{x}}\left(x^2 - \tilde{x}^2\right) \\ &= -\tfrac{1}{4}\tfrac{\tanh(\tilde{x}/2)}{\tilde{x}}\left(x - \tfrac{y\tilde{x}}{\tanh(\tilde{x}/2)}\right) + \text{Const} \qquad (9)\end{aligned}$$

$$\log \Pr(y|X) \geq -\tfrac{1}{4}\sum_{ia} \tfrac{\tanh(\tilde{X}_{ia}/2)}{\tilde{X}_{ia}}\left(X_{ia} - \tfrac{y_{ia}\tilde{X}_{ia}}{\tanh(\tilde{X}_{ia}/2)}\right) + \text{Const} \qquad (10)$$

with equality if and only if $X = \tilde{X}$. This bound suggests an iterative update of the parameter matrix $X^{(t)}$ by seeking a low-rank approximation $X^{(t+1)}$ for the following target and weight matrices:

$$A_{ia}^{(t+1)} = y_{ia}/W_{ia}^{(t+1)}, \qquad W_{ia}^{(t+1)} = \tanh(X_{ia}^{(t)}/2)/X_{ia}^{(t)}. \qquad (11)$$

Fortunately, we do not need to confront the severe problems associated with nesting iterative optimization methods. In order to increase the likelihood of our logistic model, we do not need to find a low-rank matrix minimizing the objective specified by (11), just one improving it. Any low-rank matrix $X^{(t+1)}$ with a lower objective value than $X^{(t)}$ (with respect to $A^{(t+1)}$ and $W^{(t+1)}$) is guaranteed to have a higher likelihood: A lower objective corresponds to a higher upper bound in (10), and since the bound is tight for $X^{(t)}$, the log-likelihood of $X^{(t+1)}$ must be higher than the log-likelihood of $X^{(t)}$. Moreover, if the likelihood of $X^{(t)}$ is not already maximal, there are guaranteed to be matrices with lower objective values.

Therefore, we can mix weighted low-rank approximation iterations and logistic bound update iterations, while still ensuring convergence. In many applications we would also want to associate external weights with each entry in the matrix, or equivalently accommodate missing, or multiple, samples. This can easily be done by multiplying the weights in (11) by the external weights.

Note that the target and weight matrices corresponding to the Taylor approximation and those corresponding to the variational bound are different: The variational target is always closer to the current value of X, and the weights are more subtle. This ensures the guaranteed convergence (as discussed above), but the price we pay is a lower convergence rate. The Taylor approximation provides for faster convergence in most cases, but is not guaranteed to converge.

4 Low-rank approximation with a mixture noise model

Weighted Frobenius distance low-rank approximation corresponds to finding a maximum-likelihood low-rank matrix X, where we assume that our observations are generated by $X + Z$, where Z is i.i.d. Gaussian noise. Here we tackle the problem in which Z_{ia} are still i.i.d., but now they are generated from some alternate distribution $\Pr_Z$, specified as a mixture of Gaussians $\Pr_Z(z_{ia}) = \sum_{c=1}^{m} p_r \left(2\pi\sigma_r^2\right)^{1/2} \exp\left((z_{ia} - \mu_r)^2/(2\sigma_r^2)\right)$. For an observations matrix y, we would like to find the low-rank matrix X maximizing the likelihood $\Pr(y = X + Z)$. To do so, we introduce latent variables C_{ia} specifying the

mixture component of the noise at (i, a). The problem can then be solved using EM. In the **M**aximization step we maximize:

$$\mathbf{E}_C\left[\log \Pr\left(y|X,C\right)\right] = -\sum_{ia} \mathbf{E}_{C_{ia}}\left[\frac{1}{2}\log 2\pi\sigma^2_{C_{ia}} + \frac{1}{2\sigma^2_{C_{ia}}}\left(\left(X_{ia}-y_{ia}\right)-\mu_{C_{ia}}\right)^2\right]$$

$$= -\sum_{ia}\sum_{c}\frac{\Pr(C_{ia}=c)}{2\sigma_c^2}\left(X_{ia}-\left(y_{ia}+\mu_c\right)\right)^2 + \text{Const}, = -\frac{1}{2}\sum_{ia} W_{ia}\left(X_{ia}-A_{ia}\right)^2 + \text{Const}$$

$$\text{where:} \qquad W_{ia} = \sum_c \frac{\Pr(C_{ia}=c)}{\sigma_c^2}, \qquad A_{ia} = y_{ia} + \sum_c \frac{\Pr(C_{ia}=c)\mu_c}{\sigma_c^2}/W_{ia}, \qquad (12)$$

which is a weighted low-rank approximation problem. The posteriors $\Pr\left(C_{ia}=c\right)$ are easily computed in the **E**xpectation step using the current low-rank parameter matrix X. As with the low-rank logistic regression, we can interleave the weight and target matrix updates with the weighted low-rank approximation iterations.

This can further be extended to the situation in which the error model is unknown, and we would like to search not only over the underlying low-rank structure X, but also over the appropriate error model for Z. To do so, we use two separate **M**aximization rounds, one for X and one for the noise-model parameters.

5 Conclusion

We have provided a simple and efficient algorithm for solving weighted low rank approximation problems. These problems are important in their own right and also appear as subroutines in solving a class of more general low rank formulations. Some of these were already outlined in this paper. Similar approaches can be used for other convex loss functions with a bounded Hessian. Further extensions of the methods include formulating and solving semi-supervised versions of the estimation problem, where the noise model appears as a nuisance parameter.

We are continuing to study the weighted low-rank approximation problem, understanding the sensitivity of EM and gradient methods to the weight distribution, and tracking local minima as the weights change (morphing weights as a function of iteration might also aid in convergence). We are applying these techniques to problems involving factor-gene binding arrays and robust collaborative filtering.

References

[1] Michal Irani and P Anandan. Factorization with uncertainty. In *European Conference on Computer Vision*, June 2000.

[2] Joshua B. Tenenbaum and William T. Freeman. Separating style and content with bilinear models. *Neural Computation*, 12(6):1247–1283, 2000.

[3] Dale Shpak. A weighted-least-squares matrix decomposition method with application to the design of two-dimensional digital filters. In *IEEE Midwest Symposium Circuits Systems*, pages 1070–1073, Calgary, AB, Canada, August 1990.

[4] W.-S. Lu, S.-C. Pei, and P.-H. Wang. Weighted low-rank approximation of general complex matrices and its application in the design of 2-D digital filters. *IEEE Transactions on Circuits and Systems—I*, 44(7):650–655, July 1997.

[5] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, volume 13, pages 556–562, 2001.

[6] Tommi Jaakkola and Michael Jordan. Bayesian parameter estimation via variational methods. *Statistics and Computing*, 10:25–37, 2000.