Online Supplement to "Bayesian Modeling of Shared Gene Function"

P. Sykacek [a], R. Clarkson [b], C. Print, [c], R. Furlong [d] G. Micklem[e,f]
Department of Biotechnology, BOKU University, Vienna [a],
School of Biosciences, Cardiff University[b],
Department of Molecular Medicine & Pathology,
University of Auckland[c],
Department of Pathology [d], Department of Genetics [e]
and Cambridge Computational Biology Institute,
Department of Applied Mathematics and Physics [f],
University of Cambridge
Email: peter@sykacek.net

February 14, 2007

1 Introduction

This document is an online supplement to (Sykacek et al., 2007), a paper which proposes a fully Bayesian approach to a *shared* analysis of gene function across several microarray experiments. The approach assumes that several microarray experiment with known cross annotations between transcripts (genes) should be analyzed for common genetic causes. The implementation described in this work has in particular the advantage to combine data sets *before* applying any thresholds. We also provide means to diagnose results that are quite sensitive to hyper-parameter settings. Avoiding such sensitivity is imperative for obtaining a reliable ordering with respect to shared gene function. This online supplements provides additional information about the experiment, completes the calculation of the negative free energy and provides insight into implementation issues that were for brevity left out from the original paper. We also provide a brief tutorial on how to use the MatLab package we provide online at `http://www.sykacek.net/research.html#mcabf`.

2 Experimental Supplement

2.1 Synthetic Data

Synthetic data serves two purpouses. We illustrate the working principle of Bayesian modelling of shared gene function (Sykacek et al., 2007) and compare the proposed approach with a simple meta analysis for shared gene function. The latter approach uses p-values obtained from suitably chosen contrasts (e.g. a t-test or an ANOVA) to select differentially expressed genes and searches the lists obtained from different assays for common genes. Suach analysis is for example used by (Hockley et al., 2006). There are two aspects we want to highlight. Unlike the filtering

approach in (Hockley et al., 2006) which can oly reveal an unordered list of shared genes, the proposed Bayesian approach ranks genes with respect to shared gene function. In addition, all meta analyses which are based on thresholded gene lists are likely to suffer a censoring effect. This is also avoided by the proposed Bayesian method. Censoring is caused by combining after thresholding, which implicitly assumes that genes which are not selected in all assays do not provide any information about the biological cause under investigation. This is however a dangerous simplification of the actual implication of p-values, which just assess whether a certain observation can be explained by chance. A gene which is top ranked in one assay and has with any $\epsilon > 0$ in the secodn assay a p-value of $0.01 + \epsilon$ is potentially one of the most relevant common factors. A simple meta analysis will however reject the gene if we use a p-value threshold of 0.01.

All synthetic data is generated by drawing "synthetic genes" from Gaussian distributions with unit standard deviation and means as specified in table 1. We assume two assays and draw for each of 2 classes 3 groups of identically distributed variables. The ranking capabilities are illustrated drawing data according to the specification for experiment "Ordered Separability". The data for the investigation of censoring is specified under the label "Set Dependency". We

Table 1: Synthetic Data Generation

Experiment	Group	Mean Assay 1	Mean Assay 2
Ordered separability	1	± 2	± 2
	2	± 0.5	± 0.5
	3	± 0.05	± 0.05
Set dependency	1	± 2	± 2
	2	± 4	± 0.5
	3	± 0.1	± 0.1

generated for every group 4 synthetic genes, each replicated 6 times. The sign before the mean indicates dependancy on the class label. For one class we draw with positive means, for the other we draw using the negative means. The Bayesian probabilities of shared gene function obtained with this data are reported in the original paper (Sykacek et al., 2007). These results illustrate both aspects discussed above. We report in table 2 the results of a simple analysis for common genes which is based on pattern matching of thresholded gene lists. Each assay was separately analysed for differential expression using FSPMA (Sykacek et al., 2005). Shared genes are those which appear in both lists after thresholding. Note that the combined gene lists in table 2 are ordered alphabetically since rank infromation is lost. Experiment "Set dependency" furthermore lacks 3 genes from group 2 (genes 5, 7 and 8) since they do not show up as significant in the second assay. Note that the inclusion of gene 6 is pure chance. t could have been any other gene from group 2 or no gene at all. It is thus obvious from table 2 that simple approaches for shared analysis like (Hockley et al., 2006) do not provide rank information about shared gene function and that they can miss functionally important genes. The results in (Sykacek et al., 2007) show that the proposed Bayesian approch does not suffer these drawbacks.

Our second synthetic experiment illustrates the behaviour of the hierarchical prior over

Table 2: Classical Meta Analysis

Assay 1 (ranked)	Assay 2 (ranked)	Combined (alphabetically)
	Ordered separability	
gene_3	gene_1	gene_1
gene_1	gene_4	gene_2
gene_4	gene_2	gene_3
gene_2	gene_3	gene_4
gene_8		
	Set dependency	
gene_7	gene_3	gene_1
gene_8	gene_2	gene_2
gene_5	gene_4	gene_3
gene_6	gene_1	gene_4
gene_2	gene_6	gene_6
gene_1		
gene_3		
gene_4		

$\boldsymbol{\beta}_{t,s}$. As is discussed in (Sykacek et al., 2007), the hierarchical aspect is obtained via a diagonal Gaussian prior $p(\boldsymbol{\beta}_{t,s}|\boldsymbol{\Lambda}_s)$ where the precision matrix $\boldsymbol{\Lambda}_s$ is specified indirectly with the hyper prior $p(\boldsymbol{\Lambda}_s[d,d]|g_s,h_s)$. We expect that this setting decreases the sensitivity of the posterior distributions (most importantly all $Q(I_t)$) to the choice of hyperparameters. The motovation behind this assumption is that the prior over regression parameters $p(\boldsymbol{\beta}_{t,s}|\boldsymbol{\Lambda}_s)$ adjusts to fit the data optimally. To illustrate this adjustment, we use two synthetically generated datasets and infer the marginal distributions of the correpsonding probabilistic model (a DAG similar to Figure 1 in (Sykacek et al., 2007), however with only one "system" s).

Both datasets are again two class problems and contain three groups of variables with data generated from Gaussian distributions with unit standatd deviation. The first two groups contain fifty variables (i.e. synthetic genes) each. For the synthetic genes in first and second group we drew for both datasets 6 replicates from Gaussians with class label dependant means ± 4 and 0 respectively. The third group contains 200 variables. For the first dataset these were drawn using means ± 0.2. For the second dataset they were drawn using mean 0. Having inferred separate probabilistic models for both datasets, we obtained for the first data an expected variance $1/ < \boldsymbol{\Lambda}[2,2] >_{Q(\boldsymbol{\Lambda})} = 13.97$ and the indicator probabilities in the first graph in figure 1. For the second dataset we obtained the expected variance $1/ < \boldsymbol{\Lambda}[2,2] >_{Q(\boldsymbol{\Lambda})} = 34.21$ and the indicator probabilities in the second graph. To reason about whay we get larger prior variance in the second case, we should point out that the Gaussian prior acts like a regularization term. The 200 variables in the third group result in small probabilities for class in the first assay but are irrelevant in the second assay. Taking expectations over all genes, where we need to consider the $Q(I_t)$, we have thus in the second assay on average larger regression parameters and the Gaussian prior adjusts for that. The third graph illustrates the implication of the automatic adjustment of $\boldsymbol{\Lambda}$ on the measure of gene importance. If we take the sample limit to ∞, the synthetic genes in

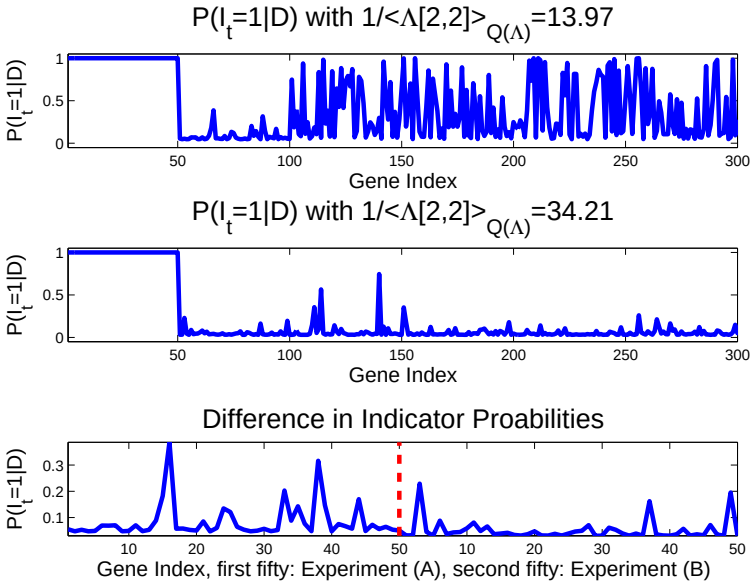

Figure 1: Indicator probabilities of synthetic gene function. The differences in $Q(\mathbf{\Lambda})$ are a result of the 200 variables in group 3. In the first experiment these variables show minimal class separability. In the second experiment they show no class separability. The expectation of the norm of the regression parameters for the model which predicts class labels using inputs (i.e. $I_t = 1$) is thus larger in the second experiment. This leads in the second experiment in turn to a larger penalty for the more complicated model and to smaller indicator probabilities $Q(I_t = 1)$, which is best seen in the third graph which compares the probabilities in the second group of variables for both experiments.

group 2 show for both experiments no class separability. Despite that these genes are identically distributed, the indicator probabilities of gene function, $Q(I_t = 1)$, are in the second experiment on average smaller (second 50 samples in the thrid graph). This is in line with findings about the Occams razor principle inherent to Bayesian inference (Jefferys and Berger, 1992). Let us consider a comparison of two models with different complexity and leave all aspects of modelling identcal except for a change in prior uncertainty. The penalty for the more complex model will in this case be harder for the more uncertain prior. From a model assessment point of view, this will in agreement with the third graph in figure 1, lead to smaller probabilities in favour of the more complex model.

This experiment illustrates that the hierarchical model will adjust the Gaussian prior $p(\boldsymbol{\beta}_{t,s}|\mathbf{\Lambda}_s)$, to the observed norm of regression parameters. If a dataset requires larger regression parameters the prior penalty decreases. This has an implication on the assessment of gene function, because we have an impicit adjustment to the situations found in every particular dataset. As such, this is of great advantage since it frees us from complex inter expeiment normalisation.

Table 3: Lactation and Involution and expected biological processes

biol. state	L_0	L_5	L_{10}	I_{12}	I_{24}	I_{48}	I_{72}	I_{96}
Type 1 Apoptosis	-	-	-	+	+	?	-	-
Type 2 Apoptosis	-	-	-	-	-	?	+	+
Apoptosis	-	-	-	+	+	+	+	+
Differentiation	+	+	+	?	-	-	-	-
Inflammation	?	-	-	+	+	?	-	-
Remodeling	-(?)	-	-	-	-	?	+	+
Acute Phase	+	-	-	-	+	+	+	+

Table 4: Biological processes in Endothelial Cells in Response to Serum Deprivation

biol. state	t_0	t_{28}
Type 2 Apoptosis	-	+
Apoptosis	-	+
Differentiation	+	-

2.2 Biological Macro States and Processes

2.2.1 Mammary Gland Development Time Course

Table 3 provides an assessment of the biological processes, we expect to find at different developmental stages during the lactation and involution periods of the mouse mammary gland. The time frame is days for the lactation and hours during the involution state. This data is one of the sets we want to assess for shared gene function. Pluses indicate that processes are active. Minuses indicate that processes are inactive and question marks are for points where we don't know the activity of the state.

2.2.2 Serum Deprived Endothelial cells

Table 4 lists the biological processes we expect to find at different durations after serum deprivation is initiated. (duration in hours)

2.3 Sensitivity of $Q(\Lambda_s)$ to Prior Choices

An important aspect of quoting a probability measure of gene function is that the measure should not depend crucially on the chosen prior, in particular if the influence is indirect. To achieve this, we specify the priors $P(I_t|\pi)$ and $p(\boldsymbol{\beta}_{t,s}|\boldsymbol{\Lambda}_s)$ hierarchically. In particular the situation with $\boldsymbol{\Lambda}_s$ is more subtle, since our choice will have an indirect effect on the measure of gene function, $Q(I_t)$, and we do not know the scale of the regression parameters. It is thus imperative to study the sensitivity of the model with respect to choices of $p(\boldsymbol{\Lambda}_s[d,d]|g_s,h_s)$ and check the effect on the approximate posterior over $\boldsymbol{\Lambda}_s[d,d]$ and the $Q(I_t)$ we get as a result. The theory of reference priors (Bernardo and Smith, 1994) suggests that varying the variance of the

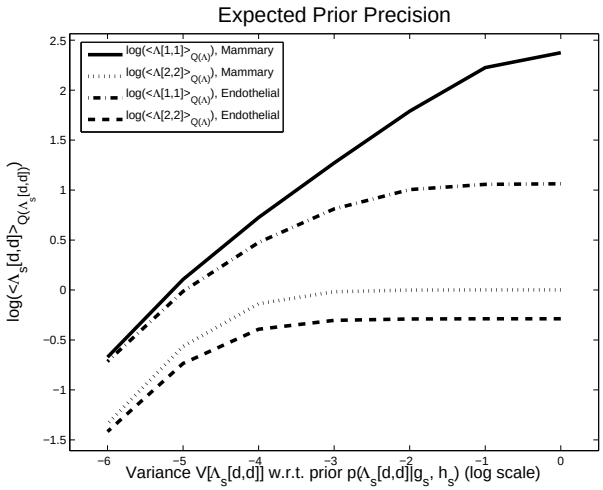

Figure 2: Dependency of the expectations of $\mathbf{\Lambda}_s[d,d]$ w. r. t. $Q(\mathbf{\Lambda}_s)$ on the hyperparameters g_s and h_s. We show the expected precisions of the Gaussian prior separately for each data set and coefficient.

Gamma prior is sufficient, since its expectation does not effect the posterior, if the variance is large enough. We thus fix the expectation of $E[\mathbf{\Lambda}_s[d,d]]$ w.r.t. the prior at 0.01 and vary the variance $V[\mathbf{\Lambda}_s[d,d]]$ linear on a log scale from 10^{-6} to 1. Translated to hyperparameters, we use $g_s = \{10^{-5}, 10^{-4}, ..., 10^2\}$ and $h_s = \{10^{-3}, 10^{-2}, ..., 10^4\}$. Figure 2 illustrates the dependency of the approximate posterior $Q(\mathbf{\Lambda}_s)$ on the hyperparameters h_s and g_s. In a setup, where we compare two different dimensional models, the probability measure $Q(I_t)$ will be mainly sensitive on variations of $\mathbf{\Lambda}_s[d,d]$, for all d that occur only in the higher dimensional model. In our case this suggests to monitor the expectations of $\mathbf{\Lambda}_s[2,2]$. From figure 2 we may thus conclude that an appropriate prior over $\mathbf{\Lambda}_s$ should have a variance larger or equal to 10^{-3}. In that range, the expectation of $\mathbf{\Lambda}_s[2,2]$ with respect to the approximate marginal $Q(\mathbf{\Lambda}_s)$ does not change any more. For any smaller variance, the prior $p(\mathbf{\Lambda}_s|g_s, h_s)$ has an undesired effect on the probability measures $Q(I_t)$.

2.4 Shared Analysis

From table 3, it is clear that typical biological state changes like from lactation to involution, will provide marker genes that are responsible for all processes that change activity synchronized with the state change. In situations, where only few processes overlap, we expect the combined gene list to be shorter and thus easier to comprehend and analyze by follow up studies than the individual lists. As an example, we analyze lactation vs. involution periods in mammary gland development together with control vs. two time points under serum withdraw in a human endothelial cell line. The original paper (Sykacek et al., 2007) illustrates the importance of a sensitivity analysis in detail. If we make sure that all hyper parameters are initialized such that all posteriors are dominated by the data, we find 2214 genes more likely than the intercept only model. Table 5 lists 20 of the top ranked genes. It shows gene symbols, the indicator probabilities ,$P(I_t|\mathcal{D})$, the overall gene measure, $P(G = t|\mathcal{D})$, which assesses genes relative to each other and finally a co-regulation indicator. We assess co-regulation if the

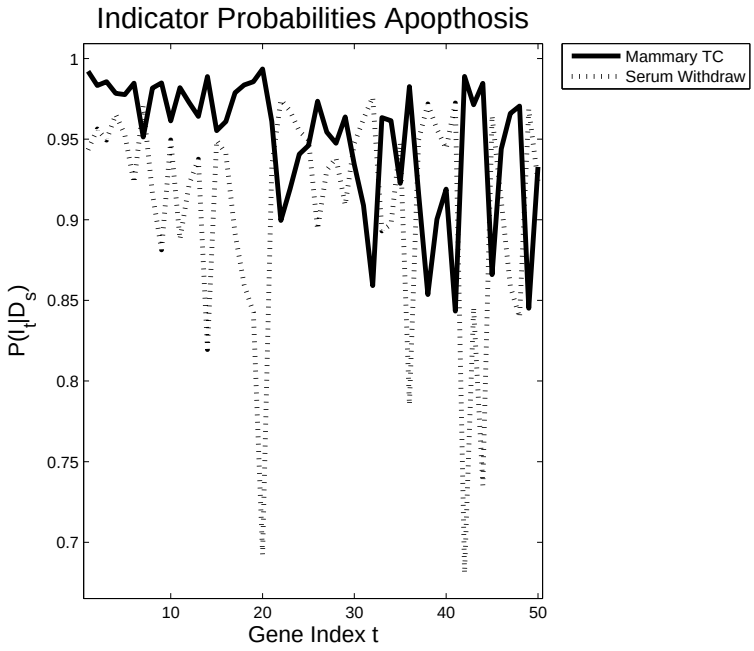

Figure 3: This figure illustrates system specific indicator probabilities, $P(I_t|D_s)$, for the, ranked according to shared gene function, top fifty genes.

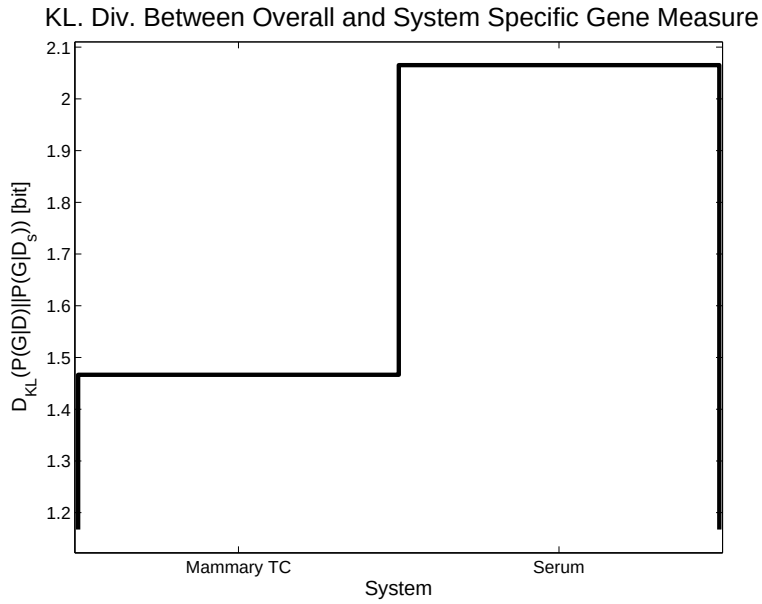

Figure 4: This figure illustrates the Kullback Leibler Divergence between the overall gene measure, $P(G = t|\mathcal{D})$, and the system specific gene measure, $P(G = t|\mathcal{D}_s)$, for the mammary development cycle and both endothelial responses. "Mammary TC" represents genomic activity in the development cycle of the mammary gland, "Serum" represents the genetic response of endothelial cells to serum withdraw.

regression coefficients that correspond to the log expression values have in all experiments the sign. The full gene list, http://www.sykacek.net/suppdata/mammaryendo.zip, is available online as zipped comma separated file (mammaryendo.csv). The online rank file provides in

Table 5: Top 20 genes found in a shared analysis of a mammary gland profile and an endothelial cell line under serum withdraw.

Gene Symbol	$P(I_t\|\mathcal{D})$	$P(G=t\|\mathcal{D})$	Co-Regulation
SAT	0.99951	0.047597	anti
ODC1	0.99921	0.029237	co
GRN	0.99921	0.029125	co
BSCL2	0.99919	0.028601	anti
MLF2	0.99884	0.019988	anti
IFRD2	0.99867	0.017425	co
BTG2	0.99843	0.014688	co
CCNG2	0.99826	0.013274	co
TNK2	0.99789	0.010943	anti
C9orf10	0.99783	0.010614	co
HAGH	0.99764	0.0097747	co
PPP2CB	0.99759	0.0095567	anti
SSR1	0.99748	0.0091528	co
MUT	0.99747	0.0091039	co
DHRS3	0.99746	0.0090926	co
PSMA1	0.99741	0.0089018	anti
HBLD2	0.99732	0.0086073	co
SYPL1	0.99724	0.0083639	co
C2F	0.99723	0.0083374	co
ATP6V1B2	0.99706	0.0078419	anti

addition two columns that illustrate the system specific indicator probabilities. This allows us to inspect how much the individual experiment supports individual genes. An illustration for the first 50 genes is provided in figure 3. The overall amount of information which is provided by each experiment is most easily visualized by the Kullback Leibler divergence between the overall gene measure $P(G = t|\mathcal{D})$ and the system specific measures $P(G = t|\mathcal{D}_s)$ (s is the system index). This analysis will provide one distance for each microarray. An illustration in figure 4 shows that the gene measure we obtain from the mammary gland development cycle is slightly closer than the measure we get from the endothelial cells. The overall gene measure and thus the indicator probabilities $P(I_t|\mathcal{D})$ do "depend" more on the mammary gland profile. The probabilities we obtain with the proposed approach will always combine data sets according to the degree of information they provide about individual genes. In this context the number of samples and the signal to noise ratio matter. By the latter we refer to the degree of overlap between lactation and involution on one hand and controls and samples under serum withdraw on the other hand.

2.4.1 Consistency Evaluations

Table 6: Generalization Results From Gene Measure (Table 5) Based Predictions

Experimental Condition	Generalization Accuracy
Mammary Gland (lactation / involution)	100 %
Endothelial Cells (control / serum withdraw)	100 %

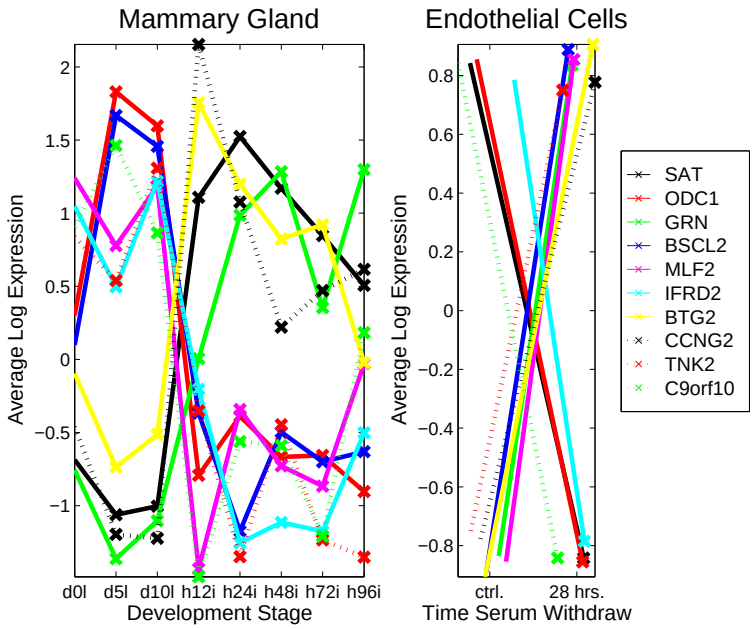

Figure 5: The left subplot illustrates average log expression values during mammary gland development. Illustrations are ordered with respect to the developmental cycle: virgin, days five and ten gestation, day 15 pregnancy, days one to ten lactation and twelve to 96 hours into involution. The right subplot illustrates average log expression values of an Endothelial cell line under growh factor withdrawal. We have controls and samples after 28 hours under growh factor withdrawal. To avoid that the average log expression traces of the Endothelial cell data overlap, we randomized the x-positions slightly. All starting points do thus belong to controls and all end points to samples that were for 28 hours under growth factor withdrawal.

We do already provide several evaluations in the full paper which support the derived gene measure. First we provided there the generalization accuracies of a 10 fold cross testing run[1]. These accuracy estimates are shown in table 6. All individual models are combined by using $P(G = t|\mathcal{D}_{fold})$ as weights for the predictions obtained from the t-th transcript. To some extent, high generalization accuracies do thus confirm the gene measure. A further sanity

[1]To obtain an unbiased estimate of generalization accuracy, we leave about a 10-th of the samples as independent test set and use the remaining samples for model building.

check can be obtained by looking at the average log expression values we find for top ranked genes at samples that were used to infer the gene measure. We provide this illustration in figure 5. Having derived a gene measure, we can follow (Al-Shahrour et al., 2004) and obtain a biological sanity check. We use all genes with known GO annotation to calculate the most likely change in biological process activity associated with the state change that lead to the gene measure. A set of sub trees of all active go categories which maintains the relations between GO categories as implied by the GO biological process DAG (directed acyclic graph) is provided at `http://www.sykacek.net/suppdata/gofishtree.xml` as xml file which is compatible with Treemap - (C) University of Maryland. Note that the treeml.dtd file is part of the Treemap package and not part of our software collection! Treemap is under a non commercial license. If it is unavailable despite that, the xml file can be inspected with a web browser.

3 Technical Supplement

The technical supplement will first provide the details of the derivation of the negative free energy and a discussion of implementation issues, that were, for reasons of brevity, removed from the original paper, (Sykacek et al., 2007). We will furthermore provide a brief tutorial on how to use the accompanying software package.

3.1 Measures of Shared Gene Function

We use variational inference of the model in Figure 1 in (Sykacek et al., 2007) to provide with $Q(I_t)$ an approximate measure of shared gene function for individual transcripts. As we model the predictions for every transcript by a two component mixture of a GLM with regressors depending on that transcript and a common alternative, we can convert the T measures $Q(I_t)$ into one measure on a T-dimensional ordinal variable G

$$P(G \equiv t|\mathcal{D}) \approx \frac{Q(I_t \equiv 1)}{Q(I_t \equiv 0)} \Big/ \Big(\sum_{k=1}^{T} \frac{Q(I_k \equiv 1)}{Q(I_k \equiv 0)} \Big). \tag{1}$$

If we assume that one of the transcripts is the "true" model, this measure quantifies according to (Bernardo and Smith, 1994), how probable this hypothesis is. Sometimes we are in addition interested how much individual experiments agree with the combined measure. We may quantify that for individual transcripts by calculating the measure $Q_s(I_t)$ which depends on data set s only. Along the lines of Equation (1), the $Q_s(I_t)$ can be converted to a corresponding$P(G|\mathcal{D}_s)$.

3.2 Negative Free Energy

Unlike typical approaches (e.g. (Attias, 1999)), where the negative free energy of the overall model is used for model assessment, it is of lesser importance here. In our implementation inference of $Q(I_t)$ takes over the role as "model selection yard stick" and Equation (2) is important to assess convergence of the algorithm and to diagnose errors in the mathematical derivation and implementation problems. For brevity, the original paper omitted these equation details.

The interested reader finds them here instead.

$$F(Q) = \sum_t F(Q(I_t)) - D_{KL}(p(\pi|\delta)||Q(\pi)) \\ - \sum_s D_{KL}(p(\mathbf{\Lambda}_s|g_s, h_s)||Q(\mathbf{\Lambda}_s)) \tag{2}$$

The first term of the negative free energy $F(Q)$ is the data dependent part and a sum over all functionals $F(Q(I_t))$. The second and third term are a summation of the negative Kullback-Leibler (KL) divergences between the prior and the approximate marginal distributions over π and all $\mathbf{\Lambda}_s$.

3.2.1 The sum over all $F(Q(I_t))$

simplifies after maximizing w.r.t. all $Q(I_t)$ to

$$\sum_t F(Q(I_t)) = \sum_t \log\left(\sum_{I_t} \exp(f_{I_t})\right), \tag{3}$$

with

$$\begin{aligned} f_{I_t} = \psi(\hat{\delta}_{I_t}) - \psi(\hat{\delta}) + \sum_s \Bigg(& -\frac{1}{2}\log|\hat{\mathbf{\Lambda}}_{I_t,t,s}| \\ & + \frac{1}{2} d_{I_t,t,s} + \frac{1}{2}\sum_{d=1}^{d_{I_t,s}} \Big(\psi(\hat{g}_{d,s}) - \log(\hat{h}_{d,s})\Big) \\ & - \frac{1}{2}\Big(\hat{\boldsymbol{\beta}}^T_{I_t,t,s} \langle \mathbf{\Lambda}_{I_t,s}\rangle \hat{\boldsymbol{\beta}}_{I_t,t,s} + \text{tr}\ \hat{\mathbf{\Lambda}}^{-1}_{I_t,t,s} \langle \mathbf{\Lambda}_{I_t,s}\rangle\Big) \\ & + \sum_n \Big(\log\left(\Phi(b_{n,s}) - \Phi(a_{n,s})\right) \\ & \quad - \frac{1}{2}\gamma \boldsymbol{x}^T_{I_t,n,t,s} \hat{\mathbf{\Lambda}}^{-1}_{I_t,t,s} \boldsymbol{x}_{I_t,n,t,s}\Big)\Bigg). \end{aligned} \tag{4}$$

3.2.2 KL divergence $D_{KL}(p(\pi|\delta)||Q(\pi))$

$$\begin{aligned} D_{KL}(p(\pi|\delta)||Q(\pi)) = & -\log\left(\frac{\Gamma(\delta_1+\delta_2)}{\Gamma(\hat{\delta}_1+\hat{\delta}_2)}\right) \\ & + \log\left(\frac{\log(\Gamma(\hat{\delta}_1)\Gamma(\hat{\delta}_2))}{\log(\Gamma(\delta_1)\Gamma(\delta_2))}\right) \\ & - \Big(\psi(\hat{\delta}_1) - \psi(\hat{\delta}_1+\hat{\delta}_2)\Big)(\delta_1 - \hat{\delta}_1) \\ & - \Big(\psi(\hat{\delta}_2) - \psi(\hat{\delta}_1+\hat{\delta}_2)\Big)(\delta_2 - \hat{\delta}_2) \end{aligned} \tag{5}$$

3.2.3 KL divergence $D_{KL}(p(\Lambda_s|g_s,h_s)||Q(\Lambda_s))$

$$D_{KL}(p(\boldsymbol{\Lambda}_s|g_s,h_s)||Q(\boldsymbol{\Lambda}_s)) = -\sum_d \Bigg(g_s \log(h_s) \\ -\hat{g}_{d,s}\log(\hat{h}_{d,s}) + \log\left(\frac{\Gamma(\hat{g}_{d,s})}{\Gamma(g_s)}\right) + \frac{\hat{g}_{d,s}}{\hat{h}_{d,s}} \\ \times(\hat{h}_{d,s}-h_s) + (g_s-\hat{g}_{d,s})\big(\psi(\hat{g}_{d,s}) - \log(\hat{h}_{d,s})\big)\Bigg) \tag{6}$$

3.3 Model Predictions

Model predictions are required in a diagnosis setting if we wish to predict the unknown classification of a microarray sample, or to obtain generalization accuracies. In the setting proposed here, predictions may be obtained by appropriately combining the predictions of individual transcripts to get $P(y_{m,s}|\boldsymbol{x}_{m,s},\mathcal{D})$ as expectation of the unknown label $y_{m,s}$, with m denoting the sample index and s the experiment. Predictions are conditional on $\mathcal{D}$, the data used for inference and on $\boldsymbol{x}_{m,s}$, the expression values of all transcripts.

$$P(y_{m,s}|\boldsymbol{x}_{m,s},\mathcal{D}) = \sum_G \int_{\boldsymbol{\beta}_s} \Big(P(y_{m,s}|\boldsymbol{x}_{m,s},\boldsymbol{\beta}_s,G) \\ \times p(\boldsymbol{\beta}_s|G,\mathcal{D})P(G|\mathcal{D})\Big) d\boldsymbol{\beta}_s \tag{7}$$

Integration with respect to $\boldsymbol{\beta}_s$ can be done analytically

$$P(y_{m,s}|\boldsymbol{x}_{m,s},\mathcal{D}) = \sum_t \Big(P(G\equiv t|\mathcal{D}) \\ \times P(y_{m,s}|\boldsymbol{x}_{m,s},G\equiv t,\mathcal{D})\Big) \tag{8}$$

where

$P(y_{m,s}\equiv 0|\boldsymbol{x}_{m,s},G\equiv t,\mathcal{D}) = \Phi(0;\hat{z}_{m,t,s},\lambda_{m,t,s})$

$P(y_{m,s}\equiv 1|\boldsymbol{x}_{m,s},G\equiv t,\mathcal{D}) = 1-\Phi(0;\hat{z}_{m,t,s},\lambda_{m,t,s})$

$\hat{z}_{m,t,s} = \boldsymbol{x}^T_{I_t\equiv 1,m,t,s}\hat{\boldsymbol{\beta}}_{I_t\equiv 1,t,s}$

$\lambda_{m,t,s} = \gamma\dfrac{1}{1+\gamma\boldsymbol{x}^T_{I_t\equiv 1,m,t,s}\hat{\boldsymbol{\Lambda}}^{-1}_{I_t\equiv 1,t,s}\boldsymbol{x}_{I_t\equiv 1,m,t,s}}$,

$P(G\equiv t|\mathcal{D})$ is the probability measure from Equation (1), $\Phi(0;\hat{z}_{m,t,s},\lambda_{m,t,s})$ a Gaussian cdf at 0, using $\hat{z}_{m,t,s}$ as mean and $\lambda_{m,t,s}$ as precision. We also have $\hat{\boldsymbol{\beta}}_{I_t\equiv 1,t,s}$ as mean and $\hat{\boldsymbol{\Lambda}}_{I_t\equiv 1,t,s}$ as precision of $Q(\boldsymbol{\beta}_{I_t\equiv 1,t,s})$ and use index $I_t\equiv 1$ to denote that these predictions arise from the GLM that uses the expression values of the corresponding transcript.

3.4 Computing Shared Gene Function

Variational inference of shared gene function will iterate all previously derived maximization steps of the Q-distributions in Equation (3) found in (Sykacek et al., 2007). Pseudo code that

Algorithm 1 *Inference of shared gene function*

initialize($Q(\pi)$)
$\forall\ s$ initialize($Q(\mathbf{\Lambda}_s)$)
$\forall\ t$ initialize($Q(I_t)$)
$\forall\ t, s$ initialize($Q(\boldsymbol{\beta}_{t,s})$)
$\forall\ n, t, s$ initialize($Q(z_{n,t,s})$)
$F = F(Q)$ % according to Eqn. (10)
$j = 0$
REPEAT
  inc(j)
  $\forall\ n, t, s$ update($Q(z_{n,t,s})$) % according to Eqn. (4)
  $\forall\ t, s$ update($Q(\boldsymbol{\beta}_{t,s})$) % according to Eqn. (5)
  update($Q(\pi)$) % according to Eqn. (6)
  $\forall\ s$ update($Q(\mathbf{\Lambda}_s)$) % according to Eqn. (7)
  $\forall\ t$ update($Q(I_t)$) % according to Eqn. (8)
  $Fn = F(Q)$ % according to Eqn. (10)
  Fd=Fn-F % to monitor convergence
  F=Fn
UNTIL ($j \equiv$ maxit or $Fd <$ maxeps)
$\forall\ t$ calculate($P(G = t|\mathcal{D})$) % acording to Eqn. (11)

illustrates these updates is shown in Algorithm 1. All equation numbers in the algorithm refer to (Sykacek et al., 2007). The evaluation of the negative free energy can also be found in this supplement as Equation (2). The result of the iterative procedure is a closed form approximation of the posterior and a probability measure over a T dimensional ordinal variable G. It is advisable to use Algorithm 1 and predictions based on Equation (13) from (Sykacek et al., 2007) to obtain cross testing results for all experiments that contribute to the analysis. This will provide generalization errors and in addition a set of gene measures, each obtained from slightly perturbed data sets as they arise from estimating the generalization error. Lack of small generalization error indicates that the measurements are not informative about the biological process and one should be cautious about assessing gene function. The data dependent variability of the gene measure provides a range in which we should assess genes identical. Large differences between individual rankings suggest we need to use a larger assay.

3.5 Using the Software

The software to calculate indicator probabilities that capture shared gene function comes as collection of MatLab libraries. The package consists of the main code, which uses the variational Bayesian approach described in (Sykacek et al., 2007) and additional functions for data handling, output generation and an EM implementation for regularized probit link regression used during initialization of all Q-distributions. To make the software distribution flexible, all MatLab functions are collected in archives each containing functions of a particular type.

Table 7: Library Files for Shared Analysis (Zip files ending with .zip instead of .tar.gz are available as well)

Library File	Description
helpers.tar.gz	generic helper functions
statsgen.tar.gz	generic statistics functions
mca_base.tar.gz	basic microarray file handling (loading various microarray data formats)
mca_fuse.tar.gz	generic handling in connection with shared analysis (cross annotation and output generation)
probitem.tar.gz	Penalized maximum likelihood (MAP) for probit link regression via an EM algorithm.
combanalysis.glb.hphp.tar.gz	Variational Bayes for shared analysis of subset probabilities in probit regression.

3.5.1 Installation

To install the package, one has to download all required archives, provided as *.zip files or tared gzip archives (*.tar.gz), unpack the archives and set appropriate MatLab paths. Scripts using a hypothetical experiment derived from a mouse testis time course kindly provided by R. Furlong, demonstrate how to use the library. All components required to successfully run the experiment, will be installed automatically, if one creates a new directory and then downloads and runs the setup script in that directory. Linux (Unix) users should use combsyssetup.sh. Windows users should either do the same after installing a cygwin environment or install the Wget and Unzip packages from GnuWin32 and then download and run combsyssetup.bat. Note that this will install all required packages and, if run at later times, install updates.

3.5.2 Files required for an analysis

After having run the script, the installation directory contains MatLab scripts and data which illustrate how to use the approach discussed in (Sykacek et al., 2007). The data are extracted from a subset of a mouse testis development time course, kindly provided by R. Furlong. The data consists of 7 time points: adult day 1 day 5 day 10 day 15 day 23 and day 35, with differential expression measured against the adult generation. To illustrate all steps from cross annotation to generation of gene lists, we divided this data artificially into two "experiments". One experiment contains the samples of the adult generation and days 1 and 15. Here we use the original gene ids. We assume that the biological state change is between adult and the other two development periods. The corresponding data file is called "exp1mca.tsv" and is formatted like FSPMA normalized raw output (gene ids as column headers and all samples as rows below). We also have a corresponding effects description as "exp1eff.tsv", which is used to generate the labels. The second experiment contains days 35, 23, 5 and 10 and artificially modified gene ids (as to mimic a situation that requires between species annotation). Here we assume that the biological states correspond to days 35 and 23 versus days 5 and 10. The files are "exp2mca.tsv" for the microarray data and "exp2eff.tsv" for the labels. Note that the assumption is that each experiment provides information about differences in late and early

Table 8: Data Files for the tutorial

File Name	Description
exp1mca.tsv, exp2mca.tsv	Normalized log ratios (location and scale adjustment)
exp1eff.tsv, exp2eff.tsv	(default) Labels
crossann.tsv	cross annotations between the different gene ids found in the gene lists
genespec.csv	mapping from unique gene ids (for the cross annotation target) to standardized symbols and gene descriptions
shareanalysis.def	specification of the cross annotated experiments that enter shared analysis

Table 9: Script Files for the tutorial (Note: to be run in this order)

File Name	Description
crossann.m	cross annotation of microarray experiments and preparation of shared analysis
runsim.m, calccoreg.m	calculate gene indicator probabilities of shared gene function by variational Bayesian inference
combres2csv.m	extraction of gene ranking w.r.t. shared gene function as a tab delimited file

stages of testis development. The analysis goal is thus similar to a problem, where we attempt to combine two experiments obtained from different platforms or species. This requires "cross annotation", which is here done according to the tab delimited file "crossann.tsv". In general, each row in this file contains a tuple that provides a unique mapping between all different unique gene ids one finds in a shared analysis. To complete the list of files, we provide in addition the tab delimited file "genespec.csv", which provides for the unique gene ids in the target genome, a mapping to gene symbols and descriptions.

3.5.3 Analysis for Shared Gene Function

Both artificial experiments have to be cross annotated. This step will align the gene ids in different experiments and provide two raw data files and a gene id to symbols and description annotation in MatLab 6 format. Cross annotation is done by the MatLab script crossann.m found in the installation folder.

After cross annotation, we have to prepare the shared analysis. This requires to specify a tab delimited text file ("shareanalysis.def") which controls this process. The minimal requirement is to specify in this control file which (previously cross annotated) data files should be analyzed for shared gene function. In addition one can specify a different set of labels. This is useful to analyze the same data for different biological classifications. We may also provide

Table 10: Files generated form the tutorial code

File Name	Description
exp1.mat, exp2.mat	cross annotated and normalized raw data
crossanngenespec.mat	reordered gene specifications (id, symbol and description)
state.mat, crosslog.mat	internal log files (see code)
sharetestres.mat	inference result about shared gene function. This file contains all results including probabilities, predictions and all Q-distributions found from variational Bayesian inference.
share_test_rank.csv	rank table as comma separated file.

independent test data, which will calculate generalization errors. Analysis is stared with the script combanalysis.m in this folder. This simulation run will, depending on the size of the problem and the mode of analysis take up to several hours (this example is though done in less than one minute or in a few minutes if we want fold results). As a result we get all simulation output in MatLab format. Details of the calculated results require to look into the code and to analyze the variables stored in the in MatLab output.

The last step in an analysis of shared function is to generate a rank table of shared gene function. This is done with the script crossann2csv.m found in this folder as well. The result is a rank list of similar structure as the one provided in the supplement of the original paper.

All intermediate results generated during a shared analysis is stored in MatLab 6 format (for Octave compatibility). The final rank table is a comma separated file.

3.5.4 Shared Analysis of Different Experiments

To run such an analysis on a different experiment, one must provide data files structured like those listed in table 8. The structure of the microarray data and the default labels is identical to the output generated by FSPMA, which can thus be used as preprocessing tool. In addition, one has to generate a file which allows cross annotation between all gene sets that appear in any one data set. If there is only one set of gene ids, cross annotation should be done anyway, to obtain the data in the format expected by "runsim.m". In this case all parts in runsim.m that specify the shared analysis will refer to the same gene id column. Inference of shared gene function requires in addition a control file similar to "shareanalysis.def". Finally one has to adjust all script file in table 9 to meet the different requirements.

Acknowledgments

The authors want to thank David MacKay for his advice. This work was funded by the BBSRC's Exploiting Genomics initiative under ref. 8/EGH16106, "Shared Genetic Pathways in Cell Number Control". Peter Sykacek is currently supported by the WWTF, Baxter and ARCS and grateful for their support.

References

F. Al-Shahrour, R. Díaz-Uriarte, and J. Dopazo. FatiGO: A web tool for finding significant associations of Gene Ontology terms with groups of genes. *Bioinformatics*, 20:578–580, 2004.

H. Attias. Inferring parameters and structure of latent variable models by variational Bayes. In *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI–99)*, pages 21–30, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

J. M. Bernardo and A. F. M. Smith. *Bayesian Theory*. Wiley, Chichester, 1994.

S. L. Hockley, V. M. Arlt, D. Brewer, I. Giddings, and D. H. Phillips. Time- and concentration-dependent changes in gene expression induced by benzo(a)pyrene in two human cell lines, MCF-7 and HepG2. *BMC Genomics*, 7(260), October 2006.

W. Jefferys and J. Berger. Ockham's razor and Bayesian analysis. *American Scientist*, 80:64–72, 1992.

P. Sykacek, R. Furlong, and G. Micklem. A Friendly Statistics Package for Microarray Analysis. *Bioinformatics*, page to appear, 2005. URL: `http://bioinformatics.oxfordjournals.org/cgi/reprint/bti663?ijkey=UxcmV7ypaHrMDUN&keytype=ref`.

P. Sykacek, R. Clarkson, C. Print, R. Furlong, and G. Micklem. Bayesian Modeling of Shared Gene Function. Technical report, Department of Biotechnology, BOKU University, Vienna, 2007. [Available at `http://www.sykacek.net/pubs.html#sykacek_etal_TR071`].