# Expression profiles, clustering, and latent processes

If we have multiple array measurements concerning cell populations under different treatments relative to control, we can put together an expression profile for each gene: it's expression across the different treatments. Such profiles are useful in finding genes that behave similarly across the different experiments, presumably because they participate in the same processes that are activated (or suppressed) due to the treatments. We can also identify treatments that lead to similar expression responses, i.e., have similar consequences as far as transcriptional regulation is concerned.

There are many difficulties associated with this type of cluster analysis. Since biological processes are not independent of each other, many genes participate in multiple different processes. Each gene therefore should be assigned to multiple clusters whenever clusters are identified with processes. We also shouldn't necessarily expect to find gene profiles that look the same over all the experiments. The similarities would be restricted to those experiments that tap into the processes common to both genes. The available experiments may exercise only a fraction of the underlying processes. Thus the profiles of two genes might look the same because we have not carried out the experiments where they would differ. Finally, the cell populations are not uniform but may involve different cell types. Genes with similar aggregate profiles, averaged over the cell types, may look the same even though they differ substantially for each cell type.

## Clustering

For simplicity we will pay attention only to the log-ratio measurements from each array experiment, omitting the fact that it would be better to look at the actual intensity measurements from the two channels (see lecture 6). Let $x_{it}$ denote the log-expression ratio for gene $i$ in experiment $t$. We assume that there are $n$ experiments (on the order of tens) and $m$ genes (in the thousands). The available expression data can be put into a matrix form

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}$$

where each row represents a gene profile and each column defines a treatment/tissue/experiment profile. We use a special notation for gene profiles $g_i = [x_{i1}, \ldots, x_{in}]^T$, cast here as column vectors. Our goal here is to group together (cluster) gene profiles so as to capture genes that participate in the same biological processes.

**Hierarchical clustering**

Perhaps the simplest approach to clustering is hierarchical agglomerative clustering. Each profile initially represents a separate singleton cluster. The algorithm successively merges two most similar clusters into a larger one. The resulting "clustering" is a binary tree where each cluster appears as a node in the tree, the gene profiles lie at the bottom as leaves, and the topmost node corresponds to a single cluster containing all the profiles. Each non-singleton cluster has two "children" which are the clusters that were merged to create the larger one. Such a tree does not really cluster the data as it doesn't commit to any particular set of clusters (level in the tree); the task of deciding which clusters are real is left for the user.

The hierarchical clustering algorithm depends critically on the definition of similarity or distance between two clusters (profiles). The similarity of two clusters is typically defined on the basis of pairwise similarities of profiles they contain; for example, as average similarity between profiles from opposing clusters.

Gene profiles are often compared in terms of sample correlation: for any two gene profiles $g_k$ and $g_l$ it is defined as

$$\mathrm{Corr}(g_k, g_l) = \frac{1}{n}\sum_{i=1}^{n} \frac{(x_{ki} - \hat{\mu}_k)}{\hat{\sigma}_k}\frac{(x_{li} - \hat{\mu}_l)}{\hat{\sigma}_l}$$

where, for example,

$$\hat{\mu}_k = \frac{1}{n}\sum_{i=1}^{n} x_{ki}$$
$$\hat{\sigma}_k^2 = \frac{1}{n}\sum_{i=1}^{n}(x_{ki} - \hat{\mu}_k)^2$$

are the sample mean and the sample variance of profile $g_k$. We can interpret the sample correlation as the cosine of the angle between two normalized profiles (profiles with zero mean and unit variance). The sample correlation always lies in the interval $[-1, 1]$; the correlation between any two identical profiles is 1 and the correlation between perfectly opposing profiles (one is the negative of the other) is $-1$.

The sample correlation characterizes the extent to which the profiles are linearly dependent. In other words, how well we can predict the values in one profile from the other using least squares linear regression. Any non-linear dependences are not manifested through correlation. So, for example, it is possible that two profiles with zero correlation are

perfectly mutually predictable (one is a function of the other). On the other hand, if the profiles have nothing to do with each other, the sample correlation will be close to zero. It is not exactly zero because we can always find a slight linear dependence between finite profiles (the sample correlation would vanish with increasing $n$ if the profiles are independent).

One advantage of the sample correlation is that it can deem similar two profiles with widely different dynamic ranges and independent of the base level of expression. For example, a transcription factor is typically expressed at relatively low levels, while a gene that it regulates may be expressed at substantially higher levels. The two can end up in the same "tight" cluster provided that the expression level of the factor is linearly related to its ability to activate the downstream gene.

The hierarchical clustering algorithm with correlation as the similarity measure nevertheless fails to overcome many of the difficulties discussed above. For example, only a subset of the experiments may be relevant for specific biological processes. If we evaluate the sample correlation based on all the available experiments, we not only loose the ability to identify genes participating in such processes but also heavily bias the comparison towards processes relevant to the most frequent type of experiment.

**Mixture models and clustering**

We can also approach clustering in a model based manner, where our assumptions about the problem structure are more explicit. We begin by casting the clustering problem as a problem of estimating a (simple) mixture model. Suppose we look for $k$ underlying (disjoint) clusters based on the gene profiles. For simplicity we assume that the profiles within each cluster, say cluster $j$, can be modeled as samples from a normal distribution specific to the cluster:

$$p(g_i|\mu_j, \sigma_j^2) = N(g_i;\, \mu_j, \sigma_j^2) = \prod_{t=1}^{n} N(x_{it};\, \mu_{jt}, \sigma_j^2)$$

where $\mu_j = [\mu_{j1}, \ldots, \mu_{jn}]^T$ represents the mean profile for the cluster and $\sigma_j^2$ captures the overall deviation from the mean (same for all experiments). These simple cluster models can be combined into an overall mixture model:

$$p(g_i|\theta) = \sum_{j=1}^{k} p_k p(g_i|\mu_j, \sigma_j^2)$$

where $p_1, \ldots, p_k$ specify the prior probabilities of each cluster (the fraction of all profiles belonging to a particular cluster). We use $\theta$ as a shorthand for all the parameters in the

mixture model: prior probabilities $p_1, \ldots, p_k$, mean profiles (vectors) $\mu_1, \ldots, \mu_k$, and the cluster variances (one number per cluster) $\sigma_1^2, \ldots, \sigma_k^2$.

The structure of this mixture model comes across clearly when we imagine drawing samples of gene profiles. We first select a cluster according to the prior probabilities $p_1, \ldots, p_k$. A sample gene profile is subsequently drawn from the normal distribution corresponding to the selected cluster. The same gene profile can in principle arise as a sample from two different clusters (whose normal distributions overlap). So the cluster assignments of observed profiles are in general somewhat uncertain and need to be carried out probabilistically (in terms of posterior probabilities). The model nevertheless assumes that each gene belongs to a single cluster, we are just not certain which one.

Mixture models, as before, can be estimated iteratively via the EM-algorithm. In the E-step we fix $\theta$ (the current mixture parameters) and evaluate the posterior assignments of genes to the $k$ possible clusters:

$$P(j|i) = P(j|g_i, \theta) = \frac{p_j \, p(g_i|\mu_j, \sigma_j^2)}{p(g_i|\theta)}, \quad j = 1, \ldots, k, \quad i = 1, \ldots, m$$

where $\sum_{j=1}^{k} P(j|i) = 1$ for all genes $i$. In the M-step we fix the posterior assignments $P(j|i)$ (no longer tied to $\theta$), and separately estimate the normal distributions associated with each cluster from weighted gene profiles. For example, the new mean profile for cluster 1 is the weighted mean of the gene profiles:

$$\mu_{1t} = \frac{\sum_{i=1}^{m} P(j=1|i) \, x_{it}}{\sum_{i=1}^{m} P(j=1|i)}, \quad t = 1, \ldots, n$$

where the observed log-ratios are weighted by $P(j = 1|i)$ (with normalization). In other words, each cluster updates its distribution based on gene profiles in proportion to inferred responsibility. The EM-algorithm converges to a fixed point where the cluster models are consistent (in the sense of the derived parameter updates) with the set of genes they are supposed to model. The solution is not unique, however. The estimation problem is complicated by the fact that the cluster models and the posterior assignments are dependent on each other. This dependence is manifested in the successive steps of the EM algorithm.

One unsatisfying aspect of the mixture model approach is that the choice of $k$ or the number of clusters has to be set in the beginning. It is certainly possible to simply run the estimation algorithm multiple times, once for each reasonable value of $k$, and decide the correct $k$ with the help of a "model selection" criterion such as BIC (Bayesian Information Criterion). The difficulty with this approach is that for larger values of $k$ the estimation problem itself becomes more prone to getting stuck in sub-optimal solutions (there are more

of them). It is, however, possible to estimate mixture models without deciding a priori how many clusters we should find (Dirichlet process mixtures).

### Mixture models and partial profiles

Here we consider the more realistic case where only a subset of the experiments may be relevant for defining a cluster. We can incorporate this into the mixture model by modifying the normal distributions associated with each cluster. Specifically, we turn the cluster models into mixture models in a manner that each observation (experiment) can be either captured by a distribution specific to the cluster or by a generic distribution. More formally:

$$p(g_i|\mu_j, \sigma_j^2) = \prod_{t=1}^{n} \left[ p_{j0}\, N(x_{it};\, \mu_{jt}, \sigma_j^2) + (1 - p_{j0})\, N(x_{it};\, \mu_{0t}, \sigma_0^2) \right]$$

where $N(x_{it};\, \mu_{jt}, \sigma_j^2)$ is defined as before (cluster specific), $p_{j0}$ represents the probability that any particular experiment should be modeled by the cluster specific distribution as opposed to a generic model (normal) model $N(x_{it};\, \mu_{0t}, \sigma_0^2)$ which is the same for all clusters. So the cluster model still relies on the underlying mean profile but it is now possible for a gene to be considered a part of the cluster even though it deviates substantially from the mean profile for some of the experiments.

The parameters in this refined model can be again found iteratively via the EM algorithm (we omit the details).

## Latent processes and matrix decomposition

Instead of clustering the data, we can try to uncover the latent biological processes and relate the expression levels of individual genes to such processes. Suppose there are $K$ underlying processes whose activities (e.g., defined by hypothesized master regulators) are given by $\{F_{it}\}$, $i = 1, \ldots, K$, $t = 1, \ldots, n$. In other words, $F_{it}$ defines the activity of the $i^{th}$ process in the $t^{th}$ experiment. We envision $K$ to be much smaller than either $m$ or $n$ so that we have a chance to reconstruct these activities from the available data. Now, each gene is assigned to a subset of these processes (subset that we will optimize). So, for example, the expression level of gene $i$ in experiment $t$ is modeled as

$$x_{it} \sim N(\, \mu_{it}, \sigma^2\,), \text{ where } \mu_{it} = \theta_{i0} + \sum_{j=1}^{k_i} \theta_{ij} F_{I_i(j),t}$$

In other words, $x_{it}$ is normally distributed with mean that is a linear combination of the activities of those $k_i$ underlying processes that $i$ is associated with, indexed by $I_i(1), \ldots, I_i(k_i)$. The coefficients $\theta_{ij}$ define how the gene behaves in response to the processes; these coefficients are assumed to be the same across different experiments. The variability of the expression measurements is $\sigma^2$, which, for simplicity, is the same for all genes and experiments. This model involves a number of parameters (summarized below) that we have to estimate from the available data $X$.

Latent processes: $$F = \begin{bmatrix} F_{11} & F_{12} & \cdots & F_{1n} \\ F_{21} & F_{22} & \cdots & F_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ F_{K1} & F_{K2} & \cdots & F_{Kn} \end{bmatrix}$$

Gene specific: $k_i \in \{0, \ldots, K\}$ (number of processes $i$ participates in)

$\{I_i(1), \ldots, I_i(k_i)\}$ (indexes of the processes $i$ depends on)

$\{\theta_{i0}, \ldots, \theta_{ik_i}\}$ (how $i$ depends on the latent processes)

Overall variability: $\sigma^2$

In the simplest (unconstrained) case all the genes can depend on all the underlying processes. We would have $m \cdot (1 + K) + K \cdot n + 1$ parameters to estimate from the data matrix with $m \cdot n$ entries. This unconstrained problem essentially reduces to singular value decomposition. Omitting $\theta_{i0}$ and $\sigma^2$ for simplicity, we would try to approximate $X \approx \theta F$ where

$$\theta = \begin{bmatrix} \theta_{11} & \theta_{12} & \cdots & \theta_{1K} \\ \theta_{21} & \theta_{22} & \cdots & \theta_{2K} \\ \cdots & \cdots & \cdots & \cdots \\ \theta_{m1} & \theta_{m2} & \cdots & \theta_{mK} \end{bmatrix}$$

and the approximation is in the mean squared sense. $\theta$ and $F$ could be found via singular value decomposition (but are not unique).