Meta-clustering Gene Expression Data with Positive Tensor Factorizations

Liviu Badea [1] and **Doina Tilivea** [1]

1 INTRODUCTION AND MOTIVATION

Although clustering is probably the most frequently used tool for data mining gene expression data, existing clustering approaches face at least one of the following problems in this domain: a huge number of variables (genes) as compared to the number of samples, high noise levels, the inability to naturally deal with overlapping clusters and the difficulty in clustering genes and samples simultaneously. *Fuzzy k-means* or *Nonnegative Matrix Factorization* (*NMF*) [1] could be used to produce potentially overlapping clusters, but these approaches are affected by a significant problem: the *instability* of the resulting clusters w.r.t. the initialization of the algorithm. This is not surprising if we adopt a unifying view of clustering as a constrained optimization problem, since the fitness landscape of such a complex problem may involve many different local minima into which the algorithm may get caught when started off from different initial states. And although such an instability seems hard to avoid, we may be interested in the clusters that keep reappearing in the majority of the runs of the algorithm. Note that combining clustering results is more complicated than combining classifiers, as it involves solving an additional so-called *cluster correspondence* problem, which amounts to finding the best matches between clusters generated in different runs. The cluster correspondence problem itself could be solved by a suitable *meta-clustering algorithm*.

2 TWO-WAY METACLUSTERING WITH PTF

In this paper we make the simplifying assumption that the overlap of influences (biological processes) is *additive* $X_{sg} \approx \sum_c A_{sc} \cdot S_{cg}$ (1)

where X_{sg} is the expression level of gene g in data sample s, while the expression level of g in s due to biological process c is multiplicatively decomposable into the expression level A_{sc} of the biological process (cluster) c in sample s and the membership degree S_{cg} of gene g in c. Since expression levels and membership degrees cannot be negative: $A_{sc} \geq 0,\ S_{cg} \geq 0,$ our clustering problem (1) can be viewed as a n*onnegative factorization* and could be solved using Lee and Seung's seminal *Nonnegative Matrix Factorization* (*NMF*) algorithm [1]. Such a factorization can be viewed as a "soft" clustering algorithm allowing for *overlapping clusters*, since we may have several significant S_{cg} entries on a given column g of S (a gene g may "belong" to several clusters c).

Allowing for cluster overlap alleviates but does not completely eliminate the instability of clustering, since the NMF algorithm produces different factorizations (biclusters) $(A^{(i)},S^{(i)})$ for different initializations, so meta-clustering the resulting "soft" clusters might be needed to obtain a more stable set of clusters.

In this paper, we show that a generalization of NMF called Positive Tensor Factorization (PTF) [2] is precisely the tool needed for meta-clustering "soft", potentially overlapping *biclusters* obtained by NMF object-level clustering. This unified approach solves in an elegant manner both the clustering and the cluster correspondence problem. More precisely, we first run NMF as object-level clustering r times: $X \approx A^{(i)} \cdot S^{(i)}$. (To allow the comparison of membership degrees S_{cg} for different clusters c, we scale the rows of $S^{(i)}$ to unit norm by taking advantage of the scaling invariance of the factorization.) Next, we *meta-cluster* the resulting *biclusters* $(A^{(i)}, S^{(i)})$. This is in contrast with as far as we know all existing (*one-way*) meta-clustering approaches, which take only one dimension into account and fail whenever two clusters correspond to very similar sets of genes, while differing along the sample dimension. The following *Positive Tensor Factorization* (*PTF*) of the biclusters $(A^{(i)}, S^{(i)})$ represents a *two-way* meta-clustering:

$$A_{s(ic)} \cdot S_{(ic)g} \approx \sum_{k=1}^{n_c} \alpha_{(ic)k} \cdot \beta_{sk} \cdot \gamma_{kg} \qquad (2)$$

where k are metacluster indices. (To simplify the notation, we merged the indices i and c into a single index (ic).)

The columns β_k of β and the corresponding rows $\gamma_{k\cdot}$ of γ make up a *base set of bicluster prototypes* $\beta_k \cdot \gamma_{k\cdot}$ out of which all biclusters of all individual runs can be recomposed, while α encodes the *(bi)cluster-metacluster correspondence*. Instead of a perfect bicluster correspondence, we settle for a weaker one (2) in which the rows of α can contain several significant entries, so that all biclusters $A_c^{(i)} \cdot S_c^{(i)}$ are recovered as *combinations* of bicluster prototypes $\beta_k \cdot \gamma_{k\cdot}$. The nonnegativity constraints of PTF meta-clustering are essential for obtaining sparse factorizations. (Experimentally, the rows of α tend to contain typically one or only very few significant entries.) The factorization (2) can be computed using the following multiplicative update rules:

$$\alpha \leftarrow \alpha * \frac{(A^T \cdot \beta) * (S \cdot \gamma^T)}{\alpha \cdot [(\beta^T \cdot \beta) * (\gamma \cdot \gamma^T)]}$$

$$\beta \leftarrow \beta * \frac{A \cdot [\alpha * (S \cdot \gamma^T)]}{\beta \cdot [(\alpha^T \cdot \alpha) * (\gamma \cdot \gamma^T)]} \qquad \gamma \leftarrow \gamma * \frac{[\alpha * (A^T \cdot \beta)]^T \cdot S}{[(\alpha^T \cdot \alpha) * (\beta^T \cdot \beta)]^T \cdot \gamma}$$

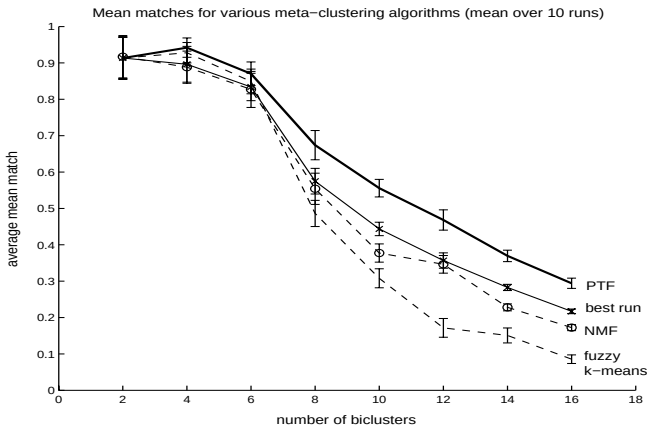
where '$*$' and '$--$' denote element-wise multiplication and division of matrices, while '$\cdot$' is ordinary matrix multiplication.

After convergence of the PTF update rules, we make the prototype gene clusters directly comparable to each other by normalizing the rows of γ to unit norm, as well as the columns of α such that $\sum_{i,c} \alpha_{(ic)k} = r$ and then run NMF initialized with (β, γ) to produce the final factorization $X \approx A \cdot S$.

[1] AI group, National Institute for Research and Development in Informatics, 8-10 Averescu Blvd. Bucharest, Romania. E-mail: *badea@ici.ro*.

3 EVALUATION ON SYNTHETIC DATA

We evaluated our algorithm on synthetic datasets that match as closely as possible real microarray data. Clusters were modelled using a hidden-variable model $X=A \cdot S+\varepsilon$, in which each hidden variable A_c corresponds to the cluster of genes influenced by A_c. We sampled the hidden variables from a $\log_2$-normal distribution with parameters μ=2, σ=0.5, while the influence coefficients S_{cg} between hidden and observable variables were sampled from a uniform distribution over the interval [1,2]. Finally, we added $\log_2$-normally distributed noise ε with parameters μ_{noise}=0, σ_{noise}=0.5. We chose problem dimensions of the order of our real-world application: $n_{samples}$=50, n_{genes}=100, number of genes (respectively samples) per cluster 30 (respectively 15). We compared 4 meta-clustering algorithms (fuzzy k-means, NMF, PTF and the best run) over 10 object-level NMF clustering runs. (Other object level clustering methods perform very poorly and are not shown here). Although all algorithms produce quite low relative errors (under 16%, except for fuzzy k-means which misbehaves for large numbers of clusters), they behave quite differently when it comes to recovering the original clusters. In a certain way, the match of the recovered clusters with the original ones is more important than the relative error. Defining the *match* between two sets of possibly *overlapping* clusters is nontrivial. For each cluster C_1 from clustering 1, we determine the single cluster C_2 from clustering 2 into which it is best included, i.e. the one with the largest $|C_1 \cap C_2|/|C_1|$. We proceed analogously for the clusters C_2 from clustering 2. Then, for each cluster C_1 (from clustering 1), we determine its match $|C_1 \cap C_2|/|C_1 \cup C_2|$ with the *union* C_2 of clusters from clustering 2, for which C_1 is the best including cluster (as determined in the previous step). Similarly, we determine matches for clusters C_2 from clustering 2. The average match of the two clusterings is then the mean of all these matches (for all C_1 and all C_2).

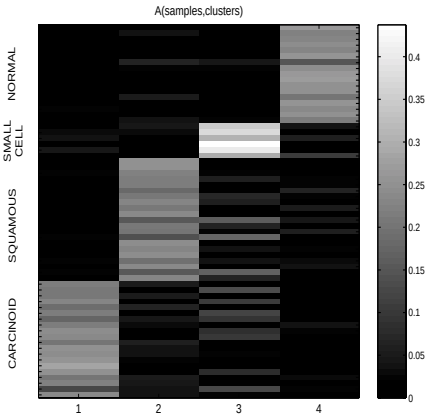


The Figure above shows that PTF consistently outperforms the other meta-clustering algorithms in terms of recovering the original clusters. Note that since clusters were generated randomly, their overlap increases with their number, so it is increasingly difficult for the meta-clustering algorithm to discern between them, leading to a decreasing match. We also observed an inverse correlation between bicluster overlap and matches (Pearson correlation coefficient -0.92). Among all *object-level* clustering algorithms tried (k-means, fuzzy k-means and NMF), only NMF behaves consistently well.

4 METACLUSTERING A LUNG CANCER GENE EXPRESSION DATASET

In the following we show that metaclustering is successful at biclustering a large lung cancer dataset from the Meyerson lab [3], containing 186 lung tumor samples (139 adenocarcinomas, 21 squamous cell lung carcinomas, 6 small cell lung cancers, 20 pulmonary carcinoids) and 17 normal lung samples. For testing our metaclustering algorithm, we first selected a subset of genes (251) that are differentially expressed between the classes (using a SNR measure). More precisely, we selected the genes with an average expression level over 100 and $|SNR| > 2$ for at least one of the classes. Since adenocarcinoma subclasses are poorly understood at the molecular level, we discarded the adeno samples from the dataset and used the histological classification of samples provided in the supplementary material to the original paper [3] as a gold standard for the evaluation of the biclustering results. To eliminate the bias towards genes with high expression values, all genes were scaled to equal norms. Since nonnegative factorizations like NMF cannot directly account for gene down-regulation, we extended the gene expression matrix with new "down-regulated genes" $g' = \text{pos}(mean(g_{normal}) - g)$ associated to the original genes g, where $mean(g_{normal})$ is the average of the gene over the *normal* samples and pos(·) is the step function. We then used our PTF metaclustering algorithm to factorize the extended gene expression matrix into 4 clusters with 20 NMF runs.

The algorithm recovered the sample clusters with high accuracy, as can be seen in the following Figure. Note that the overlap between the small cell and carcinoid sample clusters (columns 3 and 1 of A in the Figure) has a biological interpretation: both contain samples of tumors of neuroendocrine type. The low mixing coefficients indicate however that carcinoids are highly divergent from the malignant small cell tumors. We also looked in detail at some known marker genes. For example, the known small cell marker achaete scute 1 is *specific* to the small cell cluster, while keratin 5 is specific to the squamous cluster. On the other hand, known proliferative markers like *PCNA* (proliferating cell nuclear antigen), *MCM2* and *MCM6* are *common* to the small cell *and* squamous clusters, as expected. Overall, our metaclustering algorithm proved quite robust at rediscovering the known histological classification of the various lung cancer types in the Meyerson dataset.



REFERENCES

1. Lee D.D., H.S. Seung. Learning the parts of objects by non-negative matrix factorization. Nature, vol. 401, no. 6755, pp. 788-791, 1999.
2. Welling M., Weber M. Positive tensor factorization. Pattern Recognition Letters 22(12): 1255-1261 (2001).
3. Bhattacharjee et al. Classification of human lung carcinomas by mRNA expression profiling… PNAS Nov. 20; 98(24):13790-5, 2001.