BIOINFORMATICS

Vol. 24 ISMB 2008, pages i156–i164
doi:10.1093/bioinformatics/btn153

Inferring differentiation pathways from gene expression

Ivan G. Costa*, Stefan Roepcke, Christoph Hafemeister and Alexander Schliep*

Department of Computational Molecular Biology, Max Planck Institute for Molecular Genetics, Berlin, Germany

ABSTRACT

Motivation: The regulation of proliferation and differentiation of embryonic and adult stem cells into mature cells is central to developmental biology. Gene expression measured in distinguishable developmental stages helps to elucidate underlying molecular processes. In previous work we showed that functional gene modules, which act distinctly in the course of development, can be represented by a mixture of trees. In general, the similarities in the gene expression programs of cell populations reflect the similarities in the differentiation path.

Results: We propose a novel model for gene expression profiles and an unsupervised learning method to estimate developmental similarity and infer differentiation pathways. We assess the performance of our model on simulated data and compare it with favorable results to related methods. We also infer differentiation pathways and predict functional modules in gene expression data of lymphoid development.

Conclusions: We demonstrate for the first time how, in principal, the incorporation of structural knowledge about the dependence structure helps to reveal differentiation pathways and potentially relevant functional gene modules from microarray datasets. Our method applies in any area of developmental biology where it is possible to obtain cells of distinguishable differentiation stages.

Availability: The implementation of our method (GPL license), data and additional results are available at http://algorithmics.molgen.mpg.de/Supplements/InfDif/

Contact: filho@molgen.mpg.de, schliep@molgen.mpg.de

Supplementary information: Supplementary data is available at *Bioinformatics* online.

1 INTRODUCTION

Cell differentiation In all multicellular organisms somatic differentiated cells develop from embryonic stem cells in the formation phase and from adult tissue-specific stem cells in the adult phase. The study of triggers and molecular programs that drive cells through well-defined proliferation and differentiation stages is a central theme of developmental biology. In classical models of such processes external or internal factors initiate and drive differentiation steps in a non-reversible manner. They are conveniently depicted in diagrams that resemble genealogies of developmental stages, which we call developmental trees throughout this article. Recently, the gene expression programs of developmental trees have been studied extensively using microarrays, which helps to elucidate underlying molecular processes (Akashi *et al.*, 2003; Anisimov *et al.*, 2007; Ferrari *et al.*, 2007; Hyatt *et al.*, 2006; Tomancak *et al.*, 2002).

Analysis Finding functional modules of co-regulated genes during the course of development with an unsupervised learning method is a crucial initial step in the large scale analysis of developmental processes. Ideally, the method of choice should exploit inherent dependencies arising from the data. It is, for example, well accepted that models taking temporal dependencies into account are superior for analyzing gene expression time-courses (Bar-Joseph, 2004).

In previous work we demonstrated how to exploit detailed knowledge about the differentiation pathways to infer gene modules with distinct developmental profiles from genome scale gene expression data (Costa *et al.*, 2007b). There, we showed that the expression programs of developmental stages reflect the similarity of stages close by in the developmental tree. For exploring the dependencies arising from such similarities, we proposed the use of Dependence Trees (`DTree`) in which the known developmental tree imposed the dependence structure. We combined several of these `DTrees` in a mixture to model groups of co-regulated genes.

Novel contributions Here, we propose an extension of the above method for inferring developmental similarity and differentiation pathways as reflected in the dependence structure of modules of co-expressed genes, regardless of the underlying developmental tree. Our method estimates the structure of *each* component of a mixture of Dependence Trees (`MixDTrees`). Furthermore, we use maximum-a-posteriori (MAP) estimates for the parameters of the `DTree`, which makes the method robust to overfitting. We assess the performance of our model on simulated data and compare it with favorable results of other unsupervised methods.

We also infer differentiation pathways and predict functional modules in gene expression data of lymphoid development. Lymphoid development has been extensively studied, many developmental stages are known, and there is a large amount of available data on distinct stages of development and in several cell lineages (see Figure 3 (left)). We use biological annotation data from Gene Onotology (GO); (Ashburner, 2000) and Kyoto Encylcopedia of Gene and Genome (KEGG) (Kanehisa *et al.*, 2006) to assist the interpretation of the inferred gene modules. The results show that the inference of `DTree` structures for modules of co-regulated genes helps to reveal differentiation pathways and functional relevant groups of genes. A comparison with other unsupervised methods commonly used for gene expression indicates the advantage of `MixDTrees` in terms of quality and interpretability.

Related methods Mixtures of Dependence Trees are part of the graphical model (or Bayesian network) formalism (Friedman, 2004). They have been applied before, for example, in image recognition (Chow and Liu, 1968; Meila and Jordan, 2001) and detection of mutagenic trees (Beerenwinkel *et al.*, 2004), but exclusively to data of discrete nature. Moreover, our method has some relations to bi-clustering (e.g. Brunet *et al.*, 2004; Tanay *et al.*, 2002), as it is able to find not only coexpressed genes but also developmental conditions with similar expression profiles. Nevertheless, bi-clustering methods make no implicit use of any dependencies (developmental or temporal) in these data sets. There is also a relation to the estimation of sparse covariance matrices, as Dependence Trees represent a subclass of them. Chaudhuri *et al.* (2007) apply an

*To whom correspondence should be addressed.

iterative conditional fitting method for computing sparse covariance matrices from arbitrary undirected graphs. This method, however, does not offer a solution for inferring the graph structure and has high computational cost. Schaefer and Strimmer (2005), who approach a similar problem in the context of gene association networks, use a shrinkage factor in a computationally efficient way for zeroing entries in the covariance matrix, while keeping it well conditioned. Both methods are not able to find association networks, which are specific for particular gene modules, as performed by `MixDTrees`.

2 METHODS

2.1 Dependence Trees (`DTree`)

Let $X=(X_1,\ldots,X_u,\ldots,X_L)$ be a L-dimensional continuous random vector, where the variable X_u denotes the expression values of the developmental stage u and $x=(x_1,\ldots,x_L)$ denotes a realization of X representing gene expression values of a gene in the developmental stages $1,\ldots,L$. A `DTree` is defined as a probabilistic model representing dependencies between variables in X, which follow a tree structure.

Consider a directed graph (V,E) with $|V|=L$, where each vertex in V represents a variable in X, and a directed edge $(v,u)\in E$ indicates that variable X_u is dependent on variable X_v. A directed graph is a directed tree, if the graph is connected, all vertices except the root have in-degree equal to 1, and there are no cycles in the graph. For simplicity, we represent a `DTree` structure by the parent map, $pa:\{1,\ldots,L\}\mapsto\{1,\ldots,L\}$, where $pa(u)=v$ means that $(v,u)\in E$. The root of the `DTree` structure, which has no incoming edges, is represented by $pa(u)=u$.

The probability density function (pdf) of a `DTree` is a second-order approximation of a joint pdf on a L-dimensional continuous random vector X (Chow and Liu, 1968),

$$p[x]\approx p_t[x|\theta]=\prod_{u=1}^{L}p[x_u|x_{pa(u)},\tau_u], \tag{1}$$

where we denote the model parameters by $\theta=(\text{pa},\tau_1,\ldots,\tau_u,\ldots\tau_L)$ and p_t is the pdf of a `DTree`. For example, for the DTree in Figure 1 left, we have $p_t[x_A,x_B,x_C,x_D,x_E,x_F]=p[x_A]p[x_B|x_A]p[x_C|x_B]$ $p[x_D|x_A]$ $p[x_E|x_D]p[x_F|x_D]$.

We use conditional Gaussians (Lauritzen and Spiegelhalter, 1988) as probability densities, denoted as $p[x_u|x_{pa(u)},\tau_u]$ in Equation (1). Hence, for a given developmental profile x and a non-root developmental stage u with $pa(u)=v$, the pdf takes the form

$$p[x_u|x_v,\tau_{u|v}]=\frac{1}{(\sqrt{2\pi}\sigma_{u|v})}\exp\left(\frac{-(x_u-\mu_{u|v}-w_{u|v}x_v)^2}{2\sigma^2_{u|v}}\right), \tag{2}$$

where $\tau_{u|v}=(\mu_{u|v},w_{u|v},\sigma^2_{u|v})$ are the parameters for one conditional density in the model. Intuitively, this conditional density models a linear fit of x_u on x_v, where $w_{u|v}$ indicates the slope and $\mu_{u|v}$ the intercept. For the case of the root, $pa(u)=u$, we can simply set $w_{u|u}=0$ and Equation (2) will be equal to a univariate Gaussian (see Appendix A2 for parameter estimates).

2.2 Estimation of the Dependence Tree structure

For a given continuous random variable X the problem of estimating a `DTree` structure can be formulated as finding the $p_t[x|\Theta]$ that best approximates $p[x]$. We can measure the fit between $p[x]$ and the approximation $p_t[x]$ with the relative entropy (D) (Cover and Thomas, 1991), yielding the optimization problem

$$p_t^*=\text{argmin}_{p_t}\text{D}(p[x]||p_t[x|\Theta]). \tag{3}$$

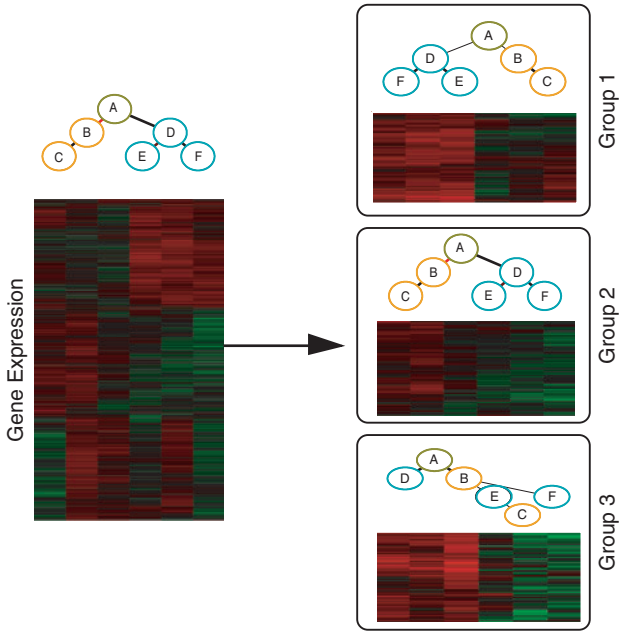

Fig. 1. Illustrative example of a developmental tree and its gene expression data (left). The developmental tree is constituted of a stem cell (stage A), an 'orange' lineage (Stages B and C) and a 'blue' lineage (Stages D, E and F). The red-green plot depicts the relative expression, where lines corresponds to gene profiles and columns to developmental stages ordered as in the above tree. In the right, we depict three groups of genes and their corresponding estimated tree structure as found by `MixDTrees` in the gene expression data in the left (see Section 2.3 for complete plot description).

This is equivalent to finding a tree (Chow and Liu, 1968), here indicated by pa, which has maximal mutual information among tree edges, that is

$$pa^*=\text{argmax}_{pa}\sum_{u=1}^{L}\text{I}(X_u,X_{pa(u)}), \tag{4}$$

where I denotes the mutual information (Cover and Thomas, 1991) (see Appendix A1 for derivations). This problem can be efficiently solved by calculating a maximum weight spanning tree from a fully connected indirected graph, where vertices are the developmental stages $(1,\ldots,L)$ and the weight of an edge (u,v) is equal to the mutual information between the corresponding variables (X_u,X_v) (Chow and Liu, 1968).

If $p[x_u,x_v]$ follows a bivariate Gaussian pdf, the mutual information above can be computed (Cover and Thomas, 1991) by

$$\text{I}(X_u,X_v)=-\frac{1}{2}\log(1-\frac{\sigma^2_{u,v}}{\sigma^2_u\sigma^2_v}). \tag{5}$$

Note that the mutual information is proportional to the correlation coefficient $\rho_{u,v}=\sigma^2_{u,v}/\sigma^2_u\sigma^2_v$. Hence, it measures the dependence between the two variables; $\text{I}(X_u,X_v)=0$ if both variables are independent. Furthermore, as the mutual information is symmetric, $\text{I}(X_u,X_v)=\text{I}(X_v,X_u)$, the estimation method does not determine direction of edges. Undirected and directed tree representations of `DTree` have equivalent pdfs (Meila and Jordan, 2001), directions of edges do not matter. For obtaining a directed tree, we select one particular node as root and direct all edges away from it.

2.3 Mixtures of Dependence Trees (`MixDTrees`)

We do not expect that all genes in a particular developmental process will share the same dependence structure, nor that the most likely `DTree` will exactly match the developmental tree *per se*. Indeed, we expect that some

genes will be particularly correlated in particular developmental lineages, but not in others. For example, group 1 from Figure 1 has genes tightly overexpressed in the blue lineage ($\{X_D, X_E, X_F\}$), as does group 2 in the orange lineage ($\{X_B, X_C\}$). We also expect that some genes, which are important for earlier developmental stages, to be tightly coexpressed in stages near the root, but not in mature cell types (leaf vertices). See for example group 3 in Figure 1, which exhibits overexpression in all earlier stages ($\{X_A, X_B, X_D\}$). To infer these group-specific dependencies, we estimate a mixture of K `DTrees`, where each component can have a distinct tree structure.

We combine a set of K `DTrees` in a mixture model $p[x|\Theta] = \sum_{k=1}^{K} \alpha_k p_t^k[x|\theta_k]$, where $\theta_k = (\text{pa}_k, \tau_{1k}, \ldots, \tau_{Lk})$ denotes the parameters of the k-th `DTree` and α_k is proportional to the number of developmental profiles assigned to the k-th `DTree`; as usual $\alpha_k \geq 0$ and $\sum_{k=1}^{K} \alpha_k = 1$. The mixture of Dependence Trees can be estimated with the Expectation–Maximization algorithm (Dempster *et al.*, 1977). The estimation of the tree structures simply requires an additional computational procedure in the M-Step (Meila and Jordan, 2001).

Furthermore, we propose a MAP estimation to regularize the parameters $w_{u|v,k}$ and $\sigma^2_{u|v,k}$ of the pdfs from Equation (2) and prevent overfitting when there is little evidence for a given model (or for low α_k). We obtain the values of the hyper-parameters in an empirical Bayes fashion (Carlin and Louis, 2000), and use MAP point estimates at each M-Step of the Expectation–Maximization (EM) algorithm (see Appendix A2 for details on parameter estimates). The EM algorithm with MAP point estimates achieves results comparable to a more computationally expensive Markov Chain Monte Carlo method (Fraley and Raftery, 2007).

Note that in principle one could use a mixture of Gaussians (MoG) with full covariance matrix to model any arbitrary dependence in data with continuous variables. Due to the unbounded likelihood function, such method is prone to overfitting (McLachlan and Peel, 2000). To prevent this, several simplifications of the parameterization of the covariance matrix have been proposed (Banfield and Raftery, 1993; Celeux and Govaert, 1995), making distinct *a priori* assumptions of the variable dependencies. `MixDTrees` represents a new type of covariance matrix parameterization that is equivalent to inputing zeros on entries of the inverse of the covariance matrix on pairs of variables with no connecting edge (Lauritzen, 1996). Thus, the number of parameters required for representing a `DTree` is linear on the data dimensionality ($3L-1$), while it is quadratic for a multivariate Gaussian with full covariance matrix ($L + L*(L-1)/2$).

Visualization of gene groups To highlight the coexpression of developmental stages, as indicated by the estimated `DTree`, we perform the following. Gene groups are depicted as a heat-map with red values indicating overexpression and green values indicating underexpression (Eisen *et al.*, 1998). There, the lines (gene profiles) are ordered as proposed in Bar-Joseph *et al.* (2001). This procedure orders genes with similar expression profiles to be close in the heat-map. Following this idea, for the columns (developmental stage profiles), we compute all possible columns orderings and select the one which has a minimal difference in the mutual information of adjacent columns. To further help the interpretation of individual groups, we compute strongly connected components (Cormen *et al.*, 2001)—SCC for short—in the graph returned after thresholding the mutual information matrix. An optimal threshold parameter is obtained by evaluating the resulting SCC with the silhouette index (Kaufman and Rousseeuw, 1990). SCC is represented by dashed shapes around developmental stages and indicates, within a `DTree`, which developmental stages in a particular branch have similar expression profiles.

3 RESULTS AND DISCUSSIONS

3.1 Simulated data

To investigate general characteristics of `MixDTrees` and compare it with other methods, we use simulated data from mixture models with different degrees of variable dependence, and apply several unsupervised learning methods.

Data We generate data from mixtures with four types of variable dependence ranging from: Gaussians with diagonal covariance matrix (Σ^{diag}), `DTree` with low variate dependence (Σ^{DTree^-}), `DTree` with high variate dependence (Σ^{DTree^+}) and Gaussians with full covariance matrix (Σ^{full}). These choices range from the independent case (Σ^{diag}) to the complete dependent case (Σ^{full}). For each setting, we generate 10 such mixtures, and sample 500 development profiles from each. In all cases, we chose the μ from the range $[-1.5, 1.5]$, $L=4$, $K=5$ and mixture coefficients equal to $\alpha = (0.1, 0.15, 0.2, 0.2, 0.35)$. For Σ^{diag} , diagonal entries are sampled from $[0.01, 1.0]$, and non-diagonal entries are set to zero. For Σ^{DTree}, we randomly generate tree structures, one for each mixture component, and then choose $\sigma^2_{u|v,k}$ from $[0.01, 1.0]$ and $w_{u|v,k}$ from $[0.0, 0.5]$ for Σ^{DTree^-} and $w_{u|v,k}$ from $[0.0, 1.0]$ for Σ^{DTree^+}. The generation of Σ^{full} is based on the eigenvalue decomposition of the covariance matrix ($\Sigma = Q\Lambda Q^T$) as in (Qiu and Joe, 2006), where Λ is drawn from $[0.01, 0.5]$. The orthogonal matrix Q is obtained by sampling values from a lower triangular matrix M from the range $[20, 40]$, followed by the Gram–Schmidt orthogonalization procedure.

We apply MoG with full and diagonal covariance matrices and `MixDTrees` with MLE and MAP estimates to all datasets. The mixture estimation method is initialized with $K=5$ random `DTrees` (or multivariate Gaussians). Subsequently, we train the mixture model using the EM algorithm. To avoid the effect of the initialization, all estimations are repeated 15 times, and the one with highest likelihood is selected. We also performed clustering with k-means (McQueen, 1967), self-organizing maps (SOM) and spectral clustering (Ng *et al.*, 2001). For all methods, the number of cluster was also set to five and for SOM, default parameters were used (Vesanto *et al.*, 2000). We compare the class information from the data generation to compute the corrected Rand index (Hubbert and Arabie, 1985) and evaluate the clustering solutions.

Results Every method performs well on the datasets generated with the corresponding model assumptions (Fig. 2). An exception is the MoG with full covariance matrices, which has low corrected Rand index for all datasets. An inspection on the specificity index (Fig. A1) indicates that the poor performance of MoG Full is caused by overfitting, since it tends to join real groups. Spectral clustering has a tendency to split real groups (see sensitivity plot in Fig. A1). In both datasets from Σ^{DTree}, `MixDTrees MAP` has higher mean values than `MixDTrees MLE`, which indicates a higher robustness of the MAP estimates (paired t-tests indicate superiority of `MixDTrees MAP` with P-values <0.05 in both Σ^{DTree^-} and Σ^{DTree^+}). Moreover, `MixDTrees MAP` achieves the highest values in all settings (paired t-tests indicate P-values <0.05), outperforming MoG Full, MoG Diagonal, k-means, SOM and spectral clustering, with the exception of MoG Diagonal in the Σ^{diag} data. These results show that `MixDTrees MAP` has a better performance on data coming from distinct dependence structures when compared to the other methods, and it is robust against overfitting.

3.2 Lymphoid development

To evaluate the application of `DTrees` and `MixDTrees` to real biological data, we use gene expression data from lymphoid cell development. First, we compare the `DTree` structure inferred from

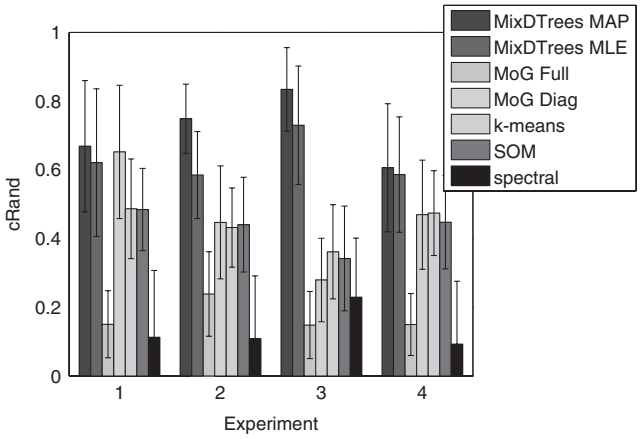

Fig. 2. We depict the mean corrected Rand index (Hubbert and Arabie, 1985) of true label recovery for distinct clustering methods (y-axis) against data generated with distinct model assumptions (x-axis) (1 for Σ^{diag}, 2 for Σ^{DTree^-}, 3 for Σ^{DTree^+} and 4 for Σ^{full}). These choices range from the independent case Σ^{diag} to the complete dependent case Σ^{full}.

the whole dataset with the lymphoid developmental tree. Then, we apply `MixDTrees` to find modules of co-regulated genes, and evaluate the results with GO and KEGG enrichment analysis. Finally, we compare our method with other unsupervised learning methods.

Data We produce an expression compendium of mouse lymphoid cell development by combining measurements of wild-type control cells from several studies (Akashi *et al.*, 2003; Niederberger *et al.*, 2005; Poirot *et al.*, 2004; Tze *et al.*, 2005; Yamagata *et al.*, 2006) based on the Affymetrix U74 platform. In detail, our data contain four stages of early development hematopoietic cells (Akashi *et al.*, 2003) [hematopoietic stem cell (HSC), multipotent progenitor (MPP), common lymphoid progenitor (CLP), common myeloid progenitor (CMP)]; three B-cell lineage stages (Tze *et al.*, 2005) [pro-B cells (Bpro), pre-B cells (Bpre) and immature B-cells (Bimm)]; one natural killer (NK) stage (Poirot *et al.*, 2004); and four T-cell lineage stages [double negative T-cells (TDN) (Niederberger *et al.*, 2005), cd4 T cells (TCD4), cd8 T-cells (TCD8) and natural killer T-cells (TNK; Yamagata *et al.*, 2006)]. The developmental tree describing the order of differentiation of the cells is depicted in Figure 3 left. We preprocess the data as follows: we apply variance stabilization (Huber *et al.*, 2002) on all chips, take median values of stages with technical replicates, use HSC values as reference values and transform all expression profiles to log-ratios. We keep genes showing at least a 2-fold change in one developmental stage. The final data consists of 11 developmental stages and 3697 genes.

Inferring the `DTree` *structure* An initial question is how well we can recover the original developmental tree, as agreed upon by developmental biologists (Fig. 3 left), if we apply the structure estimation method described in Section 2.2 to the complete gene expression data (see Fig. 3 right for the estimated `DTree`). To quantify the difference between these trees, we compute the path distance between all pairs of vertices, and calculate the Euclidean distance between the resulting distance matrices (Steel and Penny, 1993), which indicates a distance of 15.74. To assess the statistical significance of this distance, we generate 1000 random trees with the

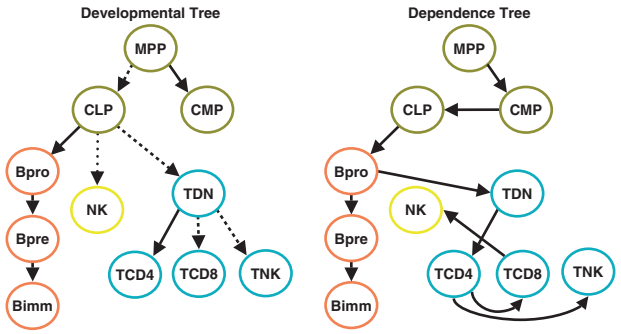

Fig. 3. We depict the developmental tree with the stages contained in the Lymphoid dataset (left). Early hematopoietic cells are depicted in olive green, B-cells in orange, NK-cell in yellow and T-cells in blue. The dashed edges represent edges wrongly assigned in the `DTree` estimated from the Lymphoid data, which connect pairs of vertices with a path distant of one, while the dotted edge represents a edge with a path distance of three. We have in the right the `DTree` estimated from the Lymphoid data.

same distribution of outgoing edges per vertex as the developmental tree. For each random tree, we compute the distance with the developmental tree. This test indicates a P-value of 0.002 of finding a distance as low as 15.74. Looking at these differences in detail, we can observe that 5 out of the 10 edges are correctly assigned, 4 edges connects vertices pairs with a path distance equal to 1 (i.e. MPP and CLP, CLP and TDN, TDN and TCD8 and TDN and TNK), and one edge connect vertices with a path distance of 3 (NK is connected to TCD8 instead the CLP). Furthermore, wrong edges have a tendency to be connected to vertices in the same level of the developmental tree (e.g. TCD8 and TNK both connected with the TCD4).

Another important question is how well does the `DTree` capture dependence in the data? One simple way to assess this is to measure the proportion of the mutual information represented in the tree edges, in comparison to the total mutual information on all pairs of variables. For a `DTree` structure pa, the treeness index can be defined as

$$T(pa) = \frac{\sum_{u=1}^{L} \mathrm{I}(X_u, X_{pa(u)})}{\sum_{u=1}^{L} \sum_{v=u+1}^{L} \mathrm{I}(X_u, X_v)}. \quad (6)$$

For example, the score for the developmental tree (Fig. 3 left) is 0.22, whereas for the estimated `DTree` (Fig. 3 right), the 'treeness' index is 0.42. For measuring the statistical significance of this, we generate random data (1000 times) by shuffling values of gene expression profiles x_i, estimate a `DTree` from this random data, and measure its corresponding treeness index. This test indicates a P-value of 0.01 of finding a treeness index as high as 0.42.

Inferring Gene Modules with `MixDTrees` We estimate `MixDTrees` with MAP estimates from the Lymphoid data following the protocol in Section 3.1. The Bayesian information criterion (McLachlan and Peel, 2000) indicates 13 groups as optimal. We analyze the functional relevance of the groups of genes found by enrichment analysis (Beissbarth and Speed, 2004) with GO and pathway data from the KEGG (Kanehisa *et al.*, 2006). For the GO (or KEGG) enrichment analysis, we use the statistic of the Fisher-exact test to obtain a list of GO terms (or KEGG pathways),

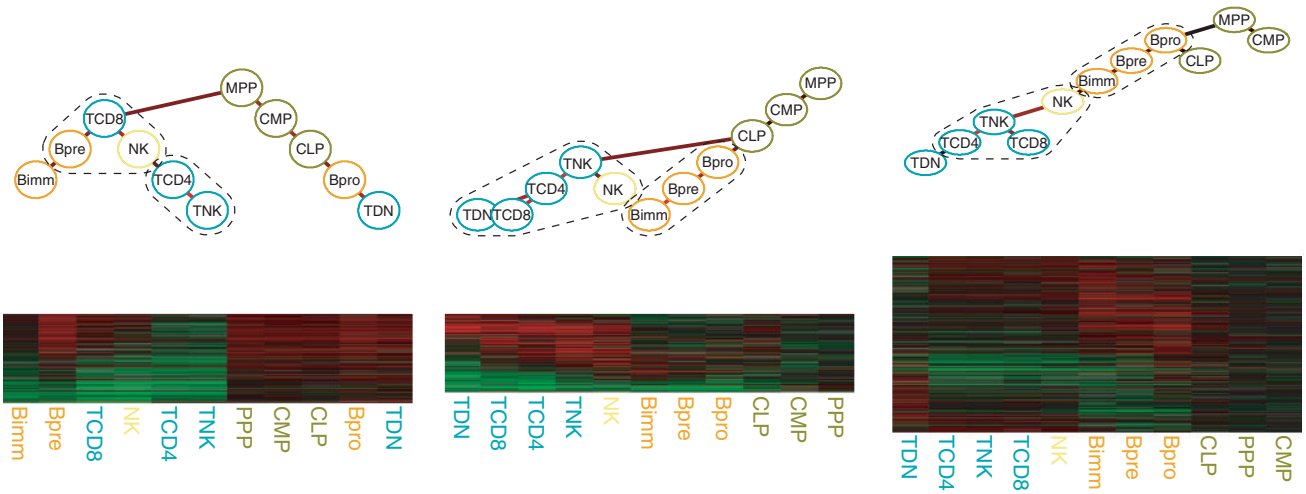

Fig. 4. We depict the `DTree` and expression profiles of groups 1, 4 and 5 from `MixDTrees MAP`. Dashed shapes around developmental stages represent the SCC. See Section 2.3 for complete description of plotting procedure.

whose annotated genes are overrepresented in a group. We correct for multiple testing following Benjamini and Yekutieli (2001). All results, group plots, list of genes per cluster, KEGG and GO enrichment analysis can be found at http://algorithmics.molgen.mpg.de/Supplements/InfDif/

First, we measure the average treeness of the `MixDTrees` (we calculate Equation (6) and take the sum weighted by α). For the `MixDTrees MAP` this value was 0.54, which indicates an increase of 28% over the treeness index for the single `DTree`. This reinforces our point that mixture of Dependence Trees with estimated structures is more successful in modeling dependencies in the data.

In relation to the groups of coexpressed genes found by `MixDTrees`, overall, stages from the same developmental lineage are at the same branches of the estimated `DTree` structure. Furthermore, groups present prototypical expression patterns such as overexpression in cells from a particular lineage, but not in other lineages (e.g. groups 2 and 5 for B-cells, groups 4 and 6 for T-cells and group 11 for NK-cells) or groups displaying under-expression in particular lineages (e.g. groups 7 and 12 for T-cells and groups 10 and 12 for B-cells).

In Figure 4, we display some of these groups, which we discuss in more details. Group 1 is an interesting case, where the `DTree` structure differs drastically from the developmental tree. The right branch, which is formed by stages MPP, CLP, CMP, TDN and Bpro, has only early developmental stages, and all display high overexpression patterns. On the other hand, the majority of stages in the SCC of the left branch (Bimm, Bpre, TCD8, NK, TCD4, TNK) are immature developmental stages (leaves in Fig. 3 left). Enrichment analyses from GO and KEGG show that group 1 is overrepresented for *cell cycle* and *DNA repair* genes (P-values <0.001). This matches the biological knowledge that earlier differentiation stages of development are cycling cells, while immature cells are resting (Matthias and Rolink, 2005; Rothenberg and Taghon, 2005). Group 4 contains an SCC (left branch) with all T-cell stages plus the closely related NK-cell. At these stages, genes display an overexpression pattern. Enrichment analysis indicates overrepresentation for GO terms such as *T-cell activation, differentiation and receptor signaling*; and KEGG pathways such as *T-cell signaling* and *NK-cell mediated cytotoxicity* (P-values <0.001). Similarly, group 5 has a SCC with all B-cell stages. Furthermore, for B-cell stages, genes are preferentially overexpressed. GO analysis also indicates enrichment for terms such as *B-cell activation* (P-values <0.001), while KEGG analysis indicates enrichment in pathways such as *Hematopoietic cell lineage* and *B-Cell receptor signaling* (P-values <0.05). These results indicate how `MixDTrees` can be used in finding groups of biologically related genes, as well that the associated `DTree` structure adds relevant information regarding expression similarly of developmental stages.

Comparison with other methods For comparison purposes, we also perform clustering of the Lymphoid data with other methods: k-means, SOM, MoG with full covariance matrix, MoG with diagonal matrix and the bi-clustering methods Samba (Tanay *et al.*, 2002) and non-negative matrix factorization (Brunet *et al.*, 2004). Additionally, we evaluate distinct variations of the `MixDTrees`: MAP and MLE with `DTree` structure estimation and MAP estimates with the `DTrees` fixed to the structure from Figure 3 left, as in our previous approach (Costa *et al.*, 2007b). For SOM and Samba, default parameters were used (Vesanto *et al.*, 2000 , Tanay *et al.*, 2002). For the mixtures, NMF, k-means and SOM, the number of clusters was set to 13. Samba, which detects the number of clusters automatically, determined 19 clusters.

To evaluate the performance of the methods, we use a heuristic of comparing P-values of KEGG enrichment analysis in a similar way as Ernst *et al.* (2005). The results of the comparison of `MixDTrees MAP` and MoG Diag can be seen in Figure 5. In short, the best method should present a higher enrichment for a higher number of KEGG pathways, i.e. `MixDTrees MAP` was superior to MoG Diag in 9 out of 11 pathways. Furthermore, most of the 11 KEGG pathways enriched with a P-value <0.05 in one of the methods (points depicted in Fig. 5) are directly involved in immune system and developmental processes. We apply the same procedure for all pairs of methods and count the events $\{P\text{-value } m_1 < P\text{-value } m_2\}$, where m_1 and m_2 are the two methods

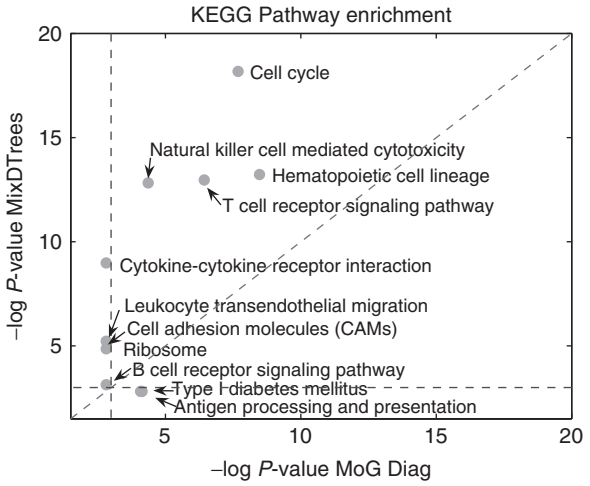

Fig. 5. We depict the scatter plot comparing the KEGG pathway enrichment of MoG Diag (x-axis) and `MixDTrees MAP` (y-axis). We use $-\log(P)$-values, where higher values indicate a higher enrichment. The blue lines corresponds to $-\log(P)$-value cut-off used (P-value of 0.05). Only KEGG pathways with a $-\log(P)$-value higher than (2.99) in one of the results are included. `MixDTrees MAP` had a higher enrichment for 9 out of the 11 KEGG pathways.

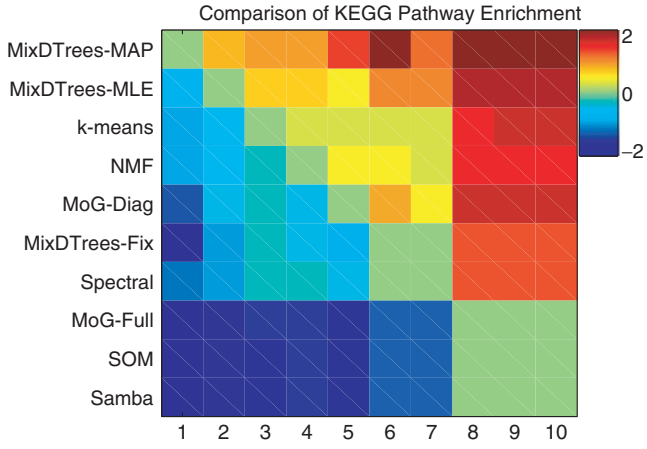

Fig. 6. Heat-map plot displaying the comparison of KEGG enrichment for 10 distinct clustering methods. More precisely, entries in the plot correspond to $\log(\#\{P\text{-value } m_y < P\text{-value } m_x\}/\#\{P\text{-value } m_y > P\text{-value } m_x\})$, where red (or blue) values indicate that the method on the y-axis (m_y) had a higher (or lower) count of enriched KEGG pathways than the method on the x-axis (m_x); numbers on the x-axis correspond to the methods on the y-axis.

in comparison. As can be seen in Figure 6, `MixDTrees MAP` outperforms all methods, while `MixDTrees MLE` and k-means also obtained higher enrichment than other methods. Overall, SOM, MoG Full and Samba obtain poor enrichment results, and were outperformed by other methods. We repeat the same analysis for GO enrichment (see Supplementary Material). The result are in agreement with the KEGG enrichment analysis, again `MixDTrees MAP` had higher enrichment than all other methods, while SOM and MoG Full obtain poor results.

4 CONCLUSIONS

Understanding the details of cell differentiation is a central question in developmental biology and also of high relevance for clinical applications, for example, when considering lymphoid development. The full spectrum of differentiation paths is still unknown, as recent studies suggest that there exist alternative paths in lymphoid development (Graf and Trumpp, 2007).

Here we present a novel statistical model called `DTree` for gene expression data measured for cells in distinct differentiation stages and an unsupervised learning method for finding functional modules and their specific differentiation pathways in the course of development. We show that the `DTree` inferred from the whole dataset approximates the dependencies intrinsic to Lymphoid development well. Furthermore, by combining several `DTrees` in a mixture, we find models specific to groups of co-regulated genes displaying distinct differentiation pathways reflected by the distinct dependence structures. These groups usually have a lineage specific expression pattern supported by term enrichment analysis of gene annotation from KEGG and GO, which indicates development-specific module function. Moreover, the `DTree` structure, which indicates the dependence between the stages, is valuable for the biological interpretation of these groups. On simulated data `MixDTrees` compares favorably to other methods routinely used for finding functional modules, even for data arising from variable dependence structures. In particular, our method is not susceptible to overfitting, which is otherwise a frequent problem in the estimation of mixture models from sparse data.

Alternative paths of differentiation can be investigated with the use of statistical models with higher order dependencies (Chaudhuri *et al.*, 2007; Thiesson *et al.*, 1998), which currently do not provide an efficient and exact method for the estimation of relevant dependencies. As our probabilistic method is based on a well-studied statistical framework (Friedman, 2004), we can easily benefit from extensions to mixtures proposed to integrate, or fuse, further biological information, such as sequence (Schönhuth *et al.*, 2006) or *in situ* data (Costa *et al.*, 2007a). Thus we will be able to explore developmental regulatory networks controlling module-specific differentiation from fused genomics and transcriptomics data.

ACKNOWLEDGEMENTS

We would like to express our gratitude to Fritz Melchers and Roland Krause (MPI for Infection Biology, Berlin) for helpful discussions.

Funding: The I.G.C. would like to acknowledge funding from the CNPq(Brazil)/DAAD.

Conflict of Interest: none declared.

REFERENCES

Akashi,K. *et al.* (2003) Transcriptional accessibility for genes of multiple tissues and hematopoietic lineages is hierarchically controlled during early hematopoiesis. *Blood*, **101**, 383–389.

Anisimov,S.V. *et al.* (2007) 'NeuroStem Chip': a novel highly specialized tool to study neural differentiation pathways in human stem cells. *BMC Genomics*, **8**, 46.

Ashburner,M. (2000) Gene ontology: tool for the unification of biology. *Nat. Genet.*, **25**, 25–29.

Banfield,J.D. and Raftery,A.E. (1993) Model-based gaussian and non-gaussian clustering. *Biometrics*, **49**, 803–821.

Bar-Joseph,Z. (2004) Analyzing time series gene expression data. *Bioinformatics*, **20**, 2493–2503.

Bar-Joseph,Z. *et al*. (2001) Fast optimal leaf ordering for hierarchical clustering. *Bioinformatics*, **17**(Suppl. 1), i22–i29.
Beerenwinkel,N. *et al*. (2004) Learning multiple evolutionary pathways from cross-sectional data. In *RECOMB '04: Proceedings of the Eighth Annual International Conference on Research in Computational Molecular Biology*. ACM Press, New York, pp. 36–44.
Beissbarth,T. and Speed,T.P. (2004) GOstat: find statistically overrepresented gene ontologies within a group of genes. *Bioinformatics*, **20**, 1464–1465.
Benjamini,Y. and Yekutieli,D. (2001) The control of the false discovery rate in multiple testing under dependency. *Annal. Stat.*, **29**, 1165–1188.
Brunet,J.-P. *et al*. (2004) Metagenes and molecular pattern discovery using matrix factorization. *Proc. Natl Acad. Sci. USA*, **101**, 4164–4169.
Carlin,B.P. and Louis,T.A. (2000) *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall/CRC, Boca Raton, FL, USA.
Celeux,G. and Govaert,G. Gaussian parsimonious clustering models. *Pattern Recognition*, **28**, 781–793.
Chaudhuri,S. *et al*. (2007) Estimation of a covariance matrix with zeros. *Biometrika*, **94**, 199–216.
Chow,C. and Liu,C. (1968) Approximating discrete probability distributions with dependence trees. *IEEE Trans. Info. Theory*, **14**, 462–467.
Cormen,T.H. *et al*. (2001) *Introduction to Algorithms*. 2n edn. MIT Press and McGraw-Hill, Cambridge, MA, USA.
Costa,I.G. *et al*. (2007a) Semi-supervised learning for the identification of syn-expressed genes from fused microarray and in situ image data. *BMC Bioinformatics*, **8**(Suppl. 10), S3.
Costa,I.G. *et al*. (2007b) Gene expression tress in blood cell development. *BMC Immunol.*, **8**, 25.
Cover,T.M. and Thomas,J.A. (1991) *Elements of Information Theory*. John Wiley and Sons, Inc., New York.
Dempster,A. *et al*. (1977) Maximum likelihood from incomplete data via the EM algorithm. *JRSSB*, **39**, 1–38.
Eisen,M. *et al*. (1998) Cluster analysis and display of genome-wide expression patterns. *Proc. Natl Acad. Sci. USA.*, **95**, 14863–14868.
Ernst,J. *et al*. (2005) Clustering short time series gene expression data. *Bioinformatics*, **21**(Suppl. 1), i159–i168.
Ferrari,F. *et al*. (2007) Genomic expression during human myelopoiesis. *BMC Genomics*, **8**, 264.
Fraley,C. and Raftery,A.E. (2007) Bayesian regularization for normal mixture estimation and model-based clustering. *J. Classif.*, **24**, 155–181.
Friedman,N. (2004) Inferring cellular networks using probabilistic graphical models. *Science*, **303**, 799–805.
Graf,T. and Trumpp,A. (2007) Haematopoietic stem cells, niches and differentiation pathways. *Poster. Nat. Rev. Immunol.* URL http://www.nature.com/nri/posters/hsc/index.html (last accessed date January 1 2008).
Hubbert,L.J. and Arabie,P. (1985) Comparing partitions. *J. Classif.*, **2**, 63–76.
Huber,W. *et al*. (2002) Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, **18**(Suppl. 1), S96–S104.
Hyatt,G. *et al*. (2006) Gene expression microarrays: glimpses of the immunological genome. *Nat. Immunol.*, **7**, 686–691.
Kanehisa,M. *et al*. (2006) From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res.*, **34**(Database issue), D354–D357.
Kaufman,M. and Rousseeuw,P.J. (1990) *Finding Groups in Data: an Introduction to Cluster Analysis*. Wiley, New York, USA.
Lauritzen,S.L. (1996) *Graphical Models*. Oxford University Press, New York, USA.
Lauritzen,S.L. and Spiegelhalter,D.J. (1998) Local computations with probabilities on graphical structures and their application to expert systems. *J. Royal Statis. Soc. B*, **50**, 157–224.
Matthias,P. and Rolink,A.G. (2005) Transcriptional networks in developing and mature B cells. *Nat. Rev. Immunol.*, **5**, 497–508.
McLachlan,G. and Peel,D. (2000) *Finite Mixture Models*. Wiley Series in Probability and Statistics. Wiley, New York.
McQueen,J. (1967) Some methods of classification and analysis of multivariate observations. In *5th Berkeley Symposium in Mathematics, Statistics and Probability*. University of California Press, Berkley, CA, USA, pp. 281–297.
Meila,M. and Jordan,M.I.(2001) Learning with mixtures of trees. *J. Mach. Learn. Res.*, **1**, 1–48.
Ng,A.Y. *et al*. (2001) On spectral clustering: analysis and an algorithm. In *Advances in Neural Information Processing Systems 13*. MIT Press, Cambridge, MA, USA, pp. 849–856.
Niederberger,N. *et al*. (2005) Thymocyte stimulation by anti-TCR-beta, but not by anti-TCR-alpha, leads to induction of developmental transcription program. *J. Leukoc. Biol.*, **77**, 830–841.
Poirot,L. *et al*. (2004) Natural killer cells distinguish innocuous and destructive forms of pancreatic islet autoimmunity. *Proc. Natl Acad. Sci. USA*, **101**, 8102–8107.
Qiu,W. and Joe,H. (2006) Generation of random clusters with specified degree of separation. *J. Classif.*, **23**, 315–334.
Rothenberg,E.V. and Taghon,T. (2005) Molecular genetics of T cell development. *Annu. Rev. Immunol.*, **23**, 601–649.
Schaefer,J. and Strimmer,K. (2005) A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Stat. Appl. Genet. Mol. Biol.*, **4**, Article 32.
Schönhuth,A. *et al*. (2006) Semi-supervised clustering of yeast gene expression. In *Japanese-German Workshop on Data Analysis and Classification*. Springer, Berlin-Heidelberg, Germany.
Steel,M.A. and Penny,D. (1993) Distributions of tree comparison metrics-some new results. *Syst. Biol.*, **42**, 126–141.
Tanay,A. *et al*. (2002) Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, **18**(Suppl. 1), S136–S144.
Thiesson,B. *et al*. (1998) Learning mixtures of dag models. In *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*. Morgan Kaufmann, San Francisco, CA, pp. 504–551.
Tomancak,P. *et al*. (2002) Systematic determination of patterns of gene expression during Drosophila embryogenesis. *Genome. Biol.*, **3**, research: 0081.1–0081:14.
Tze,L.E. *et al*. (2005) Basal immunoglobulin signaling actively maintains developmental stage in immature B cells. *PLoS Biol*, **3**, e82.
Vesanto,J. *et al*. (2000) Som toolbox for matlab. Technical report. Helsinki University of Technology, Helsinki, Finland.
Yamagata,T. *et al*. (2006) A shared gene-expression signature in innate-like lymphocytes. *Immunol. Rev.*, **210**, 52–66.

APPENDIX

A1 DTREE Structure estimation

For a given L-dimensional continuous variable X, the problem of a DTree structure estimation can be defined as finding the pdf $p_t[x]$ that best approximates $p[x]$. We summarize here the solution proposed in Chow and Liu (1968), which considered trees on discrete distributions, and we describe our extension to continuous variates. The solution is based on finding the DTree structure that minimizes the relative entropy between the $p[x]$ and the approximation $p_t[x]$, or

$$p_t^* = \text{argmin}_{p_t} \mathrm{D}(p||p_t). \tag{A1}$$

The relative entropy between $p[x]$ and $p_t[x]$ is defined as (Cover and Thomas, 1991)

$$\mathrm{D}(p||p_t) = \int_X p[x] \log \frac{p[x]}{p_t[x]}.$$

From Equation 1, we have

$$\begin{aligned}\mathrm{D}(p||p_t) &= \int_X p[x]\log p[x] - \int_X p[x] \sum_{u=1}^{L} \log p[x_u|x_{\mathrm{pa}(u)}, \tau_u], \\ &= \mathrm{H}(X) - \int_X p[x] \sum_{u=1}^{L} \log p[x_u] \\ &\quad - \int_X p[x] \sum_{u=1}^{L} \log \frac{p[x_u|x_{\mathrm{pa}(u)}, \tau_u]}{p[x_u]}\end{aligned}$$

By Bayes rule and the definition of entropy (H) and mutual information (I) (Cover and Thomas, 1991), this previous formula

can be simplified to,

$$\mathrm{D}(p||p_t) = \mathrm{H}(X) - \sum_{u=1}^{L} \mathrm{H}(X_u) - \int_X \sum_{u=1}^{L} p[x_u, x_{\mathrm{pa}(u)}] \log \frac{p[x_u, x_{\mathrm{pa}(u)}]}{p[x_u] p[x_{\mathrm{pa}(u)}]},$$

and hence,

$$\mathrm{D}(p||p_t) = \mathrm{H}(X) - \sum_{u=1}^{L} \mathrm{H}(X_u) - \sum_{u=1}^{L} \mathrm{I}(X_u, X_{\mathrm{pa}(u)}). \quad \text{(A2)}$$

Since $\mathrm{H}(X)$ and $\mathrm{H}(X_u)$ are independent of p_t, Equation (A2) reduces to

$$\mathrm{pa}^* = \mathrm{argmax}_{\mathrm{pa}} \sum_{u=1}^{L} \mathrm{I}(X_u, X_{\mathrm{pa}(u)}). \quad \text{(A3)}$$

This problem can be efficiently solved by a maximum weight spanning tree algorithm on the fully connected undirected graph, in which vertices corresponds to the variables and edge weights to the mutual information between variables (Chow and Liu, 1968). This algorithm has a worst case complexity of $O(L^2 \log L)$.

We need to compute $\mathrm{I}(X_u, X_{\mathrm{pa}(u)})$ for a multivariate Gaussian. Given $\mathrm{pa}(u) = v$, the mutual information is defined as,

$$\mathrm{I}(X_u, X_v) = \int_{X_u} \int_{X_v} p[x_u, x_v] \log \frac{p[x_u, x_v]}{p[x_u] p[x_v]} dx_u dx_v. \quad \text{(A4)}$$

Expanding the terms

$$\mathrm{I}(X_u, X_v) = \int_{X_u} \int_{X_v} p[x_u, x_v] \log p[x_u, x_v] dx_u dx_v - \int_{X_u} \int_{X_v} p[x_u, x_v] \log p[x_u] dx_u dx_v - \int_{X_u} \int_{X_v} p[x_u, x_v] \log p[x_v] dx_u dx_v,$$

and by definition of H, we have

$$\mathrm{I}(X_u, X_v) = \mathrm{H}(X_u) + \mathrm{H}(X_v) - \mathrm{H}(X_u, X_v). \quad \text{(A5)}$$

The entropy of a L-dimensional multivariate Gaussian pdf is defined (Cover and Thomas, 1991) as

$$\mathrm{H}(X) = \frac{1}{2} \log (2\pi e)^L |\Sigma_X|, \quad \text{(A6)}$$

where Σ_X is the covariance matrix of X. By substituting Equation (A6) in Equation (A5), we obtain

$$\mathrm{I}(X_u, X_v) = \frac{1}{2} \log (2\pi e \sigma_{X_u}^2) + \frac{1}{2} \log (2\pi e \sigma_{X_v}^2) - \frac{1}{2} \log ((2\pi e)^2 |\Sigma_{X_u, X_v}|),$$

given $|\Sigma_{X_u, X_v}| = \sigma_u^2 \sigma_v^2 - (\sigma_{u,v})^2$,

$$\mathrm{I}(X_u, X_v) = \frac{1}{2} \log \left(\frac{(2\pi e)^2}{(2\pi e)^2} \right) - \frac{1}{2} \log \left(\frac{\sigma_u^2 \sigma_v^2 - (\sigma_{u,v})^2}{\sigma_u^2 \sigma_v^2} \right),$$

and hence,

$$\mathrm{I}(X_u, X_v) = -\frac{1}{2} \log \left(1 - \frac{\sigma_{u,v}^2}{\sigma_u^2 \sigma_v^2}\right). \quad \text{(A7)}$$

A2 MAP Parameters estimates

We use the Expectation Maximization algorithm with MAP point estimates for estimating the `MixDTrees MAP`. We describe here the derivations of the parameter estimates maximizing the MAP. This corresponds to the parameters used in the M-Step of the EM algorithm. All other parameters follow the basic EM framework, and we refer the reader to McLachlan and Peel (2000) for more details.

Let x_{iu} be the expression value of the gene i in development stage u, $1 \le i \le N$ and $1 \le u \le L$. Then, $x_i = (x_{i1}, \ldots, x_{iu}, \ldots, x_{iL})$ is the developmental profile of gene i, and $\mathbf{X}$ corresponds to a data set with N observed genes.

In short, we want to find estimates maximizing

$$p[\Theta|\mathbf{X}, \mathbf{Y}] \approx p[\mathbf{X}, \mathbf{Y}|\Theta] p[\Theta]$$

where $y_i \in \mathbf{Y}$ corresponds to the hidden variable indicating, which mixture component gene profile x_i belongs to. Since `MixDTrees` are based on first-order dependencies, it is sufficient to find the parameters in a simple bivariate scenario $(\mathbf{X}_u, \mathbf{X}_{\mathrm{pa}(u)})$, where $\mathrm{pa}(u) = v$. This simplifies the formula above to

$$p[\Theta|\mathbf{X}_u, \mathbf{X}_v, \mathbf{Y}] = p[\mathbf{X}_u, \mathbf{X}_v, \mathbf{Y}|\Theta] p[\Theta], \quad \text{(A8)}$$

where

$$p[\mathbf{X}_u, \mathbf{X}_v, \mathbf{Y}|\Theta] = \prod_{k=1}^{K} \prod_{i=1}^{N} (\alpha_k \cdot p_t^k [\mathbf{X}_u, \mathbf{X}_v|\Theta_k])^{r_{ik}},$$

and thus

$$p[\Theta] = \prod_{k=1}^{K} p[\Theta_k] = \prod_{k=1}^{K} p[w_{u|v}|\sigma_{u|v,k}^2, \alpha_k] p[\sigma_{u|v,k}^2|\alpha_k] p[\alpha_k],$$

where $\alpha_k = \sum_{i=1}^{N} r_{ik}$ and $r_{ik} = p[y_i = k|x_i]$ is the posterior probability (or responsibility) (McLachlan and Peel, 2000) that gene i belongs to `DTree` k.

A2.1 Priors on parameters

We use the following conjugate priors to regularize the parameter $w_{u|v,k}$ and $\sigma_{u|v,k}^2$ and avoid overfitting when there is low evidence for a given model (or low α_k).

A2.1.1 Prior on deviation parameter For simplicity of computation we work with a precision parameter $\lambda_{u|v,k} = (\sigma_{u|v,k}^2)^{-1}$. We define the prior of $\lambda_{u|v,k}$ to be proportional to

$$p[\lambda_{u|v,k}|\nu_{u|v,k}, \alpha_k] \sim \mathrm{Exponential}\left(\frac{\lambda_{u|v,k}}{\nu_{u|v,k}\alpha_k}\right)$$

where $\nu_{u|v,k}$ is the hyper-parameter.

A2.1.2 Prior on regression parameter The prior of $w_{y|x,k}$ is defined as

$$p[w_{u|v,k}|\sigma_{u|v,k}^2, \alpha_k, \beta_{u|v,k}] = N(0, \beta_{u|v,k}(\lambda_{u|v,k}\alpha_k\sigma_{v|k}^2)^{-1}),$$

which is invariant to the scale of the variates $\mathbf{X}_u$ and $\mathbf{X}_v$, and has $\beta_{u|v,k}$ as a hyper-parameter.

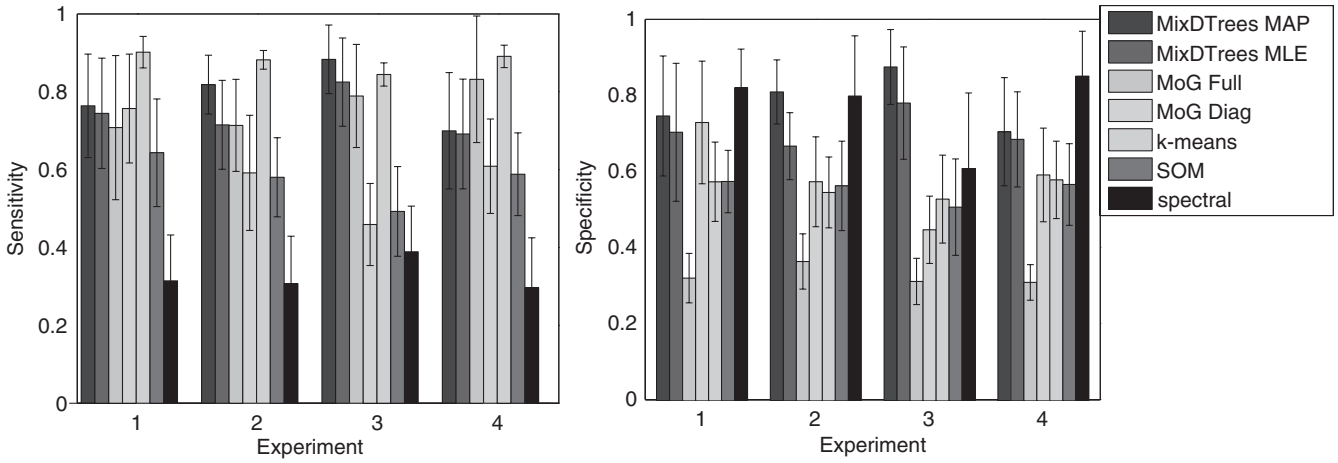

Fig. A1. We depict the sensitivity and specificity of label recovery for distinct clustering methods (y-axis) for distinct model assumptions (x-axis) (1 for Σ^{diag}, 2 for Σ^{DTree^-}, 3 for Σ^{DTree^+} and 4 for Σ^{full}). A low sensitivity is a indicator of joining real clusters; while a low specificity indicates a tendency to split real clusters.

A2.2 MAP estimates

The MAP estimates are obtained by taking the derivative of Equation (A8) in relation to each parameter, which leads to the estimates

$$\hat{\mu}_{u|v,k} = \hat{\mu}_{u|k} - \hat{\mu}_{v|k} w_{u|v,k} \tag{A9}$$

$$\hat{w}_{u|v,k} = \frac{\hat{\sigma}_{u,v|k}}{\hat{\sigma}^2_{v|k}(1+\beta^{-1}_{u|v,k})}. \tag{A10}$$

When $\beta_{u|v,k} \to \infty$, the prior becomes non-informative; that is, the MAP and maximum likelihood (ML) estimates are equal.

$$\hat{\sigma}^2_{u|v,k} = \hat{\sigma}^2_{u|k} - w^2_{u|v,k}\hat{\sigma}^2_{v|k}(1+\beta^{-1}_{u|v,k}) - \nu^{-1}_{u|v,k}. \tag{A11}$$

Again, when $\beta_{u|v,k} \to \infty$ and $\nu_{u|v,k} \to \infty$, the prior becomes non-informative, and MAP and ML estimates are equal.

All the estimates make use of the following sufficient statistics

$$\hat{\mu}_{u|k} = \frac{\sum_{i=1}^N r_{ik} x_{iu}}{\sum_{i=1}^N r_{ik}} \tag{A12}$$

$$\hat{\sigma}^2_{u|k} = \frac{\sum_{i=1}^N r_{ik}(x_{iu} - \hat{\mu}_{u|k})^2}{\sum_{i=1}^N r_{ik}}, \text{ and} \tag{A13}$$

$$\hat{\sigma}_{u,v|k} = \frac{\sum_{i=1}^N r_{ik}(x_{iv} - \hat{\mu}_{v|k})(x_{iu} - \hat{\mu}_{u|k})^2}{\sum_{i=1}^N r_{ik}}. \tag{A14}$$

A2.3 Hyper parameters estimates via empirical Bayes

In an empirical Bayes approach (Carlin and Louis, 2000), we can estimate the MAP value of $\beta_{u|v,k}$ and $\nu_{u|v,k}$ from the data, by taking the derivative of Equation (A8) in relation to the hyper-parameters, that is

$$\hat{\beta}_{u|v,k} = \frac{\sum_{i=1}^N r_{ik}}{\frac{2\sigma^2_{u|k}\sigma^2_{v|k}}{\sigma^2_{u,v|k}} - 2}, \tag{A15}$$

and

$$\hat{\nu}_{u|v,k} = -\frac{\sum_{i=1}^N r_{ik}}{2\sigma^2_{u|v,k}}. \tag{A16}$$

Both empirical priors also penalize variables with large variances or with low evidence enforcing lower $w_{u|v,k}$ and higher $\sigma^2_{u|v,k}$, respectively.