Probabilistic sparse matrix factorization with an application to discovering gene functions in mouse mRNA expression data

Delbert Dueck, Quaid Morris, and Brendan Frey
Department of Electrical and Computer Engineering
University of Toronto
Toronto, Ontario, Canada M5S 3G4
{delbert,quaid,frey}@psi.toronto.edu

Abstract

We address the problem of sparse matrix factorization using a generative model. Our algorithm, probabilistic sparse matrix factorization (PSMF), is a probabilistic extension of a previous hard-decision algorithm for this problem (Srebro and Jaakkola 2001). PSMF allows for varying levels of sensor noise in the data, uncertainty in the hidden prototypes used to explain the data, and uncertainty as to which prototypes are selected to explain each data vector. We present experimental results demonstrating that our method can better recover functionally relevant clusterings in mRNA expression data than standard clustering techniques, and we show that by computing probabilities instead of point estimates, our method avoids converging to poor solutions.

1 Introduction

Many kinds of data can be viewed as consisting of a set of vectors, each of which is a noisy combination of a small number of noisy prototype vectors. Moreover, these prototype vectors may correspond to different hidden variables that play a meaningful role in determining the measured data. For example, a gene’s expression is influenced by the presence of transcription factor proteins, and two genes may be activated by overlapping sets of transcription factors. Consequently, the activity of each gene can be explained by the activities of a small number of transcription factors. This task can be viewed as the problem of factorizing a data matrix, while taking into account constraints reflecting structural knowledge of the problem and probabilistic relationships between variables that are induced by known uncertainties in the problem. A simple example of a technique for finding such a factorization is principal components analysis (PCA). In this paper, we study algorithms for finding matrix factorizations, but with a specific focus on sparse factorizations, and on properly accounting for uncertainties while computing the factorization.

Our interest in algorithms for “probabilistic sparse matrix factorization” (PSMF) is motivated by a problem identified during our collaborations with researchers working in the area of molecular biology. Advances in bio-molecular sensing technologies have enabled researchers to probe the transcriptional state of cells and quantify the abundance of specific nucleotide sequences including mRNAs (which code for proteins). These technologies, called DNA microarrays, gene expression arrays, or simply expression arrays (Schena et al. 1995) can be applied to a variety of problems, ranging from genome sequencing to detecting the activation of a specific gene in a tissue sample (e.g., for the purpose of detecting disease or monitoring drug response). Although researchers now have digital representations of the actors in cellular processes and the ability to probe these actors, it is not yet known how the information is digitally encoded and how the digital components interact to regulate cellular activities.

An expression array is a $1'' \times 2''$ biology slide containing tens of thousands of probes, each of which detects a specific nucleic acid sequence. These arrays can be used to measure the mRNA abundance levels by first extracting mRNA from a biological sample of interest, tagging the sequences with a fluorescent molecule, and placing the tagged sequences on the array. mRNA sequences from the tissue that match each probe will tend to hybridize (stick to the probe), so when the slide is scanned by a laser, the level of fluorescence at each probe indicates the amount of the corresponding mRNA sequence in the tissue sample. By varying the type of sample (e.g., heart versus brain) or the conditions under which the sample is taken (e.g., healthy versus diseased), multiple arrays can be used to construct a vector of expression levels for each gene, where
the elements of the vector correspond to the different experimental conditions. Such a vector is called an "expression profile".

By viewing expression profiles as points in a vector space, researchers have been able to use well-known vector-space data analysis techniques to identify new patterns of biological significance, and quickly make predictions based on previous studies that required a large amount of time and resources. In particular, the construction of these profiles has enabled the large-scale prediction of gene function for genes with unknown function, based on genes with known function (Hughes et al. 2000). Because many biological functions depend upon the coordinated expression of multiple genes, similarity of expression profile often implies similarity of function (Marcotte et al. 1999). This relationship has been exploited to predict the function of uncharacterized genes by various researchers (see, e.g., Brown et al. 2000). These schemes make use of annotation databases, which assign characterized genes to or one or more predefined functional categories.

However, noise in the expression measurements has limited the predictive accuracy of these algorithms, especially in those categories containing a small number of genes. These "small" categories can be of greater interest because less is typically known about them; furthermore, they often make specific (and thus more easily confirmed) predictions about gene function. Here, we introduce an unsupervised technique that jointly denoises expression profile data and computes a sparse matrix factorization. Our technique models the underlying causes of the expression data in terms of a small number of hidden factors. The representation of the expression profile in terms of these hidden factors is less noisy; here we explore whether this representation is more amenable to functional prediction.

Our technique explicitly maximizes a lower bound on the log-likelihood of the data under a probability model. The sparse encoding found by our method can be used for a variety of tasks, including functional prediction and data visualization. We report p-values, which show that our technique predicts functional categories with greater statistical significance than a standard method, hierarchical agglomerative clustering. Also, we show that our algorithm, which computes probabilities rather than making hard decisions, obtains a higher data log-likelihood than the version of the algorithm that makes hard decisions.

2 Methods for matrix factorization

One approach to analyzing data vectors lying in a low-dimensional linear subspace is to stack them to form a data matrix, \( \mathbf{X} \), and then find a low-rank matrix factorization of the data matrix. Given \( \mathbf{X} \in \mathcal{R}^{G \times T} \), matrix factorization techniques find a \( \mathbf{Y} \in \mathcal{R}^{G \times C} \) and a \( \mathbf{Z} \in \mathcal{R}^{C \times T} \) such that \( \mathbf{X} \approx \mathbf{Y} \cdot \mathbf{Z} \).

Interpreting the rows of \( \mathbf{X} \) as input vectors \( \{x_g\}_{g=1}^G \), the rows of \( \mathbf{Z} \) (i.e. \( \{z_c\}_{c=1}^C \)) can be viewed as vectors that span the \( C \)-dimensional linear subspace, in which case the \( g \)th row of \( \mathbf{Y} \) contains the coefficients \( \{y_{gc}\}_{c=1}^C \) that combine these vectors to explain the \( g \)th row of \( \mathbf{X} \).

A variety of techniques have been proposed for finding matrix factorizations, including non-probabilistic techniques such as principal components analysis, independent components analysis (Bell and Sejnowski 1995), and network component analysis (Liao et al 2003), and also probabilistic techniques that account for noise, such as factor analysis. In many situations, we anticipate that the data consists of additive combinations of positive sources. In these cases, non-probabilistic techniques have been proposed for finding factorizations under non-negativity constraints (Lee and Seung 1999, 2001).

Global gene expression is regulated by a small set of “transcription factors” (TFs) — proteins which bind to the promoter region associated with each gene. Typically, each individual gene is influenced by only a small subset of these TFs. Inspired by this, we model the gene expression vector as an additive, weighted combination of a small number of prototype vectors — each prototype representing the influence of a different factor (or factors). Although each prototype may represent the aggregate effect of a number of TFs that always act in concert, to simplify presentation, we describe each prototype as a transcription factor, and the prototype profile — which represents the effect of the TF on gene expression — as the TF profile.

This type of problem was called “sparse matrix factorization” in (Srebro and Jaakkola 2001), and is related to independent component analysis (ICA). In their model, Srebro and Jaakkola augment the \( \mathbf{X} \approx \mathbf{Y} \cdot \mathbf{Z} \) matrix factorization setup with the sparseness structure constraint that each row of \( \mathbf{Y} \) has at most \( N \) non-zero entries\(^1\). They then describe an iterative algorithm for finding a sparse matrix factorization that makes hard decisions at each step.

On the other hand, our method finds such a factorization while accounting for uncertainties due to (1) different levels of noise in the data, (2) different levels of noise in the factors used to explain the data, and (3) uncertainty as to which hidden prototypes are selected to explain each input vector.

\(^1\)When \( N = 1 \), this scheme degenerates to clustering with arbitrary data vector scaling; \( N = C \) yields ordinary low-rank approximation.
3 Probabilistic sparse matrix factorization (PSMF)

Let $\mathbf{X}$ be the matrix of gene expression data such that each row corresponds to each of $G$ genes and columns to each of $T$ tissues (i.e. entry $x_{gt}$ represents the amount by which gene $g$ is expressed in cells of tissue type $t$). We denote the collection of unobserved transcription factor profiles as a matrix, $\mathbf{Z}$, with rows corresponding to each of $C$ factors and $T$ columns corresponding to tissues, as before.

We model each gene expression profile, $x_g$, as a linear combination of a small number ($r_g$) of these transcription factor profiles, $z_{sc}$, plus noise:

$$x_g = \sum_{s=1}^{r_g} y_{gssn} z_{sc} + \text{noise} \quad (1)$$

The transcription factor profiles contributing to the $g^{th}$ gene expression profile are indexed by $\{s_1, s_2, \ldots, s_{r_g}\}$, with corresponding weights $\{y_{gss1}, y_{gss2}, \ldots, y_{gssr_g}\}$. This is identical to the $\mathbf{X} \approx \mathbf{Y} \cdot \mathbf{Z}$ matrix factorization with $\{\mathbf{S}, \mathbf{r}\}$ representing the sparseness structure constraint. We account for varying levels of noise in the observed data by assuming the presence of gene-specific isotropic Gaussian sensor noise with variance $\psi_g^2$ so the likelihood of $x_g$ is as follows:

$$P(x_g|y_g, z_g, r_g, \psi_g^2) = \mathcal{N}(x_g; \sum_{s=1}^{r_g} y_{gssn} z_{sc}, \psi_g^2 I) \quad (2)$$

We complete the model with prior assumptions that $X$ing the sparseness structure constraint. We account for noise in the sparseness constraint. We account for varying levels of noise in the observed data by assuming the presence of gene-specific isotropic Gaussian sensor noise with variance $\psi_g^2$ so the likelihood of $x_g$ is as follows:

$$P(x_g|y_g, z_g, r_g, \psi_g^2) = \mathcal{N}(x_g; \sum_{s=1}^{r_g} y_{gssn} z_{sc}, \psi_g^2 I) \quad (2)$$

We complete the model with prior assumptions that the factor indices ($s_{gn}$) are normally distributed and that the factor indices ($s_{gn}$) are uniformly distributed. The number of causes, $r_g$, contributing to each gene’s profile is multinomially distributed such that $P(r_g=n) = \nu_n$, where $\nu$ is a user-specified $N$-vector. We make no assumptions about $Y$ beyond the sparseness constraint, so $P(Y) \propto 1$.

Multiplying these priors by (2) forms the following joint distribution:

$$P(X, Y, Z, S, r|\Psi) \propto \prod_{g=1}^{G} \prod_{c=1}^{C} P(x_g|y_g, z_g, r_g, \psi_g^2) \cdot P(y_g) \cdot P(z_{sc}) \cdot P(s_{gn}) \cdot P(r_g) \quad (3)$$

3.1 Factorized Variational Inference

Exact inference with (3) is intractable so we utilize a factorized variational method (Jordan et al. 1998) and approximate the posterior distribution with a mean-field decomposition:

$$P(Y, Z, S, r|X, \Psi) \approx \prod_{g=1}^{G} \prod_{c=1}^{C} P(y_g) \cdot \prod_{c=1}^{C} P(z_{sc}) \cdot \prod_{g=1}^{G} \prod_{n=1}^{N} P(s_{gn}) \cdot \prod_{g=1}^{G} P(r_g) \quad (4)$$

We parameterize the Q-distribution as follows:

$$\lambda_{gc} \text{ is a point estimate of } y_{gc} \quad \text{but for } s_{gn} \in \mathcal{C}$$

$$Q(y_g) = \prod_{n=1}^{N} \delta (y_{gssn} - \lambda_{gn}) \cdot \prod_{c=1}^{C} \delta (y_{gc}) \cdot \prod_{c \notin \mathcal{C}} \delta (y_{gc}) \quad (5)$$

Using this approach, inference corresponds to bringing the Q-distribution as close as possible to the P-distribution by setting the variational parameters ($\lambda$, $\zeta$, $\phi$, $\sigma$, $\rho$) to minimize the relative entropy, $D(Q||P)$:

$$\min_{\lambda, \zeta, \phi, \sigma, \rho} \int Q(Y, Z, S, r) \cdot \log \frac{Q(Y, Z, S, r)}{P(Y, Z, S, r|X, \Psi)} \quad (5)$$

The constraints $\sum_{c=1}^{C} \sigma_{gn} = 1$, $\sum_{n=1}^{N} \rho_{gn} = 1$ become Lagrange multipliers in this optimization problem.

There is no closed-form expression for the posterior (denominator in (5)), but we can subtract $\log D(Q||P)$ inside the integral (it is independent of the variational parameters) to form the readily-minimized free energy, $F$:

$$F = D(Q||P) - \log P(X) \quad \text{or} \quad \int Q(Y, Z, S, r) \cdot \log \frac{Q(Y, Z, S, r)}{P(Y, Z, S, r|X, \Psi)}$$

$$\vdots$$

$$= \sum_{g=1}^{G} \sum_{n=1}^{N} \sum_{c=1}^{C} \left( \sigma_{gn} \cdot \log \frac{\sigma_{gn}}{1/C} \right)$$

$$+ \sum_{g=1}^{G} \sum_{n=1}^{N} \left( \rho_{gn} \cdot \log \frac{\rho_{gn}}{\nu_n} \right) + \frac{T}{2} \sum_{g=1}^{G} \log 2 \pi \psi_g^2$$

$$- \frac{T}{2} \sum_{c=1}^{C} \left( 1 + \log \phi_c^2 \right) + \frac{T}{2} \sum_{c=1}^{C} \left( \zeta_c + \phi_c^2 \right)$$

$$+ \frac{1}{2} \sum_{g=1}^{G} \sum_{t=1}^{T} \sum_{n=1}^{N} \sum_{c=1}^{C} \rho_{gn} \psi_g^2 \phi_c \sum_{c=1}^{C} \rho_{gn} \phi_c$$

$$\cdot \sum_{c=1}^{C} \sigma_{gn} \cdot \zeta_c$$

$$= \sum_{c=1}^{C} \sum_{g=1}^{G} \sum_{n=1}^{N} \left( \lambda_{gt} - \sum_{n'=1}^{n} \lambda_{gtn'} \cdot \zeta_{n'} \right)^2 + \sum_{n'=1}^{n} \lambda_{gtn'}^2 \phi_{n'}^2$$
The free energy can be minimized sequentially with respect to each variational parameter \((\lambda, \zeta, \phi, \sigma, \rho)\) by analytically finding zeros of the partial derivatives with respect to them. This coordinate descent represents the E-step in variational EM (Jordan et al. 1998) that alternates with a brief M-step, where the global sensor noise is fit by similarly solving \(\partial F/\partial \Psi = 0\). For a more detailed treatment of the algorithm and parameter update equations, see (Dueck and Frey 2004).

3.2 Non-negativity Constraints

Recalling the biological intuition behind the model, gene expression profiles across tissues are thought to consist of a linear combination of transcription factor profiles. Since gene expression data represents gene abundances, data matrix \(\mathbf{X}\) should only have non-negative entries. To ensure this, it is convenient to constrain the transcription factor profile vectors, \(\mathbf{Z}\), to also be non-negative.

Similarly, we wish to apply a non-negativity constraint to the factor strengths (weights), \(\mathbf{Y}\), to be compatible with the notion of allowing only additive combinations of factors; side-effects of this are described in (Lee and Seung 1999, 2001). This constraint could be enforced by introducing additional Lagrange multipliers to constrain \(\lambda\) and \(\zeta\) or by modifying the priors for \(\mathbf{Y}\) and \(\mathbf{Z}\) to disallow negative values, but this adversely affects the differentiability of \(F\). In practice, we simply zero-thresholded the parameters after each update, sidestepping the issue of further complicating the update equations. This heuristic has no noticeable impact on free energy minimization.

4 Experimental Results

To experimentally analyze the performance of our algorithm, we use the expression data from Zhang et al. in (Zhang, Morris et al. 2004). Each gene’s profile is a measurement of its expression in a set of 55 mouse tissues. Expression levels are rectified at the median expression level, so that the data set is entirely non-negative, consistent with the notion of light intensities also being non-negative.

The functional category labels for the genes with known biological function were taken from (Zhang, Morris et al. 2004). An example of a category label is “cell wall biosynthesis”, which indicates the gene expresses a protein that is involved in building the cell wall. These labels are derived from Gene Ontology Biological Process (GO-BP) category labels (Ashburner et al. 2000) assigned to genes by EBI and MGI.

Among the 22,709 genes in the Zhang et al. database, 9499 have annotations in one or more of 992 different GO categories. The category sizes range from 3 to 456 genes, with more than half the categories having fewer than 20 genes.

We present results for the 22,709 genes \(\times\) 55 tissues data set shown in Figure 1. The data matrix, \(\mathbf{X}\), is shown alongside the model’s approximation, \(\hat{\mathbf{X}}\), also expressed as \(\hat{\mathbf{X}} = \mathbf{Y} \cdot \mathbf{Z}\). A total of \(C = 50\) factors were used, and the user-specified prior on the number of factors \((r_g)\) explaining each expression profile was set to \(\nu = [0.55, 0.27, 0.18]\), making \(N = 3\).

Gene expression profiles (row vectors) are first sorted by ‘primary’ factor \((s_{g1})\); next, within each \(s_{g1}\) grouping, genes are sorted by ‘secondary’ factor \((s_{g2})\), and so on. This organization is easily visualized in the hierarchical diagonal structure of \(\mathbf{Y}\).

4.1 Unsupervised characterization of mRNA data

The overall objective for this research is to develop a model that captures the functionally relevant hidden factors that explain gene expression data. As such we can gauge success by measuring the similarity of the hidden factor assignments are for genes with similar functions.

We cluster genes on the basis of their TF assignments and use p-values calculated from the hypergeometric distribution to measure the enrichment of these clusters for genes with similar function (Tavazoie et al. 1999). For a given gene cluster of size \(M\) of which \(K\) elements had a given functional annotation, the hypergeometric p-value is the probability that a random set of genes of size \(M\) (selected without replacement from all genes in the dataset) would have \(K\) or more elements with given functional annotation. For \(N = 3\), we generate three different clustering of the genes: by ‘primary’ factor, ‘secondary’ factor, and ‘tertiary’ factor assignments. So, for example, genes \(g\) and \(g’\) are in the same ‘primary’ factor cluster if \(s_{g1} = s_{g’1}\). We label each cluster with the GO-BP category having the lowest hypergeometric p-value for the genes in that cluster, making this p-value the p-value for the entire cluster.

---

\(^2\)A uniform prior \(\nu\) (reflecting no knowledge about the distribution of \(r\)) would give equal preference to all values of a particular \(r_g\). For any given \(r_g < N\), a factor can almost always be found that, if present with infinitesimal weight \((y_{gN})\), will imperceptibly improve the cost function \((\mathcal{F})\), with the end result that almost all \(r_g\) would then equal \(N\). Weighting the prior towards lower values ensures that factors will only be included if they make a noteworthy difference. We only choose \(\nu_n \propto 1/n\), \(\forall n < N\) for simplicity.
For reference, we also compute the functional enrichment of gene clusters generated randomly and those generated using hierarchical agglomerative clustering (HAC), SMF, and ICA. To generate the HAC clusters, we used average linkage with Pearson’s correlation coefficient on the expression profiles. We generated SMF clusters in the same way as we generated the PSMF clusters. Note that unlike HAC, PSMF and SMF allow us to generate ‘primary’, ‘secondary’, and ‘tertiary’ clusters. To generate the ICA clusters, we used the FastICA package (Hyvärinen 1999) to recover 55 source vectors (i.e. factors) for X, each of length 22,709. We ignore the mixing matrix recovered by FastICA, and cluster the genes by the index of the largest absolute value among the corresponding 55-dimensional slice in the matrix of source vectors.

Histograms of these p-values, as well as those for hierarchical agglomerative clustering, are shown in Figure 2. We summarize these histograms in two ways: the proportion of clusters that are significantly enriched for at least one functional category at \( \alpha = 0.05 \) (after a Bonferroni correction) and the mean \( \log_{10}(p\text{-value}) \). Figure 3 shows these quantities for \( N = \{1, 2, 3\} \) over \( 0 < C \leq 100 \).

The plot shows that PSMF with \( N = 1 \) outperforms hierarchical agglomerative clustering and clustering by dominant sources with ICA. For \( N > 1 \), the functional enrichment of ‘primary’ clusters for PSMF and SMF remains constant but the ‘secondary’ and ‘tertiary’
clusters are also more functionally enriched than random. Note that for all values of \( N \), the PSMF clusters are more enriched than the corresponding SMF clusters.

Genes can perform multiple functional roles, and Figures 2 and 3 suggest that the 'secondary' and 'tertiary' factors may capture this additional level of functional organization. Figure 4 shows the probability distribution of 'secondary' factor assignment conditioned on various 'primary' factor assignments. Factors are labeled with their most significantly enriched GO-BP category and only factors with statistically significant functional enrichment (26 primary and 17 secondary) are shown.

Two possible types of functional organization can be seen in this figure. Consider the primary factor labelled “response to pest/pathogen/parasite” (circled horizontally) in Figure 4. Genes with this as their primary factor are primarily assigned into groups enriched for either “mitotic cell cycle”, “amino acid transport”, or “humoral immune response” by their secondary factor, possibly representing different strategies for generating an immune response. In contrast, consider the secondary factor enriched for “amino acid transport” (circled column in Figure 4). Genes having this secondary factor are primarily drawn from primary factors enriched for as “response to pest/pathogen/parasite” (from before), “cytokinesis”, “fatty acid metabolism”, and “pattern specification”. Perhaps representing different processes which require the transportation of amino acids.

4.2 Avoiding local minima by accounting for uncertainty

The benefit of using a soft-decision factorized variational method (PSMF) — as opposed to hard-decision iterated conditional modes (SMF) (Srebro and Jaakkola 2001, Besag 1986) — is also apparent in Figure 3. This is the case because SMF is too inflexible to properly allow factor groupings to evolve, as shown in the likelihood plots of Figure 5.

In our simulations, we observe that SMF appears to get trapped in a local log-likelihood maximum immediately after the first several iterations. An additional advantage of using probabilistic inference is that the free energy provides a tighter bound on the marginal
Figure 4: Conditional probability distributions of secondary factors (columns) given the primary factor (rows), i.e. only significant factors are shown. Probabilistic Sparse Matrix Factorization was performed with $C = 50$ factors and $N = 2$ ($\nu = 0.67, 0.33$). The GO categories with gene memberships most similar to each factor are labelled along each axis. The circled row and column are discussed in $\S$??.

Figure 5: Complete log-likelihood maximization of sparse matrix factorization (SMF) versus free energy minimization of probabilistic sparse matrix factorization (PSMF). The complete log-likelihood, i.e. $\ell_C$ (log of (3)) is calculated after each iteration of both ICM (which directly maximizes $\ell_C$) and variational EM using maximum-likelihood latent variable estimates. Log-likelihood of the data (not shown), which is greater than the complete log-likelihood.

5 Summary

Many kinds of data vectors can most naturally be explained as an additive combination of a selection of prototype vectors, which can be viewed as computational problem of finding a sparse matrix factorization. While most work on biological gene expression arrays has focused on clustering techniques and methods for dimensionality reduction, there is recent interest in performing these tasks jointly, which corresponds to sparse matrix factorization. Like (Srebro and Jaakkola 2001), our algorithm computes a sparse matrix factorization, but instead of making point estimates (hard decisions) for factor selections, our algorithm computes probability distributions. We find that this enables the algorithm to avoid local minima found by iterated conditional modes.

Compared to a standard technique used in the bioinformatics community for clustering gene expression profiles, hierarchical agglomerative clustering, our technique finds clusters that have higher statistical significance (lower p-values) in terms of enrichment of gene function categories. An additional advantage of our method over standard clustering techniques is that
the secondary and higher-order labels found for each expression vector can be used for more refined visualization and functional prediction. We are currently exploring this possibility.

Acknowledgements

The authors thank Tim Hughes from the Banting and Best Department of Medical Research at the University of Toronto for making the data available, and Sam Roweis, Tommi Jaakkola and Nati Srebro for helpful discussions. We also gratefully acknowledge the financial support of Natural Sciences and Engineering Research Council of Canada.

References


