Johns Hopkins University, Dept. of Biostatistics Working Papers

11-7-2006

PENALIZED LIKELIHOOD AND BAYESIAN METHODS FOR SPARSE CONTINGENCY TABLES: AN ANALYSIS OF ALTERNATIVE SPLICING IN FULL-LENGTH cDNA LIBRARIES

Corinne Dahinden
Seminar fur Statistik

Giovanni Parmigiani
The Sydney Kimmel Comprehensive Cancer Center, Johns Hopkins University & Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, gp@jimmy.harvard.edu

Mark C. Emerick
Department of Physiology, Johns Hopkins School of Medicine

Peter Buhlmann
Seminar fur Statistik

Penalized Likelihood and Bayesian Methods for Sparse Contingency Tables: An Analysis of Alternative Splicing in Full-Length cDNA Libraries

Corinne Dahinden
Seminar für Statistik
ETH Zürich
CH-8092 Zürich, Switzerland
email: dahinden@stat.math.ethz.ch

Giovanni Parmigiani
Departments of Oncology and Biostatistics,
Johns Hopkins Schools of Medicine and Public Health
Baltimore, MD
email: gp@jhu.edu

Mark C. Emerick
Department of Physiology,
Johns Hopkins School of Medicine
Baltimore, MD
email: memeri@jhmi.edu

Peter Bühlmann
Seminar für Statistik
ETH Zürich
CH-8092 Zürich, Switzerland
email: buhlmann@stat.math.ethz.ch

Summary. We develop methods to perform model selection and parameter estimation in log-linear models for the analysis of sparse contingency tables to study the interaction of two or more factors. Typically, datasets arising from so-called full-length cDNA libraries, in the context of alternatively spliced genes, lead to such sparse contingency tables. Maximum Likelihood estimation of log-linear model coefficients fails to work because of zero cell entries. Therefore new methods are required to estimate the coefficients and to perform model selection. Our suggestions include computationally efficient ℓ_1- penalization (Lasso-type) approaches as well as Bayesian methods using MCMC. We compare these procedures in a simulation study and we apply the proposed methods to full-length cDNA libraries, yielding valuable insight into the biological process of alternative splicing.

KEYWORDS: Graphical models, Hierarchical models, Interactions, Lasso, Log-linear models, Variable selection

1 Introduction

One of the most striking discoveries of the genomic era is the unexpectedly small number of genes in the human genome. This number has decreased from more than 100000 (Liang *et al.*, 2000) through 30000-35000 (Int. Consortium, 2001; Ewing and Green, 2000; Venter *et al.*, 2001) and is now estimated to be roughly between 20000 and 25000 (Int. Consortium, 2004; Southan, 2004), tens of thousands less than initially expected and essentially the same number as found in phenotypically much simpler organisms. Thus, a question of overriding biological significance is, how complex phenotypes of higher organisms arise from limited genomes. Part of the explanation may be that many genes undergo a process called alternative RNA splicing, which can generate many distinct proteins from a single gene.

RNA splicing is a post-transcriptional process that occurs prior to mRNA translation. After the gene has been transcribed into a pre-messenger RNA (pre-mRNA), it consists of intronic regions destined to be removed during pre-mRNA processing (RNA splicing), as well as exonic sequences that are retained within the mature mRNA. Occurring after transcription is the actual splicing process, during which it is decided which exons are retained in the mature message and which are targets for removal. This is modeled as a non-deterministic process where exons and introns are retained and deleted in different combinations to create a diverse array of mRNAs from a common coding sequence. This process is known as alternative RNA splicing. Mapping of large numbers of expressed sequence tags (ESTs) onto genomic DNA has revealed that many genes are alternatively spliced. Depending on the source, the percentage lies between 35% and 60% (Mironov *et al.*, 1999; Brett *et al.*, 2000; Int. Consortium, 2001; Brett *et al.*, 2002; Carninci et al., 2005; Zavolan, van Nimwegen, and Gaasterland, 2003; Imanishi *et al.*, 2004). However, the information which can be derived from ESTs as far as alternative splicing is concerned is limited for various reasons. One of these is transcript end bias resulting from the fact that ESTs are prevalently derived from sequencing the ends of cDNAs. And as ESTs are short in length (typically around 300-500bp), only a portion of the cDNA can be covered. This means that splice sites in the middle region of the gene are strongly underrepresented in EST libraries and therefore hard to detect by these means. One way to overcome this difficulty is by screening many full-length cDNAs. By recording the complete cDNA from a mature RNA for the same gene again and again, a full-length cDNA library, also known as single-gene library (SGL), builds up and detailed information about how specific exon combinations go together becomes available. The functional regions of the proteins are grouped in domains which in many cases correspond to a single exon which encodes these domains. For example a transcription factor consists of a DNA binding domain and a regulatory domain. Thus the alteration of the exon structure corresponds to an alteration in the function of this particular domain. The central premise is that correlated expressions of domains point to a functional association. If domains interact functionally then their splicing should be co-regulated. It is further believed that such interaction patterns are regulated in a tissue and development specific way.

As more investigators become interested in this type of information, and large-scale single-gene libraries become available, there is a strong need for reliable statistical methods for analyzing the resulting datasets. Due to the large number of potential combinations in highly alternatively spliced genes, any library will only comprise a small portion of the total theoretically possible

inventory of combinations. Statistically, this leads us to deal with sparse contingency tables in which dimensions represent exons and cells represent variants. Investigating interactions among exons in the formation of a message requires addressing a model selection problem that is challenging both inferentially and computationally.
Here, within the context of log-linear models, we develop different statistical methods to analyse sparse contingency tables, such as e.g. those arising from single-gene libraries. These methods are compared in a simulation study and then applied to full-length cDNA datasets. As far as these are concerned, the main focus lies in identifying the interaction structure, estimating the interaction strength, and assessing how the interaction structure varies over different tissues or stages of development in a single tissue. The analysis of these interaction patterns is a first step towards understanding the underlying regulatory program.
Section 2 is an introduction to contingency tables and log-linear models. In Section 3, we describe different frequentist and Bayesian model selection procedures for log-linear models. Detailed algorithms and implementations of these are given in Section 4. The summary of a simulation study is given in Section 5, and the proposed methods are applied to real single-gene libraries in Section 6. Sections 2 and 3 are presented in general terms, as the methodology developed there can be applied to a broad spectrum of problems.

2 Contingency Tables and Log-linear Models

2.1 General Methodology

In this section we provide general definitions and notations.
A contingency table is formed by classifying a number of objects according to a set C of criteria which correspond to categorical variables. The classified objects can be represented as the cell counts of a so-called $|C|$-way contingency table, where $|C|$ represents the number of elements in C. If we adopt the notation of Dellaportas and Forster (1999), which goes back to Darroch, Lauritzen, and Speed (1980), the table is the set $I = \prod_{c\in C} I_c$, where I_c is the set of levels of the factor c. An individual cell is denoted by $i = (i_c, c \in C)$ and the corresponding cell count by n_i. The total number of cells in the table is $m = |I| = \prod_{c\in C} |I_c|$.
A natural way of representing the distribution of the cell counts is via a vector of probabilities $\mathbf{p} = (p_i, i \in I)$. If a total number of n individuals is observed and the objects are classified independently, then the distribution of the corresponding cell counts $\mathbf{n} = (n_1, n_2, \ldots, n_m)^t$ is multinomial with probability $\mathbf{p}$. A general log-linear model represents $\mathbf{p}$ as $\log(\mathbf{p}) = \mathbf{X}\boldsymbol{\beta}$, where $\boldsymbol{\beta}$ is a vector of unknown regression coefficients. The choice of the design matrix $\mathbf{X}$ will be discussed below. A specific parametrization of the log-linear model is in terms of the "u parameters", introduced by Birch (1963), see for example Bishop, Fienberg, and Holland (1975). The resulting model is called a log-linear interaction model:

$$\log p_i = \sum_{a\subseteq C} u_a(i_a) \quad (i \in I), \tag{1}$$

where i_a is the marginal cell $i_a = (i_\gamma, \gamma \in a)$, indicating the levels of a subset a of C. Thus the vector $U_a = (u_a(i_a), i \in I)$ depends only on the corresponding cell i via the marginal cell i_a.

In matrix formulation, this corresponds to a matrix $\mathbf{X} = [X_a, a \subseteq C]$, where $X_\emptyset$ is a column of 1's (intercept) and $X_{c_1} \in \mathbb{R}^{m \times |I_{c_1}|}$ for $c_1 \in C$ is an incidence matrix where each row has a unit entry in the column of the level to which it belongs; $X_{c_1 c_2} := X_{c_1} : X_{c_2}$ is defined by taking each column of X_{c_1} and multiplying it element-wise by each column of X_{c_2}; $X_{c_1 c_2 c_3} = X_{c_1} : (X_{c_2} : X_{c3})$; and this can be generalized to any number of factors $a = \{c_1, \ldots, c_l\} \subseteq C$.
The model (1) and the corresponding matrix $\mathbf{X}$ are highly overparametrized. To ensure identifiability, we impose sum-to-zero constraints on u_a:

$$\sum_{\substack{i \in I \\ i_\gamma = const}} u_a(i_a) = 0 \quad \forall a \subseteq C, \forall \gamma \subset a, \forall i_\gamma \in I_\gamma, \tag{2}$$

while $u_\emptyset$ is a normalizing constant ensuring that all cell probabilities add up to 1. Equation (1) in vector formulation becomes

$$\log(\mathbf{p}) = \sum_{a \subseteq C} U_a. \tag{3}$$

One can prove that under the constraints (2) it holds that $U_a \perp U_b$ for $a \neq b$, i.e. $\sum_{i \in I} u_a(i_a) u_b(i_b) = 0$ (see Lemma 1 in the Appendix A for details). In matrix formulation, the constraints (2) impose constraints on the sub-matrices X_a of $\mathbf{X}$ ($a \subseteq C$) in the representation $\log(\mathbf{p}) = \mathbf{X}\boldsymbol{\beta}$: $X_b^t X_a = 0$ for $a \neq b$. If we reparametrize X_a by choosing orthonormal columns, it holds that X_a is an orthonormal basis of $span(U_a)$: X_a has dimensionality $\mathbb{R}^{|I| \times d_a}$, where $d_a = \prod_{\gamma \in a}(|I_\gamma| - 1)$. Imposing the constraints (2) on the design matrix $\boldsymbol{X}$ corresponds in terms of ANOVA to choosing a poly-contrast. The log-linear interaction model (1) or (3) with the constraints (2), takes on the following form in matrix formulation:

$$\log(\mathbf{p}) = \mathbf{X}\boldsymbol{\beta}. \tag{4}$$

The correspondence to (3) holds by using $U_a = X_a \beta_a$, where β_a is the part of the vector $\boldsymbol{\beta}$ corresponding to the interaction term a. In case of factors with only 2 levels, β_a is a scalar, otherwise it is a vector of dimension d_a. If one assumes a smaller model without some of the interaction terms, the model takes on the same form (4) with some columns removed from the design matrix $\mathbf{X}$.

2.2 Contingency Tables and Log-Linear Models for Binary Factors

Translating the formalism above to binary factors, the domain of our problem, is straightforward. The set $C = \{1, \ldots, D\}$ of criteria corresponds in our case to D factors with 2 levels 1/-1. These represent the D exons, which are either retained or deleted. The set I is the whole array of 2^D theoretically possible exon combinations. A single cell i of the contingency table can therefore be represented by a D-dimensional binary vector $(i_1, \ldots, i_D)$, with each i_j indicating whether the corresponding exon is present or absent. The corresponding log-linear interaction model (1) with the constraints (2) can be written in the following way:

$$\log p_i = \beta_\emptyset + \sum_{l \in \{1,\ldots,D\}} \beta_l i_l + \sum_{\substack{i,k \\ j<k \in \{1,\ldots,D\}}} \beta_{jk} i_j i_k + \ldots + \beta_{12\ldots D} i_1 i_2 \cdots i_D.$$

From this representation, one can straightforwardly derive the design matrix $\mathbf{X}$ and the parametrization (4).
The formulation above corresponds to the situation where we have D cassette exons. Cassette exons are segments of the DNA which are either spliced in or spliced out. We note here that the term *exon* throughout this work represents either a complete exon or an exon segment, as alternative splicing at times occurs within exon boundaries, resulting in inclusion of exon fragments in the mature transcript. The situation corresponds to D factors with two levels (spliced in or spliced out). The methodology in Section 2.1 is held very general so that it can also be applied to problems with more than two levels per factor. For example in the context of single-gene library analysis, if two exons are mutually exclusive, an appropriate representation for the pair is given by using a single factor with 3 levels.

3 Model Selection

In this section we introduce different model selection strategies in log-linear models. In Section 3.2 we develop first an ℓ_1-regularization model selection approach, which is then expanded to the new so-called *level*-ℓ_1-regularization approach in Section 3.3. We favor the latter over the former; see also the results of the simulation study in Section 5. In Sections 3.4 and 3.5, Bayesian model selection strategies are introduced.

3.1 Non-Hierarchical Versus Hierarchical Models

Hierarchical models are a subclass of models such that if an interaction term β_a is zero, than all higher order interaction terms β_b for $b \supseteq a$ are also zero. While it is possible that the true underlying interaction model may not be hierarchical from a biological standpoint, a difficulty in the use of non-hierarchical models arises from the fact that they are not invariant under reparametrization. We have chosen the design matrix $\boldsymbol{X}$ with sum-to-zero constraints on u_a (see (2)) to ensure identifiability, and we used a specific, namely an orthonormal basis of $span(U_a)$. In terms of ANOVA, this choice is equivalent to choosing a poly-contrast. We could have imposed different constraints or have chosen a different basis of $span(U_a)$, and this would have resulted in a different design matrix $\boldsymbol{X}$ or in terms of ANOVA, a different choice of contrast. Suppose we have found an interaction vector $\boldsymbol{\beta}$ for one parametrization of the log-linear model and that this vector corresponds to a non-hierarchical model, meaning there is at least one lower order interaction term β_a equal to zero, while $\beta_b \neq 0$ for at least one $b \supseteq a$. If we reparametrize the model, using a different design matrix, the coefficient for the model term a may not be zero anymore. On the other hand, by reparametrizing a hierarchical model, all zero terms remain zero after reparametrization. Therefore, hierarchicity is preserved after reparametrization while non-hierarchicity depends on the parametrization. This is a distinct advantage of working within the hierarchical class. In a hierarchical model, all zero coefficients can directly be interpreted in terms of conditional independence, while for non-hierarchical models, the zero terms of the hierarchized model ($\beta_a = 0$ with $\beta_b = 0\ \forall b \supseteq a$) feature this interpretation whereas lower order zero interaction terms may only be interpreted together

with the according parametrization.

3.2 ℓ_1-Regularized Model Selection

The Lasso, originally proposed by Tibshirani (1996) for linear regression, performs regularized parameter estimation and variable selection at the same time. It is defined as follows:

$$\widehat{\boldsymbol{\beta}}^{\lambda} = \arg\min_{\boldsymbol{\beta}} \left[\sum_i (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})_i^2 + \lambda \sum_i |\beta_i| \right],$$

where $\mathbf{Y} = (Y_1, \ldots, Y_n)$ is the response vector. It can also be viewed as a penalized Maximum Likelihood estimator, as $\sum_i (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})_i^2$ is proportional to the negative log-likelihood function for Gaussian linear regression. While the MLE for the general regression model is no longer uniquely defined and very poor in the case of more variables than observations, the Lasso estimator is still reasonable for $\lambda > 0$. For our analysis, we have a similar problem, namely that the MLE is not defined in case of zero counts in the contingency table: a detailed description of the existence of the MLE in general log-linear interaction models is given in Christensen (1991). Inspired by the Lasso, we estimate our parameter vector $\boldsymbol{\beta}$ by the following expression:

$$\widehat{\boldsymbol{\beta}}^{\lambda} = \arg\min_{\boldsymbol{\beta}} \left[-l(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{m} | \beta_j| \right], \tag{5}$$

where $l(\boldsymbol{\beta})$ is the log-likelihood function $l(\boldsymbol{\beta}) = \log \mathbb{P}_{\boldsymbol{\beta}}[\mathbf{n}] \propto \sum_j \mathbf{n}_j (\mathbf{X}\boldsymbol{\beta})_j$. This minimization has to be calculated under the additional constraint that the cell probabilities add to 1:

$$\sum_{j=1}^{m} \exp\{(\mathbf{X}\boldsymbol{\beta})_j\} = 1. \tag{6}$$

The problem of the optimization (5) is that the solution is no longer independent of the choice of the orthogonal subspaces X_a. That is, if any set of orthogonal columns X_a of $\mathbf{X}$ is reparametrized by a different orthogonal set, we get a different solution. To avoid this undesirable outcome we use a penalty that is intermediate between the ℓ_1- and the ℓ_2-penalties. This penalty, called group-ℓ_1-penalty, has the following form:

$$\sum_{a \subseteq C} \|\beta_a\|_{\ell_2}, \text{ where } \|\beta_a\|_{\ell_2}^2 = \sum_j (\beta_a)_j^2$$

It has been proposed by Yuan and Lin (2006) for the linear regression problem with factor variables. The estimator of $\boldsymbol{\beta}$ then becomes

$$\widehat{\boldsymbol{\beta}}^{\lambda} = \arg\min_{\boldsymbol{\beta}} \left[-l(\boldsymbol{\beta}) + \lambda \sum_{\substack{a \subseteq C \\ a \neq \emptyset}} \|\beta_a\|_{\ell_2} \right], \tag{7}$$

subject to the constraint in (6). By imposing a penalty function on the coefficients of the log-linear interaction terms, overfitting as it might occur by using MLE is prevented. Furthermore, the ℓ_1-penalty encourages sparse solutions as far as the single components of $\boldsymbol{\beta}$ are concerned, the group ℓ_1-penalty encourages sparsity at the interaction level, meaning that the vector β_a, which corresponds to the interaction term a is either present or absent in the model as a whole. In case of factors with only 2 levels, the group ℓ_1-penalty and the ℓ_1-penalty are equivalent.
For both the ℓ_1-, and the group ℓ_1-regularization, the parameter λ can be assessed e.g. by 10-fold cross-validation: we divide the individual counts into ten equal parts and in turn leave out one part for the rest (90%) to form a training contingency table with cell counts $\boldsymbol{n}_{train}$. The solution for an array of values for λ, the so-called solution path, is calculated according to an algorithm described in Section 4.1. The corresponding vectors of cell probabilities are denoted by $p\left(\widehat{\boldsymbol{\beta}}^{\lambda}\right)$. We then use the remaining 10% of the cell counts $\mathbf{n}_{test}$ to calculate the predictive negative log-likelihood score

$$\frac{-\sum_{j=1}^{m} \mathbf{n}_{test,j} \cdot \log\left(p_j(\widehat{\boldsymbol{\beta}}^{\lambda})\right)}{\sum_{j=1}^{m} \mathbf{n}_{test,j}}, \tag{8}$$

which is proportional to the out-of-sample negative log-likelihood. This score is on the same scale when varying the number of observations and may therefore be used to compare contingency tables of the same dimension but with different numbers of cell entries. The parameter λ is chosen as the value which minimizes the cross-validated score in (8).
The resulting model does not necessarily have to be hierarchical and if we consider the hierarchical model induced by this procedure, it might happen that the final model is large, e.g. if a single high order interaction is estimated to be active. Therefore we set up a regularization approach which we call *level-*ℓ_1-regularized model selection to prevent the algorithm to choose single high-order interaction.

3.3 Level-ℓ_1-Regularized Model Selection

The algorithm fits models as described above for each level of possible interaction order. This means, a model is fitted with main effects only, and the predictive negative log-likelihood score (8) is calculated for the best main effects model (level 1). The same is done for the model including all main effects and first order interactions (level 2). Proceeding accordingly, we get $|C|$ log-likelihood scores corresponding to the $|C|$ levels. Finally, the model with minimal score (8) among all levels is chosen.
With this procedure we tend to select smaller models which can be better hierarchized and interpreted in terms of conditional independence in contrast to the ordinary ℓ_1-model selection procedure.

3.4 Non-Hierarchical Bayesian Model Selection

The Bayesian approach we choose is most closely related to what was proposed by Ntzoufras, Forster, and Dellaportas (2000), George and McCulloch (1993) and Geweke

1994. We use a Markov chain Monte Carlo algorithm based on Stochastic Search Variable Selection (SSVS): SSVS is a procedure proposed by George and McCulloch (1993) to perform variable selection in the standard linear regression model. We adapt this procedure to log-linear models. But instead of assuming a normal mixture model for the coefficients of interest as in SSVS, we follow an approach proposed by Geweke (1994), and assume the coefficients to be a mixture of a point mass at zero and a normal distribution. The complete model is described as follows:

$$\begin{gathered}\mathbf{n} \sim Multinom(\mathbf{p}) \text{ with } \log(\mathbf{p}) = \mathbf{X}\boldsymbol{\beta}, \\ \beta_a|\gamma_a \sim (1-\gamma_a)I_0 + \gamma_a \mathcal{N}(0, \sigma_a^2 1_{d_a}) \text{ independent for all } a \subseteq C, \\ \gamma_a \sim Ber(pr_{\gamma_a}) \text{ independent for all } a \subseteq C, \\ \sigma_a^2 \sim \Gamma^{-1}(l,u) \text{ independent for all } a \subseteq C,\end{gathered} \tag{9}$$

where I_0 is a point mass at zero and γ_a is a Bernoulli variable with probability parameter pr_{γ_a} reflecting prior belief that the corresponding interaction term U_a is present. The parameters σ_a^2 follow an inverse gamma distribution with parameters l and u. In our simulation study, we also considered fixed values for σ_a^2. The choice of the prior parameter l, u and pr_{γ_a} is discussed in Section 4.2. In the absence of strong prior belief, it is reasonable to assume that all σ_a^2 are identically distributed. By imposing prior distributions on the log-linear parameters β_a, it would be possible to incorporate further prior knowledge in the form of existence of correlation or signs of correlation between the different criteria C. One way is to use a prior with expectation different from zero for the corresponding log-linear term ($\mathbf{E}[\beta_a|\gamma_a = 1] \neq 0$). See for example Dellaportas and Forster (1999) for a more detailed discussion on normal priors for the log-linear parameters β_α.

We introduce variables α_a, where $\alpha_a \sim \mathcal{N}(0, \sigma_a^2 1_{d_a})$ and we set $\beta_a = \alpha_a$ if $\gamma_a = 1$ and $\beta_a = 0$ if $\gamma_a = 0$ independent of the value of α_a: $\beta_a = \alpha_a\gamma_a$ has then the desired distribution in (9). This construction is mentioned, but not implemented, in Geweke (1994).

The calculation of the posterior distribution $f(\boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2|\mathbf{n})$ is now required. This cannot be done directly and Monte Carlo approximations are needed, for example from Gibbs sampling. We first calculate the univariate conditional distributions of the parameters α_a or components of $\boldsymbol{\alpha_a}$ if it is a vector:

$$f(\alpha_a|\mathbf{n}, \boldsymbol{\gamma}, \boldsymbol{\alpha}_{\setminus a}, \boldsymbol{\sigma}^2) \propto f(\mathbf{n}|\boldsymbol{\gamma}, \boldsymbol{\alpha}) f(\alpha_a|\sigma_a^2) \propto \exp\{\mathbf{n} \cdot (X_\emptyset \alpha_\emptyset + X_a \alpha_a \gamma_a)\} f(\alpha_a|\sigma_a^2).$$

Although this univariate conditional density is not of any recognized form, we can prove that it is log-concave (see Lemma 2 in the Appendix A for details) and therefore sampling from it can be efficiently done using adaptive rejection sampling, as proposed by Gilks and Wild (1992). Sampling σ_a^2 is straightforward, as

$$f(\sigma_a^2|\mathbf{n}, \boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2_{\setminus a}) = f(\sigma_a^2|\alpha_a) \propto f(\alpha_a|\sigma_a^2) f(\sigma_a^2), \tag{10}$$

and we can easily show that $\sigma_a^2|\alpha_a \sim \Gamma^{-1}(\alpha_a^2/2 + l, u + 1/2)$. Therefore we can sample σ_a^2 from an inverse gamma distribution. In the case where σ_a^2 is assumed to be fixed, this sampling step can be omitted. To sample from $f(\gamma_a|\mathbf{n}, \boldsymbol{\gamma}_{\setminus a}, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2)$, we compute the conditional Bayes factor BF in favour of $\gamma_a = 1$ versus $\gamma_a = 0$. The conditional posterior distribution of γ_a is Bernoulli with $p_{\gamma_a} = \frac{BF}{1+BF}$. Thus we can sample

$$\gamma_a \sim Ber(p_{\gamma_a}).$$

The Bayes factor BF is given by

$$BF = \frac{f(\mathbf{n}|\gamma_a = 1, \boldsymbol{\gamma}_{\setminus a}, \boldsymbol{\alpha}) pr_{\gamma_a}}{f(\mathbf{n}|\gamma_a = 0, \boldsymbol{\gamma}_{\setminus a}, \boldsymbol{\alpha})(1 - pr_{\gamma_a})}.$$

The parameters α_a, σ_a^2 and γ_a are updated in turn for all $a \subseteq C$. In this way we are able to efficiently sample from the full posterior $f(\boldsymbol{\alpha}, \boldsymbol{\gamma}, \boldsymbol{\sigma}^2|\mathbf{n})$ and derive from it the posterior of $f(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\sigma}^2|\mathbf{n})$. From the marginal posterior distribution $f(\boldsymbol{\gamma}|\mathbf{n})$, we can estimate the model probabilities by the sample proportions for $\boldsymbol{\gamma}$, with the most promising models corresponding to the most frequently observed $\boldsymbol{\gamma}$. From $f(\boldsymbol{\beta}|\mathbf{n}, \gamma)$ we can derive the distribution for the interaction strength vector $\boldsymbol{\beta}$ conditional on the model $\boldsymbol{\gamma}$.

3.5 Hierarchical Bayesian Model Selection

We adapt the algorithm described above in a way that allows only moves from one hierarchical model to another, so that we never leave the class of hierarchical models. A hierarchical model is determined by its generators, that is the maximal terms $a \subseteq C$ which are present in the model. The only individual model term which may be removed from a hierarchical model so that it remains hierarchical is a generating term. In addition, Edwards and Havranek (1985) define the dual generators, which are the minimal terms that are not present in the model. The only individual model terms which may be added to the model so that it remains hierarchical are the dual generators.

We consider all hierarchical models to be equally likely and denote the set of generators and dual generators of a hierarchical model corresponding to $\boldsymbol{\gamma}$ with $G_{\boldsymbol{\gamma}}$. We use a Metropolis Hastings algorithm to sample from the full posterior distribution $f(\boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2|\mathbf{n})$. We propose a move from one model $\boldsymbol{\gamma}^t$ to the next model $\boldsymbol{\gamma}^{t+1}$ by choosing an element $G_{\boldsymbol{\gamma}^t}$. Thus we randomly sample an element $a \in G_{\boldsymbol{\gamma}^t}$ and the corresponding γ_a is set to one or zero respectively. The resulting $\boldsymbol{\gamma}$ is denoted as $\boldsymbol{\gamma}^{t+1}$. The corresponding move is accepted with acceptance probability:

$$\min\left(1, \frac{f(\mathbf{n}|\boldsymbol{\gamma}^{t+1}, \boldsymbol{\alpha}^t)|G_{\boldsymbol{\gamma}^t}|}{f(\mathbf{n}|\boldsymbol{\gamma}^t, \boldsymbol{\alpha}^t)|G_{\boldsymbol{\gamma}^{t+1}}|}\right).$$

The sampling procedure for α_a and σ_a^2 is performed exactly as in the non-hierarchical case described in Section 3.4.

4 Implementation

4.1 Algorithm for ℓ_1-Regularization for Factors With Two Levels

For the regularization approaches we calculate $\widehat{\boldsymbol{\beta}}^{\lambda}$ over a large number of values of λ in order to do some cross-validation using (8). For this purpose, an efficient algorithm is required. As one can easily verify by introducing Lagrange multipliers, finding the solution to (7) under the

constraint (6) is equivalent to minimizing an unconstrained function $g(\boldsymbol{\beta})$:

$$g(\boldsymbol{\beta}) = -l(\boldsymbol{\beta}) + n\sum_{j=1}^{m}\exp(\mu_j) + \lambda\sum_{\substack{a\subseteq C\\ a\neq\emptyset}}\|\beta_a\|_{\ell_2}, \tag{11}$$

with $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$. Here, g is a convex function. If each factor has two levels only, as in our application with single-gene libraries, we can set up an algorithm, which efficiently yields the estimates for a whole sequence of parameters λ. Let $\mathcal{A}$ denote the set of active interaction terms, which means for $a \in \mathcal{A}$ it holds that $\beta_a \neq 0$; $\mathbf{X}_\mathcal{A}$ is the corresponding sub-matrix of $\mathbf{X}$, $\boldsymbol{\beta}_\mathcal{A}$ the corresponding sub-vector of $\boldsymbol{\beta}$ and $g_\mathcal{A}$ is g restricted to the subspace $\boldsymbol{\beta}_\mathcal{A}$. We restrict ourselves to the currently active set $\mathcal{A}$, where $\nabla g_\mathcal{A}$ and $\nabla^2 g_\mathcal{A}$ are well-defined:

$$\begin{aligned}\nabla g_\mathcal{A}(\boldsymbol{\beta}_\mathcal{A}, \lambda) &= -\mathbf{X}_\mathcal{A}^t\{\mathbf{n} - n\cdot\exp(\mathbf{X}_\mathcal{A}\boldsymbol{\beta}_\mathcal{A})\} + \lambda(0, sign(\boldsymbol{\beta}_\mathcal{A}))^t\\ \nabla^2 g_\mathcal{A}(\boldsymbol{\beta}_\mathcal{A}, \lambda) &= n\cdot\mathbf{X}_\mathcal{A}^t diag\{\exp(\mathbf{X}\boldsymbol{\beta})\}\mathbf{X}_\mathcal{A}.\end{aligned}$$

The algorithm, which is an adaption of the path following algorithm proposed by Rosset (2005), is set up as follows:

(1) Start with $\widehat{\boldsymbol{\beta}} = (-\log(m), 0, \ldots, 0)$

(2) Set: $\lambda_0 = \max_{\substack{j\in C\\ j\neq\emptyset}}|(\mathbf{X}^t\mathbf{n})_j| = n, \mathcal{A} = \{\emptyset\}$ and $t = 0$.

(3) While $(\lambda_t > \lambda_{min})$

(3.1) $\lambda_{t+1} = \lambda_t - \epsilon$

(3.2) $\mathcal{A} = \mathcal{A}\cup\{j\notin\mathcal{A} : |[\mathbf{X}^t\cdot\mathbf{n} - n\cdot\exp\left(\mathbf{X}\widehat{\boldsymbol{\beta}}\right)]_j| > \lambda_{t+1}\}$

(3.3) $\widehat{\boldsymbol{\beta}}$ is updated as $\widehat{\boldsymbol{\beta}}_{t+1} = \widehat{\boldsymbol{\beta}}_t - \nabla^2 g_\mathcal{A}(\widehat{\boldsymbol{\beta}}_t, \lambda_{t+1})^{-1}\cdot\nabla g_\mathcal{A}(\widehat{\boldsymbol{\beta}}_t, \lambda_{t+1})$.

(3.4) $\mathcal{A} = \mathcal{A}\setminus\{j\in\mathcal{A} : |\widehat{\boldsymbol{\beta}}_{t+1,j}| < \delta\}$

(3.5) $t = t+1$

The pairs $(\widehat{\boldsymbol{\beta}}_t, \lambda_t)$, obtained from the algorithm above, represent the estimates from (7) under the constraint (6) for a range of penalty parameters λ_t e.g. $(t = \epsilon, 2\epsilon, \ldots)$. The choice of the step length ϵ represents the tradeoff between computational complexity and accuracy. To increase accuracy, one can perform more than one Newton step (3.3) if the gradient starts deviating from zero. The coefficient δ is also flexible. Typically it is chosen in the order of ϵ. The lowest λ for which one wants the solution to be calculated is denoted by λ_{min}.
Technical details concerning the algorithm can be found in the Appendix B.

4.2 Prior Specification for Bayesian Methods

For the Bayesian estimation of the parameter vector, we must specify the parameters for the prior distribution of σ_a^2: σ_a^2 plays a role that is similar to that of the parameter λ in the Lasso. The lower σ_a^2, the smaller the estimated coefficient $\widehat{\beta}_a$. An empirical Bayes approach to the implementation could be to specify this parameter by cross-validation. While feasible for the ℓ_1-regularization approaches above, cross-validation becomes prohibitive for the MCMC approaches because of the computational demands. Dellaportas and Forster (1999) proposed a fixed value of two for all a in C, e.g. $\sigma_a^2 = \sigma^2$. Placing a normal prior with mean zero and variance two on each α_a means that with probability 0.95, each of these effects will increase or decrease the ratio of any two cell probabilities by a factor of no more than 10. This is a relatively vague prior, and can be appropriate when no prior information is available. However, our simulation study will illustrate that the final results can be highly sensitive to the choice of this value. To mitigate this sensitivity, we assume σ^2 to have an inverse gamma distribution with mean and variance equal to one, as described in Section 3.4.

In addition, for non-hierarchical model selection, we have to specify the prior distribution for γ_a. We set $\gamma_a \sim Ber(pr_{\gamma_a})$, where pr_{γ_a} reflects prior belief that the corresponding interaction term U_a is present. Without prior knowledge, we assume here that all possible models are a priori equally likely, corresponding to $pr_{\gamma_a} = 1/2$ for all $a \subseteq C$.

This prior is especially attractive when coupled with MAP estimation, as done here, because it effectively cancels out of the MAP calculation. In other situations, this prior may be less compelling. For example, it may be of interest to report posterior probabilities of properties of sets in the model space, such as marginal posteriors of the inclusion of certain coefficients or marginal posteriors of the presence of high order interactions. Then one has to evaluate carefully the mass that priors give to those sets, and one might have to reconsider the choice of the prior distributions to get reasonable posterior probabilities of these sets. In addition, as D, the number of exons, increases, estimating the MAP in the model space becomes difficult and marginal posteriors of summaries such as the model size or the maximum order of interaction may be all that can be reliably estimated. In those circumstances, we suggest graphing these posteriors along with the corresponding priors probabilities, and/or to report Bayes factors.

5 Simulation Study

5.1 Data

We choose the true underlying interaction vector $\boldsymbol{\beta}$ consisting of 5 factors of 2 levels. By enumerating the factors from 1 to 5, the generators of the model are $345 + 235 + 234 + 135 + 123 + 14$, which means that all third and fourth order interactions are absent, only five of ten second order interactions and all first order interactions are present. This defines $\boldsymbol{\gamma}$ and the corresponding coefficients of $\boldsymbol{\beta}$ are independently simulated using a normal distribution with mean zero and variance one.

Then, 250 draws from a multinomial distribution with probability vector $\mathbf{p}$ where $\log(\mathbf{p}) = \mathbf{X}\boldsymbol{\beta}$, are taken. This corresponds to a reasonable number of cDNA in a single-gene library. This

is then repeated 10 times, independently of each other. With our choice of $\boldsymbol{\beta}$, the resulting contingency tables are sparse. With the simulated cell counts, $\widehat{\boldsymbol{\beta}}$ is estimated with different methods described in the previous sections and these methods are then compared in various ways described in the next Section 5.2.

5.2 Criteria

For the MCMC approaches, the maximum a posteriori (MAP) estimators are used. As a model selection score (MSS), the fraction of correctly assigned model terms is reported:

$$\text{MSS} = \frac{1}{m}\sum_{j=1}^{m} |1_{\{\boldsymbol{\beta}_j \neq 0\}} - 1_{\{\widehat{\boldsymbol{\beta}}_j \neq 0\}}|.$$

Moreover, we consider the root mean squared error for the interaction coefficients,

$$RMSE = \sqrt{\frac{1}{m}\sum_{j=1}^{m}(\widehat{\beta}_j - \beta_j)^2}.$$

For assessing how much the estimation of $\boldsymbol{\beta}$ varies over multiple datasets, we calculate for every coefficient $\widehat{\beta}_j$ the estimated standard deviation $\widehat{\sigma}_j$. The means of these standard deviations are reported as

$$Var = \frac{1}{m}\sum_{j=1}^{m}\widehat{\sigma}_j,$$

a measure of variability.

To compare the different procedures for estimation of probabilities $\mathbf{p} = \exp(\mathbf{X}\boldsymbol{\beta})$, we simulate a new dataset $\mathbf{n}_{new}$ of 4000 observations and calculate the out-of-sample negative log-likelihood score (NLS) similar to the score in (8):

$$NLS(\widehat{\boldsymbol{\beta}}) = -\sum_{j=1}^{m} \mathbf{n}_{new,j} \cdot \log\left\{p_j(\widehat{\boldsymbol{\beta}})\right\}$$

5.3 Results

The results are summarized in Table 1. We notice that the penalty-based regularization approaches proposed in this article leads to comparable or better results than the Bayesian approaches with respect to the NLS-score, RMSE and the variation (Var).

The level- and the relaxed ℓ_1-regularization are both competitive and can be better than MCMC for model selection.

The results of the MCMC procedures are sensitive to the choice of the prior value or the prior distribution for σ^2. A flat prior for α_a ($\sigma^2 = 2$) results in worse estimations than with a prior that shrinks the coefficients more towards zero ($\sigma^2 = 1/2$). This suggests that specification of

Table 1: Comparison of different methods to estimate the interaction strength vector $\boldsymbol{\beta}$. MSS, NLS, RMSE and Var are described in Section 5.2. The additional methods relaxed ℓ_1-regularization and ℓ_2-regularization listed in the Table are explained in Section 5.3.

	MSS	NLS	RMSE	Var
Penalty-based regularization methods:				
ℓ_1-regularization	69.7%	8835	0.228	0.144
Level-ℓ_1-regularization	89.7%	8918	0.237	0.179
Relaxed ℓ_1-regularization	82.2%	8900	0.233	0.154
ℓ_2-regularization	-	8833	0.238	0.130
MCMC without model selection:				
$\sigma^2 = 2$	-	9296	0.747	0.401
$\sigma^2 = 1$	-	9105	0.467	0.287
$\sigma^2 = 1/2$	-	8970	0.294	0.201
MCMC with model selection:				
$\sigma^2 \sim \Gamma^{-1}(2,3)$	81.5%	8933	0.294	0.231
$\sigma^2 = 2$	76.6%	9023	0.431	0.342
$\sigma^2 = 1$	78.4%	8951	0.331	0.265
$\sigma^2 = 1/2$	76.6%	8934	0.281	0.225
MCMC with hierarchical model selection:				
$\sigma^2 \sim \Gamma^{-1}(2,3)$	84.1%	8879	0.255	0.180
$\sigma^2 = 2$	80.6%	9176	0.415	0.284
$\sigma^2 = 1$	83.4%	9059	0.308	0.221
$\sigma^2 = 1/2$	83.4%	8966	0.247	0.178
$\sigma^2 = 1/10$	86.3%	8814	0.236	0.097
$\sigma^2 = 1/100$	69.7%	9128	0.420	0.033

this prior hyperparameter may be difficult in practice, while we can easily optimize λ in the regularization approach by cross-validation.

The MCMC approaches without model selection perform poorly, as should be expected from data generated by a sparse model. MCMC methods based on a non-hierarchical model selection are also clearly inferior to the hierarchical counterpart. This is not surprising, as we have simulated data from a hierarchical model.

In Table 1 we have also added an additional approach, denoted by ℓ_2, the equivalent to the ℓ_1-regularization but instead of an ℓ_1-penalty, using an ℓ_2-penalty on the coefficients of the log-linear model. This method is equivalent to the MAP estimator with Gaussian priors on β_a in (9), with the parameter of the distribution optimized by cross-validation. This Ridge-type method does not perform variable selection, but it is very competitive for all other criteria that we assessed.

In addition, the *relaxed* ℓ_1-regularization approach is listed. Rather than using a single penalty

parameter λ, the idea of this method is to control variable selection and parameter estimation by incorporating two penalty parameters. For linear regression it has been proven theoretically as well as empirically (Meinshausen, 2005) that relaxed ℓ_1-regularization is often better than Lasso. Details can be found in the Appendix C.
Overall, the level-ℓ_1-regularization has good model selection performance in combination with low negative log-likelihood score (NLS) and a low mean squared error for the true $\boldsymbol{\beta}$ (RMSE). In addition, it is feasible to optimize the tuning parameter λ by cross-validation as the computational cost is very low compared to the MCMC approaches. On the other hand, posterior distributions of estimates from MCMC methods provide additional information about uncertainty in the model space, compared to point estimates from ℓ_1- or ℓ_2- regularization.

6 Real Data from Single-Gene Libraries

6.1 Dataset

We estimate the splicing interaction pattern for a dataset corresponding to the itpr1 gene, one of three mammalian genes encoding receptors for the second messenger inositol 1,4,5-trisphosphate ($InsP_3$). This gene is subject to alternative RNA splicing, with seven sites of transcript variation, 6 of these within the ORF and among these, $D = 5$ were completely assessed in the single-gene libraries. Five single-gene libraries were built, one for adult rat cerebrum as well as four for different stages of postnatal cerebellar development, namely on days 6, 12, 22 and 90, the latter being considered as adult. Each library consists of between 179 and 277 transcripts which were assessed, i.e. $\sum_{j=1}^{m} n_j \in [179, 277]$. This gene is 89% identical at the cDNA level and 95% identical at the amino acid level with the human receptor gene. The complete dataset can be found in Regan *et al.* (2005).

6.2 Results

Unless stated differently, we report the results using the level ℓ_1-penalization method. We display the interaction vector $\widehat{\boldsymbol{\beta}}$ graphically by plotting the components $\widehat{\beta}_j$ for the different tissue and development stages in Figure 1. We clearly see that the exons interact mainly in pairs and there is no estimated higher order interaction in the splicing interaction pattern of rat cerebellum. We further notice that the main interaction pattern is very well conserved over different developmental stages. A strong mutual interaction between the exons number three, four and five can be observed in all development stages of rat cerebellum as well as in the cerebral tissue. The biggest changes in the interaction pattern during development of rat cerebellum occur from postnatal day six to day 12. This can be seen at position number 10 on the x-axis in Figure 1, and it corresponds to the first order interaction between exons two and three, and from day 12 to day 16, the first main effect changes in sign and magnitude. The first main effect decreases progressively from day 6 to adult, reversing in sign between day 12 and 22. Between day 22 and 90, the interaction pattern is strongly conserved. Comparing the splicing interaction patterns between cerebellum and cerebrum in the adult rat, we see a

much more complex pattern in the cerebrum, involving several second order interactions, and therefore a clear distinction from that of the cerebellum.

A natural way of visualizing a log-linear model is in terms of a graph. A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consists of a finite set $\mathcal{V}$ of vertices and a finite set $\mathcal{E}$ of edges between these vertices. In our context, the vertices correspond to the different alternatively spliced exons. We form the so-called *independence graph* by connecting all pairs of vertices that appear in the same generator. From this graph we can directly read off all marginal and conditional independences by the global Markov property for undirected graphs which states: if two sets of variables a and b are separated by a third set of variables c then a and b are conditionally independent given c ($a \perp\!\!\!\perp b|c$), where for three subsets a, b and c of $\mathcal{V}$, we say c separates a and b if all paths from a to b intersect c.

The independence graphs for the estimated log-linear models are drawn in Figure 2, where the thickness of the edges are proportional to the largest corresponding coefficient of the interaction vector $\widehat{\boldsymbol{\beta}}$ and the radius of the vertices are chosen proportional to the corresponding main effect coefficient. Figure 2 graphically exploits the strongly conserved interactions between exons three, four and five. Except for a rather strong interaction between exon two and three on day six, all other interactions appear to be rather small. The graphical representation of the interaction pattern of adult rat cerebrum reveals a more complex interaction pattern with no conditional independences.

We have also estimated $\boldsymbol{\beta}$ with the hierarchical Bayesian approach using MCMC. For the choice of $\sigma^2 = 1$ this resulted in very similar interaction patterns as for the level ℓ_1-penalization method (see Figure 3). For $\sigma^2 = 2$ it led to remarkably different results. Details can be found in the Supplementary Material. In addition to this, a further dataset was analyzed where the details can be found in the Supplementary Material as well.

As mentioned in Section 4.2, we report Bayes Factors in favour of certain model sizes to get an idea of which order the models are. For rat cerebellum day six, these Bayes factors are 0 for the main effects model, 1.92 for first order interaction, 18.29 for second, 93.95 for third and 0 for fourth order interaction. Similarly for the other developmental stages, it is always the third order interaction model with the largest Bayes Factor. Interestingly, the MAP in the model space is a model involving only second order interactions, but the Bayes Factors speak in favour of a third order interaction model.

7 Conclusions

We have developed efficient frequentist and Bayesian methods for identifying interaction patterns in single-gene libraries. In a simulation study, the results of the new level-ℓ_1-regularization method are superior to hierarchical Bayesian approaches and other frequentist regularization methods. With real data, the level ℓ_1-regularization and hierarchical Bayesian approach led to similar results, subject to a specific choice of priors for the Bayesian method. Massive computational advantages are on the side of the level-ℓ_1-method: the algorithm is sufficiently efficient such that cross-validation becomes feasible which in turn allows for an objective choice of the tuning parameter. On the other hand, posterior distributions of estimates from the hierarchical

Bayes approach can provide a measure of uncertainty on the model or models selected that is harder to derive in a penalized likelihood setting.
The approaches and results presented here can provide valuable insight into the underlying processes in alternative splicing in general, and specifically in the brain development experiments considered here. Most striking is the strong conservation over developmental stages at day 12, 22 and 90 (adult); some differences are showing between postnatal day six and day 12. Also, the conservation between the cerebellum and cerebrum is less pronounced than over developmental stages. Finally, second- or higher-order interaction terms seem to be of minor relevance, suggesting that in this gene/tissue combination, direct interaction mainly happens between pairs of exons, but not combinations of three or more exons.
The level-ℓ_1-method is not restricted to analyzing alternative splicing data, but is a much more general tool which can be applied to a wide variety of problems involving sparse contingency tables. An R package will be made available soon for download under http://stat.ethz.ch/~dahinden/R/loglin.html.

Figure 1: The upper panel shows the estimated splicing interaction vectors $\widehat{\boldsymbol{\beta}}$ of rat cerebellum tissues at postnatal days six, 12 and 22. The lower panel shows the splicing interaction vector $\widehat{\boldsymbol{\beta}}$ of rat cerebellum tissues at the age of 90 days as well as the splicing interaction vector $\widehat{\boldsymbol{\beta}}$ of rat cerebral tissue at the age of 90 days. Within an interaction degree, the sequence of coefficients is ordered from left to right as follows: e.g. for 2nd order interactions, $123, 124, 125, \ldots, 345$, where $1, \ldots, 5$ represent exons 12, 23B, 40, 41, and 42 in the rip3r1 gene, as described in Regan *et al.* (2005).

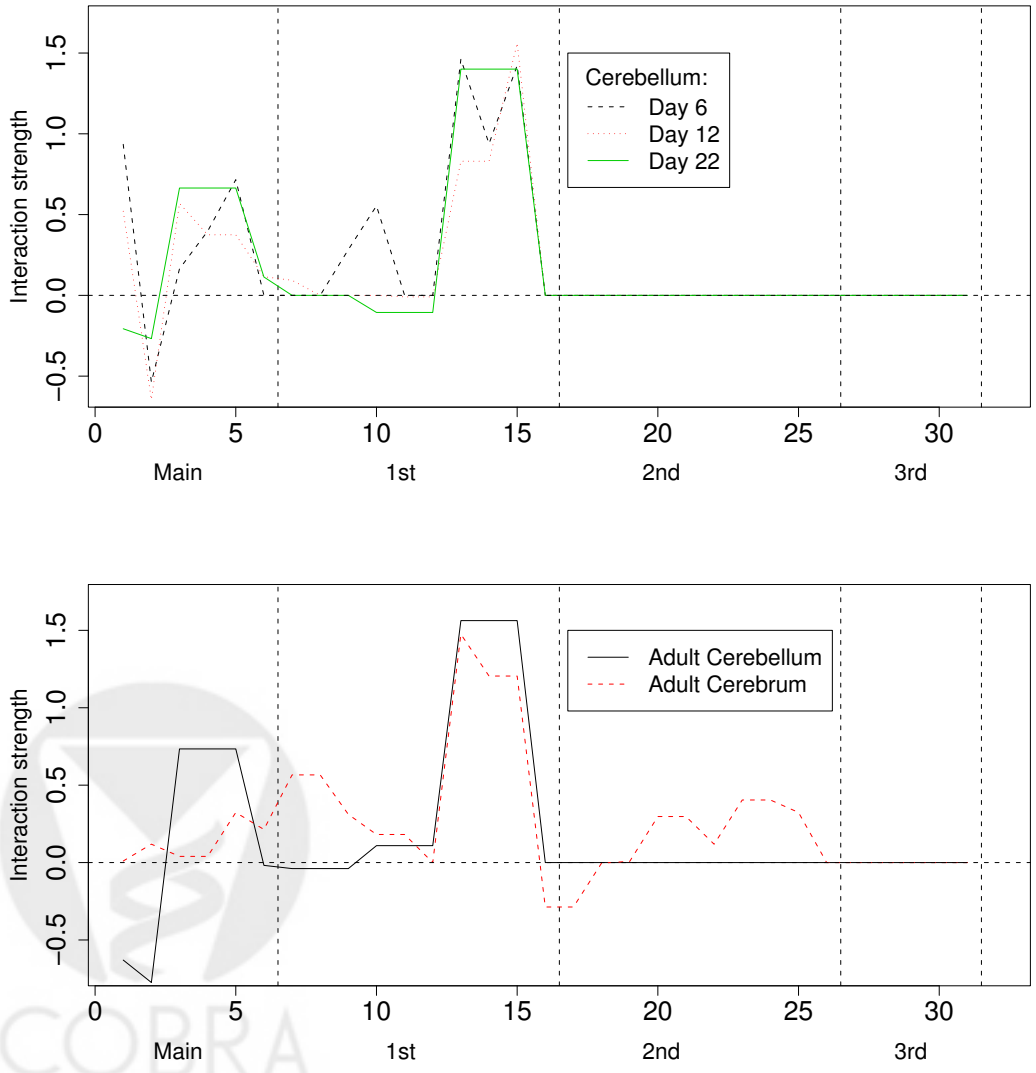


Figure 2: Independence graphs for the estimated log-linear models for the itpr1 gene. For each graph, the predictive probability score (8) is reported as a goodness of fit measure. Note the strong mutual interaction between exons three, four and five.

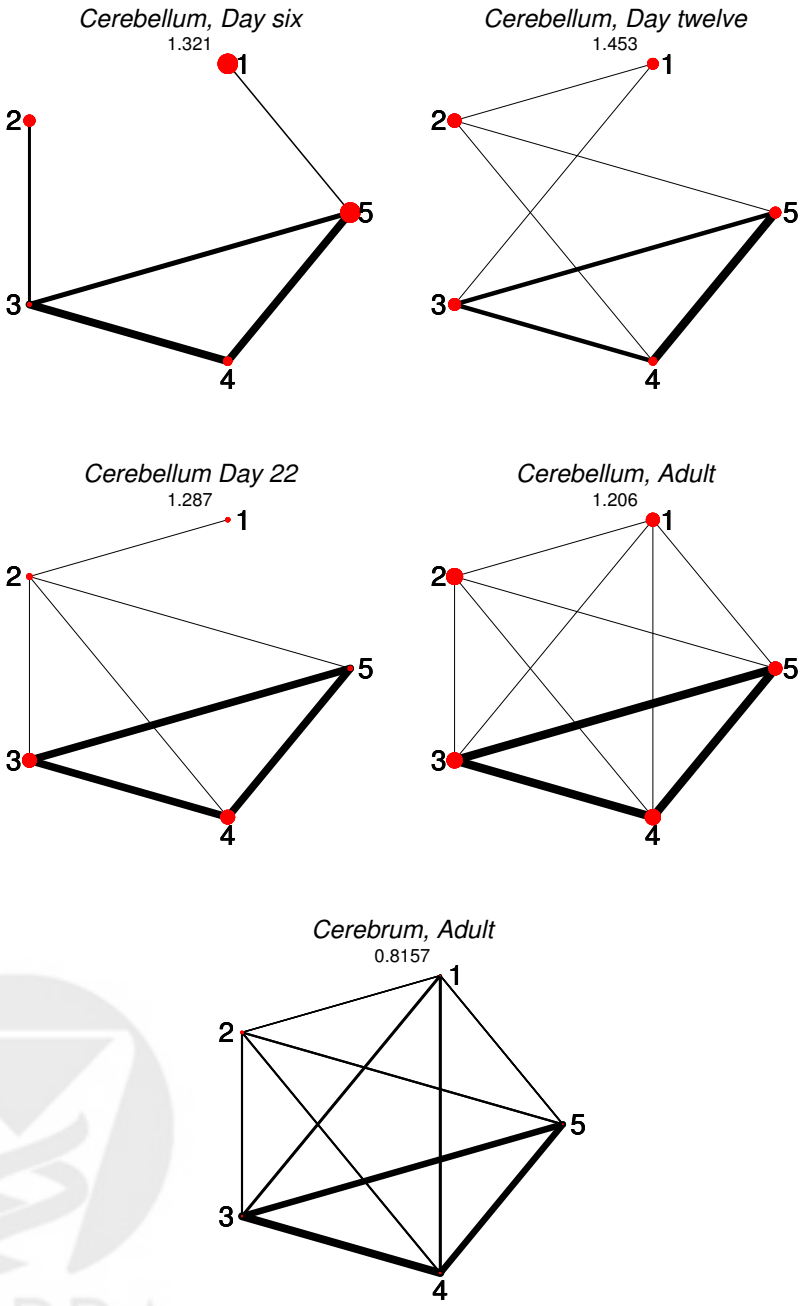
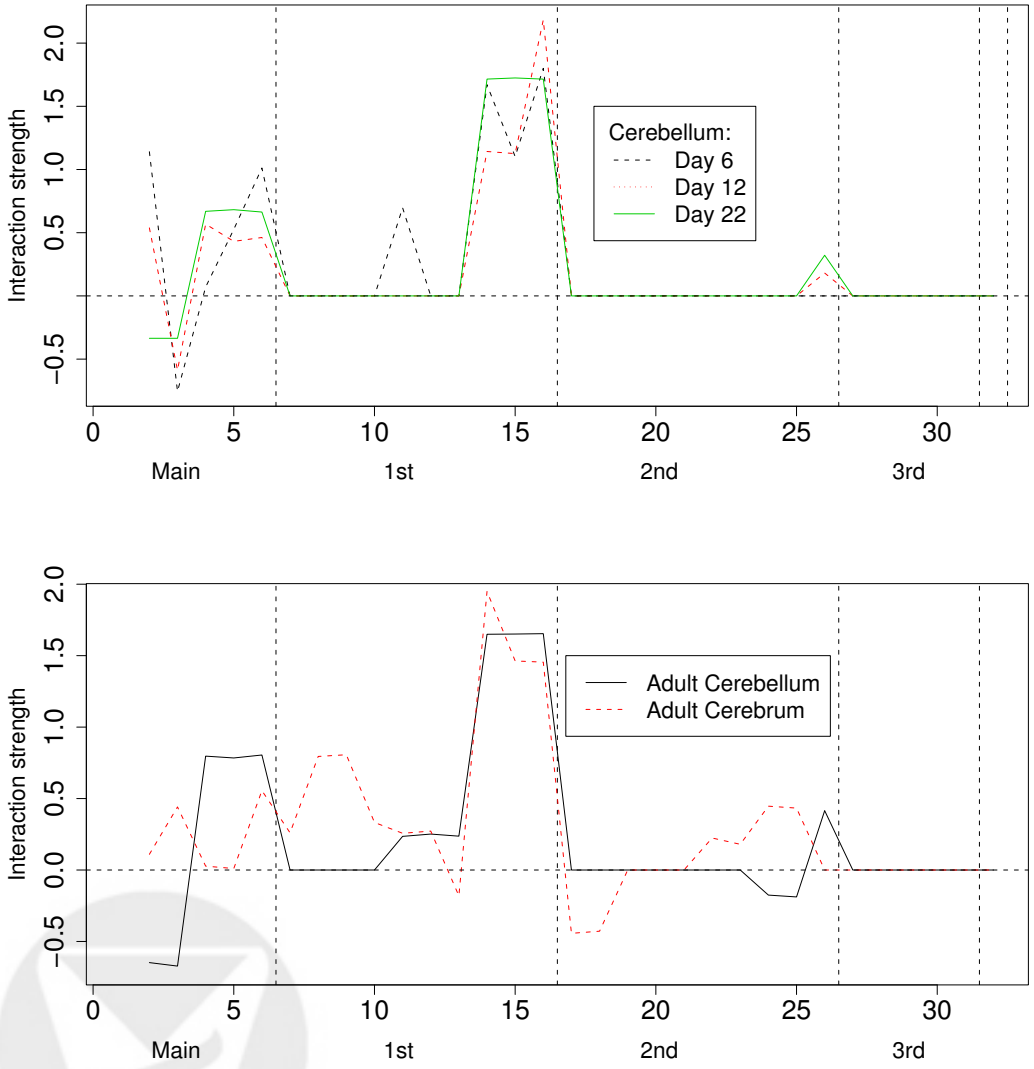


Figure 3: Interaction vectors $\widehat{\boldsymbol{\beta}}$ for the gene itpr1 estimated by the hierarchical MCMC estimator with $\sigma_a^2 = 1$ for all a. Note the close similarity between this interaction pattern and the one from the level-ℓ_1-regularization estimator in Figure 1.



Appendix A

Lemma 1. *$U_a \perp U_b$ for $a \neq b$, i.e. $\sum_{i \in I} u_a(i_a) u_b(i_b) = 0$.*

Proof. For $a \cap b = \emptyset$ it holds that

$$\begin{aligned}\sum_i u_a(i_a)u_b(i_b) &= \sum_{i_a} \sum_{\substack{i,with \\ i_a=const}} u_a(i_a)u_b(i_b) = \sum_{i_a} u_a(i_a) \sum_{\substack{i,with \\ i_a=const}} u_b(i_b) \\ &= \sum_{i_a} u_a(i_a) \frac{1}{|i_a|} \sum_i u_b(i_b) = 0, \text{ because } \sum_i u_b(i_b) = 0,\end{aligned}$$

while $|i_a|$ is the total number of different marginal cells i_a. For $a \cap b = \gamma$ it holds that

$$\begin{aligned}\sum_i u_a(i_a)u_b(i_b) &= \sum_{i_\gamma} \sum_{\substack{i_b,with \\ i_{b\cap\gamma}=i_\gamma}} \sum_{\substack{i,with \\ i_b=const}} u_a(i_a)u_b(i_b) = \sum_{i_\gamma} \sum_{\substack{i_b,with \\ i_{b\cap\gamma}=i_\gamma}} u_b(i_b) \sum_{\substack{i,with \\ i_b=const}} u_a(i_a) \\ &= \sum_{i_\gamma} \sum_{\substack{i_b,with \\ i_{b\cap\gamma}=i_\gamma}} u_b(i_b) \frac{1}{|i_{b\setminus\gamma|}} \sum_{\substack{i,with \\ i_\gamma=const}} u_a(i_a) = 0 \text{ because of (2) .}\end{aligned}$$

□

Lemma 2. *The function $f(\alpha_a|\mathbf{n}, \boldsymbol{\gamma}, \boldsymbol{\alpha}_{\setminus a})$ is log-concave for the prior distributions chosen as described in (9).*

Proof. Without loss of generality we assume that α_a is univariate. The proof for the case that α_a is a vector is exactly the same but for a single component of α_a. We have to prove that the function $h(\alpha_a)$ is concave for

$$h(\alpha_a) = n\alpha_\emptyset + \mathbf{n}^t X_a \alpha_a \gamma_a - \frac{1}{2\sigma^2}\alpha_a^2,$$

where $\alpha_\emptyset$ is the normalizing constant ensuring that all cell probabilities add up to 1. This constant depends on α_a. As the last two terms are concave it remains to be shown that $n\alpha_\emptyset(\alpha_a)$ is concave. For $\gamma_a = 0$ this term is constant and $h(\alpha_a)$ is therefore concave. For $\gamma_a = 1$, we set $\boldsymbol{X}' = \boldsymbol{X}_{\setminus\emptyset}$ and $\boldsymbol{\alpha}' = \boldsymbol{\alpha}_{\setminus\emptyset}$, it then holds

$$\begin{aligned}h(\alpha_a) &= n\alpha_\emptyset = -n \log \sum_{i=1}^m \exp\{(\boldsymbol{X}'\boldsymbol{\alpha}')_i\}, \\ h'(\alpha_a) &= -n \frac{X_a^t \exp(\boldsymbol{X}'\boldsymbol{\alpha}')}{\sum_{i=1}^m \exp\}(\boldsymbol{X}'\boldsymbol{\alpha}')_i\}}, \\ h''(\alpha_a) &= -n \frac{(X_a^2)^t \exp(\boldsymbol{X}'\boldsymbol{\alpha}') \sum_{i=1}^m \exp\{(\boldsymbol{X}'\boldsymbol{\alpha}')_i\} - \{X_a^t \exp(\boldsymbol{X}'\boldsymbol{\alpha}')\}^2}{[\sum_{i=1}^m \exp\{(\boldsymbol{X}'\boldsymbol{\alpha}')_i\}^2]},\end{aligned}$$

where $\exp(\boldsymbol{X}'\boldsymbol{\alpha}')$ has to be understood as the componentwise application of the exponential function and likewise for X_a^2. We now have to show that $h''(\alpha_a)$ is less than zero. If we denote $\exp(\boldsymbol{X}'\boldsymbol{\alpha}')$ by $\mathbf{u}$ and X_a with $\mathbf{x}$, it is sufficient to prove that

$$\sum_{j=1}^{m} x_j^2 u_j \sum_{i=1}^{m} u_i - (\mathbf{x}^t\mathbf{u})^2 \geq 0.$$

The above expression is

$$\sum_{\substack{i,j \\ j<j}}((x_j^2 u_j u_i + x_i^2 u_i u_j) - (2x_i u_i x_j u_j)) = \sum_{\substack{i,j \\ j<j}}(x_j^2 + x_i^2 - 2x_i x_j)u_i u_j = \sum_{i,j,i<j}(x_j - x_i)^2 u_i u_j,$$

which is greater than zero, as $\mathbf{u} > 0$. This proves Lemma 2. □

Appendix B

We note that if $\boldsymbol{\beta}$ is a minimum of g, then $\boldsymbol{\beta}_{\mathcal{A}}$ is a minimum of $g_{\mathcal{A}}$.
In our application with single-gene libraries, all factors have two levels only, which allows to construct an efficient algorithm. Since the gradient

$$\nabla\left[-l(\boldsymbol{\beta}) + n\sum_{j=1}^{m}\exp(\mu_j)\right] = -\mathbf{X}^t \cdot \{\mathbf{n} - n \cdot \exp(\mathbf{X}\boldsymbol{\beta})\},$$

where $\exp(\mathbf{X}\boldsymbol{\beta})$ is understood as the componentwise exponential function, it follows that for a minimum $\boldsymbol{\beta}_{\mathcal{A}}$ of $g_{\mathcal{A}}$, the following equation holds:

$$\nabla g_{\mathcal{A}}(\boldsymbol{\beta}_{\mathcal{A}}) = -\mathbf{X}_{\mathcal{A}}^t \cdot \{\mathbf{n} - n \cdot \exp(\mathbf{X}_{\mathcal{A}}\boldsymbol{\beta})\} + \{0, sign(\boldsymbol{\beta}_{\mathcal{A}})\}^t \cdot \lambda = 0 \tag{12}$$

Without loss of generality, we can restrict ourselves to the subspace $\boldsymbol{\beta} \in \mathbb{R}^- \times \mathbb{R}^{m-1}$, because the constraint (6) can only be satisfied for $\beta_\emptyset < 0$ as is proved in the following Lemma 3. Therefore $\beta_\emptyset \in \mathcal{A}$.

Lemma 3. *$\beta_\emptyset < 0$ for a minimum of $g(\boldsymbol{\beta})$ for all $\lambda \in \mathbb{R}^+$.*

Proof.

$$\log(\mathbf{p}) = \mathbf{X}\boldsymbol{\beta} < 0 \text{ which yields } (1,\ldots,1)\mathbf{X}\boldsymbol{\beta} = m\beta_0 < 0 \text{ this implies } \beta_\emptyset < 0.$$

This holds because $(1,\ldots,1)$ is orthogonal to all columns of $\mathbf{X}$ except for the first one. □

Additionally for $\boldsymbol{\beta}$ being a minimum, a necessary condition is:

$$|[\mathbf{X}^t \cdot \{\mathbf{n} - n \cdot \exp(\mathbf{X}\boldsymbol{\beta})\}]_j| < \lambda, \forall j \notin \mathcal{A}. \tag{13}$$

Conditions (12) and (13) are sufficient for $\boldsymbol{\beta}$ being a minimum of (11). To find the $\boldsymbol{\beta}$'s that solve these equations for an array of values for λ, we set up a so-called path following algorithm.

The idea is to start from an optimal solution $\boldsymbol{\beta}^{\lambda_0}$ for λ_0, and follow the path for decreasing λ, using a second-order approximation for $\boldsymbol{\beta}_{\mathcal{A}}$. In the following, we restrict ourselves to the currently active set $\mathcal{A}$, omitting the index $\mathcal{A}$. It then holds:

$$\begin{aligned} \nabla g(\boldsymbol{\beta}_{t+1}, \lambda_{t+1}) &= 0 \approx \nabla g(\boldsymbol{\beta}_t, \lambda_{t+1}) + \nabla^2 g(\boldsymbol{\beta}_t, \lambda_{t+1}) \delta\boldsymbol{\beta}. \text{ This implies} \\ \delta\boldsymbol{\beta} &= -\nabla^2 g(\boldsymbol{\beta}_t, \lambda_{t+1})^{-1} \nabla g(\boldsymbol{\beta}_t, \lambda_{t+1}). \end{aligned} \tag{14}$$

The algorithm tries to follow the optimal path as close as possible. At each step, it aims to meet the conditions (12) and (13). In step (3.2), the active set $\mathcal{A}$ is identified, which forces $\widehat{\boldsymbol{\beta}}$ to meet the condition (13). In step (3.3), a Newton step as described in (14) is performed. Starting from a solution which meets condition (12), the new $\widehat{\boldsymbol{\beta}}^{\lambda}$ approximately meets (12) again

Appendix C

Relaxed ℓ_1-Regularized Model Selection

The two-stage *relaxed Lasso* is defined as follows:

$$\widehat{\boldsymbol{\beta}}^{\lambda,\mu} = \arg\min_{\boldsymbol{\beta}} \left[-l(\boldsymbol{\beta}_{\mathcal{M}_\lambda}) + \mu \sum_{j \in \mathcal{M}_\lambda} |\beta_j| \right],$$

where $\mathcal{M}_\lambda = \{1 \le k \le m | \widehat{\beta}_k^{\lambda} \neq 0\}$, $\widehat{\boldsymbol{\beta}}^{\lambda}$ as in (7) and $\boldsymbol{\beta}_{\mathcal{M}_\lambda}$ denotes a vector consisting only of components in $\mathcal{M}_\lambda$.

For selecting the parameters λ and μ, a similar approach is chosen as with ℓ_1-regularization in Section 3.2. First, we compute all possible submodels $\mathcal{M}_\lambda$ for the full dataset. Then, for each submodel the second parameter μ is selected as for the regular ℓ_1-regularization by computing the negative log predictive probability score (8). Finally, the parameters (λ, μ) are chosen to minimize the score (8). When we prefer hierarchical over non-hierarchical models, we consider a hierarchical $\mathcal{M}_\lambda$, meaning that λ and μ are assessed using the hierarchical model induced by $\mathcal{M}_\lambda$, i.e. the smallest hierarchical model which contains all elements of $\mathcal{M}_\lambda$. Due to the second-stage penalization, $\widehat{\boldsymbol{\beta}}^{\lambda,\mu}$ is not necessarily hierarchical though.

Supplementary Material

Under http://stat.ethz.ch/~dahinden/Biometrics/Suppmat.pdf the supplementary material containing results from additional datasets as well as results for additional variable selection strategies can be viewed.

Acknowledgements

Dahinden's research was partially founded by the Swiss National Science Foundation (SNF). Parmigiani was partly supported by NSF grant DMS034211.

References

Aitchison, J. (1986), *The Statistical Analysis of Compositional Data*, Chapman and Hall, London.

Birch, N.W. (1963), Maximum Likelihood in Three-Way Contingency Tables, Journal of the Royal Statistical Society, Ser. B, 25, 220–233.

Bishop, Y., Fienberg, S., and Holland, P. (1975), *Discrete Multivariate Analysis*, MIT Press, Cambridge, Massachusetts and London, UK.

Brett, D., Hanke, J., Lehmann, G., Haase, S., Delbruck, S., Krueger, S. R., and J. Bork, P. (2000), EST Comparison Indicates 38% of Human mRNAs Contain Possible Alternative Splice Forms, FEBS Letters 474, 83–86.

Brett, D., Pospisil, H., Valcarcel, J., Reich, J., and Bork, P. (2002), Alternative Splicing and Genome complexity, Nature Genetics 30, 29–30.

Christensen, R. (1991), *Linear Models for Multivariate Time Series, and Spatial Data*, Springer-Verlag.

Darroch, J.N., Lauritzen, S.L., and Speed, T.P. (1980), Markov Fields and Log-Lineaer Interaction Models for Contingency Tables, Annals of Statistics 8, 522–539.

Dellaportas, P. and Forster, J. (1999), Markov Chain Monte Carlo Model Determination for Hierarchical and Graphical Log-Linear Models, Biometrika 86, 615–633.

Edwards, D. and Havranek, T. (1985), A Fast Procedure for Model Search in Multidimensional Contingency Tables, Biometrika 72, 339–351.

Emerick, M.C., Stein, R., Kunze, R., McNulty, M., Regan, M.R., Hanck, D.L., and Agnew, W.S. (2006), Profiling the Array of $Ca_v3.1$ Variants From the Human T-Type Calcium Channel Gene CACNA1G: Alternative Structures, Developmental Expression and Biophysical Variations. Proteins: Structure, Function and Bioinformatics .

Ewing, B. and Green, P. (2000), Analysis of Expressed Sequence Tags Indicates 35000 Human Genes, Nature Genetics 25, 232–234.

FANTOM Consortium, RIKEN Genome Exploration Research Group, and the Genome Science Group (Genome Network Project Core Group) (2005) The Transcriptional Landscape of the Mammalian Genome, Science 309, 1559–1563.

George, E. I. and McCulloch, R.E. (1993), Variable Selection via Gibbs Sampling, Journal of the American Statistical Association 88, 881–889.

Geweke, J. (1996), Variable Selection and Model Comparison in Regression, in *Bayesian Statistics 5*, edited by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith, University Press, Oxford, 609–620.

Gilks, W.R. and Wild, P. (1992), Adaptive Rejection Sampling for Gibbs Sampling, Applied Statistics 41, 545–557.

Imanishi, T., Itoh, T., Suzuki, Y., O'Donovan, C., Fukuchi, S., Koyanagi, KO, Barrero, RA., Tamura, T., Yamaguchi-Kabata, Y., and Tanino, M. (2004), Integrative Annotation of 21037 Human Genes Validated by Full-Length cDNA Clones, PloS Biology 2, 1–20.

International Human Genome Sequencing Consortium (2001), Initial Sequencing and Analysis of the Human Genome, Nature 409, 860–921.

International Human Genome Sequencing Consortium (2004), Finishing the Euchromatic Sequence of the Human Genome, Nature 431, 931–945.

King, R. and Brooks, S.P. (2001), Prior Induction in Log-Linear Models for General Contingency Table Analysis, Annals of Statistics 29, 715–747.

Liang, F., Holt, I., Pertea, G., Karamycheva, S., Salzberg, S., and Quackenbush, J. (2000), Gene Index Analysis of the Human Genome Estimates Approximately 120000 Genes, Nature Genetics 25, 239–240.

Madigan, D. and York, J.C. (1997), Bayesian Methods for Estimation of the Size of a Closed Population. Biometrika 84, 19–31.

Meinshausen, N. (2005), Lasso with Relaxation, Technical Report 129, ETH Zürich .

Mironov, A.A., Fickett, J.W., and Gelfand, M.S. (1999), Frequent Alternative Splicing of Human Genes, Genome Research 9, 1288–1293.

Ntzoufras, I., Forster, J., and Dellaportas, P. (2000), Stochastic Search Variable Selection for Log-linear Models, Journal of Statistical Computation and Simulation 68, 23–37.

Regan, M.R., Lin, D.D.M., Emerick, M.C., and Agnew, W.S. (2005), The Effect of Higher Order RNA Processes on Changing Patterns of Protein Domain Selection: A Developmentally Regulated Transcriptome of Type 1 Inositol 1,4,5-Trisphosphate, Proteins: Structure, Function and Bioinformatics 59, 312–331.

Rosset, S. (2005), Following Curved Regularized Optimization Solution Paths, in *Advances in Neural Information Processing Systems 17*, edited by L. K. Saul, Y. Weiss, and L. Bottou, MIT Press, Cambridge, 1153–1160.

Southan, C. (2004), Has the yo-yo Stopped? An Assessment of Human Protein-Coding Gene Number, Proteomics 4, 1712–1726.

Tibshirani, R. (1996), Regression Shrinkage and Selection via the Lasso, Journal of the Royal Statistical Society, Ser. B, 58, 267–288.

Venter, J., Adams, MD., Myers, EW., Li, PW., and Mural, RJ. *et al.* (2001), The Sequence of the Human Genome, Science 291, 1304–1351.

Yuan, M. and Lin, Y. (2006), Model Selection and Estimation in Regression with Grouped Variables, Journal of the Royal Statistical Society, Ser. B, 68(1), 49–67.

Zavolan, M., van Nimwegen, E., and Gaasterland, T. (2003), Splice Variation in Mouse Full-Length cDNAs Identified by Mapping to the Mouse Genome, Genome Research 12, 1377–1385.