Lasso model selection in multi-dimensional contingency tables?

Gilbert MacKenzie¹ and Susana Conde²

¹ ENSAI, France and University of Limerick, Ireland
² Manchester University, UK, and The University of Glasgow, UK.

Email: gilbert.mackenzie@ul.ie and susana.conde@manchester.ac.uk

Abstract: We develop a Smooth Lasso for sparse, high dimensional, contingency tables and compare its performance with the usual Lasso and with the now classical backwards elimination algorithm. In simulation, the usual Lasso had great difficulty identifying the correct model. Irrespective of the sample size, it did not succeed in identifying the correct model in the simulation study! By comparison the smooth Lasso performed better improving with increasing sample size. The backwards elimination algorithm also performed well and was better than the Smooth Lasso at small sample sizes. Another potential difficulty is that Lasso methods do not respect the marginal constraints on hierarchy and so lead to non-hierarchical models which are unscientific. Furthermore, even when one can demonstrate, classically, that some effects in the model are inestimable, the Lasso methods provide penalized estimates. These problems call Lasso methods into question.

Keywords: False estimation, Lasso, Model selection, Non-hierarchical models, Smooth Lasso

1 Introduction

Sparse contingency tables arise often in genetic, bioinformatic and database applications. Then the target is to estimate the dependence structure between the variables modelled via the interaction terms in a log-linear model. High dimensionality will force attention on identifying important low-order interactions - a technical advance since most model selection work relies only on main effects. Penalized likelihood attaches a penalty function of the parameters to the likelihood in order to achieve some purpose such as smoothing (Eilers and Marx, 1996), or sparsity (Friedman, 2008). Using the LASSO (L₁-norm penalty), some of the parameters go to zero allowing a more parsimonious model to be found. Dahinden (2007) extended the LASSO (Tibrishani, 1996) to contingency tables and log-linear models. However, in the Lasso the penalty is a non-differentiable function of the parameters thus necessitating specialized optimization algorithms.
We present the smooth LASSO, a penalized likelihood, which does not require specialized optimization algorithms such as the method of coordinate descent. It uses a convex, parametric, analytic penalty function that asymptotically approximates the LASSO: minimization is accomplished using standard Newton-Raphson algorithms and standard errors are available.

2 Model Formulation

2.1 Log-linear modelling
Assume \( X_1, \ldots, X_v \) correlated binary variables (off=0, on=1) and these form a \( v \)-dimensional contingency table with \( q = 2^v \) cells. Let \( Y_i \) be the random variable indicating the frequency in the \( i \)th cell, \( i = 1, \ldots, q \) and let \( \mu_i = E(Y_i) \). We consider a log-linear regression model: \( \log(\mu_i) = A^T \theta \) where \( A \) is a \((q \times p)\) design matrix of fixed constants with typical element \( a_{ij} \), and \( \theta \) is a vector with \( p \) dimensions measuring the influence of the effects (constant, main effects and interactions) on the response vector of counts \( Y \). We use Yates’ design matrix coding scheme whence the columns of \( A \) are orthogonal. Finally, let \( n = \sum_{i=1}^q Y_i \) denote the total number of observations. Estimation is via the log-likelihood, which may be taken in Poisson form: \( \ell(\theta | y) \propto \sum_{i=1}^q \{ y_i (a_i^T \theta) - \exp(a_i^T \theta) \} \), as the maximum likelihood estimators are the same in multinomial and independent Poisson schemes provided \( \sum_{i=1}^q \mu_i = n \) (Birch 1963). The log-likelihood may be maximized numerically using iterative proportional fitting or by generating the design matrix \( A \) and using the \texttt{nlm} procedure in the R software package.

2.2 A Smooth LASSO
The penalized log-likelihood is:
\[
\ell_\lambda(\theta) = \ell(\theta) - \text{pen}_\lambda
\]
(1)
where \( \text{pen}_\lambda \), is the penalty term, \( \lambda > 0 \). For the LASSO \( \text{pen}_\lambda = \lambda \sum_{j=2}^p |\theta_j| \) omitting the intercept term and for the Smooth LASSO \( \text{pen}_\lambda = \lambda \sum_{j=2}^p Q_\omega(\theta_j) \) where \( Q_\omega(\theta_j) = \omega \log \left[ \cosh \left( \frac{\theta_j}{\omega} \right) \right] \) for a constant \( \omega \) that regulates the approximation of the function to that of the absolute value function (Salje et al, 2005). Note that \( Q_\omega(\theta_j) \in C^\infty \), the set of functions that are infinitely differentiable, and is convex. Following we define the maximum penalised likelihood estimator (MPLE) as
\[
\hat{\theta} := \arg \max_{\theta \in \Theta} \{ \ell(\theta) - \text{pen}_\lambda(\theta) \}.
\]
(2)
We should more properly write \( \hat{\theta}_\lambda \), rather than \( \hat{\theta} \), but the dependence on \( \lambda \) will be understood in what follows. For a large \( \lambda \), all the estimates go to 0 and for \( \lambda = 0 \), there is no constraint, whence \( \hat{\theta}_{\lambda=0} \) is equivalent to the usual maximum likelihood estimator (MLE).
3 Non-hierarchical model

We digress to make an important methodological point by comparing Yates’ and Binary design matrix coding schemes in a non-hierarchical model using a well known example. Agresti (2002) gave the following $2^3$ table $\mathbf{y} = (19, 11, 0, 6, 132, 52, 9, 97)$ of counts classified by: A = defendant’s race (0. white, 1. black), B = victim’s race (0. white, 1. black) and C = death penalty (0. yes, 1. no). The contingency table is written in vector notation in which the leftmost subscript varies fastest. Table 1 shows the result of

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Estimated $\dot{\gamma}$</th>
<th>Estimated $\dot{\beta}$</th>
<th>$se_{\dot{\gamma}}$</th>
<th>$se_{\dot{\beta}}$</th>
<th>$z_{\dot{\gamma}}$</th>
<th>$z_{\dot{\beta}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_0$</td>
<td>3.520</td>
<td>3.689</td>
<td>0.067</td>
<td>0.079</td>
<td>52.714</td>
<td>46.670</td>
</tr>
<tr>
<td>$\theta_A$</td>
<td>0.018</td>
<td>-1.825</td>
<td>0.055</td>
<td>0.274</td>
<td>0.332</td>
<td>-6.66</td>
</tr>
<tr>
<td>$\theta_{AB}$</td>
<td>0.630</td>
<td>0.492</td>
<td>0.067</td>
<td>0.160</td>
<td>9.442</td>
<td>3.074</td>
</tr>
<tr>
<td>$\theta_{AC}$</td>
<td>0.031</td>
<td>2.171</td>
<td>0.055</td>
<td>0.256</td>
<td>0.554</td>
<td>8.480</td>
</tr>
</tbody>
</table>

$\ell(\dot{\gamma}) = -161.7495$, $\ell(\dot{\beta}) = -150.65$

fitting the non-hierarchical model A, AB, AC with with Yates’ ($\dot{\gamma}$) and Binary ($\dot{\beta}$) design matrices. We have the same data, the same model, but the likelihoods differ and the effects have different interpretations in the two models. This simple example shows that we should restrict model selection to hierarchical models.

Even when fitting hierarchical models, only effects in the generating set of the fitted model are invariant to the choice of design matrix. The application of Wald tests to other effects is mistaken. The likelihoods, however, are invariant. These findings apply to all statistical models with interaction terms.

4 Lasso Model Selection

4.1 Simulation

We conducted a small simulation study designed to study the percentage of correct models identified by three algorithms: Backwards Elimination, the usual Lasso and the Smooth Lasso. For the purposes of illustration we simulated a $2^5$ contingency table when the main effects model was true. The number of replications was $m = 1000$ and we started with the all 2-way interactions design-matrix. For the backwards elimination method we used a
TABLE 2: Simulation: Percentage of correct models identified by three methods.

<table>
<thead>
<tr>
<th>Sample size</th>
<th>BE</th>
<th>Lasso</th>
<th>SL-95</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>62.3</td>
<td>0*</td>
<td>0.1</td>
</tr>
<tr>
<td>100</td>
<td>51.6</td>
<td>0</td>
<td>11.8</td>
</tr>
<tr>
<td>500</td>
<td>33.0</td>
<td>0</td>
<td>50.1</td>
</tr>
<tr>
<td>1000</td>
<td>29.2</td>
<td>0</td>
<td>51.6</td>
</tr>
</tbody>
</table>

* The Lasso persistently over fits effects.

R function written Conde (2011), for the usual Lasso we used the \texttt{glmnet} R package and for the Smooth Lasso we used another R function, which called \texttt{nml}. The tuning parameter $\lambda$ was estimated by 10 fold cross-validation in the Lasso functions. The sample sizes studied were: $n = 50, 100, 500, 1000$. For the Smooth Lasso one must pick a level of statistical significance, as with ordinary regression methods (Conde & MacKenzie, 2010). Thus SL-95 corresponds to the 5% level. It will be noticed that the 5% level produces poor results when the sample size is small, but improves with increasing sample size, while the classical Backward Elimination algorithm performs better for smaller sample sizes.

4.2 Obesity Data Analysis

We now present the results of analysing a set of obesity data comprising 8 binary comorbidities measured on $n = 5550$ patients. The resulting contingency table has $2^8$ cells of which 45.3% are zero cells. We compare the three algorithms described above using the same fitting methods. Table 3 presents the generating sets defining the final models together with their AICs. Several interesting features emerge.

First the fitted Lasso-based solution comprised non-hierarchical models. Each non-hierarchical model was then augmented by adding in effects to produce a minimum hierarchical model. The models were re-estimated (Table 3). Unfortunately, this idea does not always work - often, in sparse tables, one finds that minimum hierarchical model contains effects which are non-estimable, whence one is stuck with a Lasso solution which is non-hierarchical. Such solutions are unscientific.

A second problem arises with the Lasso methods investigated. If one pre-processes the table one can identify effects which are inestimable in the classical paradigm (using a theorem due to the first author). On first noticing this we hoped that if the Lasso was going to produce a sparse model it would somehow identify the inestimable effects and shrink these to zero.
TABLE 3: Generating sets of models found by Backwards Elimination, LASSO and Smooth LASSO.

<table>
<thead>
<tr>
<th>Model</th>
<th>BE</th>
<th>LASSO*</th>
<th>SL-95*</th>
</tr>
</thead>
<tbody>
<tr>
<td>[c1c6, c1c8, c2c3, c2c4, c2c5, c3c4, c5c6, c1c4, c4c5c7, c4c6c8, c6c7c8]</td>
<td>[c1c2c4, c1c2c7, c1c3c7, c1c3c8, c1c5c6, c1c5c7, c1c5c8, c1c6c7, c1c6c8, c1c7c8, c2c3c5, c2c3c6, c2c3c7, c2c4c7, c2c4c8, c2c6c8, c3c4c6, c3c4c7, c3c4c8, c3c5c6, c3c6c8, c4c5c6, c4c5c7, c4c5c8, c4c6c7, c4c6c8, c4c7c8, c5c6c8, c5c7c8, c6c7c8]</td>
<td>[c1c6, c1c7, c1c8, c2c4, c2c5, c3c4, c3c6, c4c5, c4c8, c6c7, c6c8]</td>
<td></td>
</tr>
<tr>
<td>AIC</td>
<td>722.687</td>
<td>749.831</td>
<td>1254.699</td>
</tr>
</tbody>
</table>

*Minimal hierarchical model that includes the effects in the support. For the smooth LASSO, \( \omega = 1 \).

However this is not the case and we have many examples of the Lasso and Smooth Lasso solutions producing penalized estimates of inestimable effects. One might be tempted to regard this as an “advantage”, but this seems naïve. The solution is inconsistent with the classical theory. One possible explanation is that the penalized likelihoods have a Bayesian interpretation in which the penalty plays the role of a prior. So false estimation of inestimable effects may just correspond to a value assigned by the prior. If so, this is yet another reason for discarding such solutions.

Accepting these caveats, we note that: (a) the BE algorithm always produces a hierarchical model, (b) the BE algorithm is best as judged by the AIC, (c) it is also fastest, (d) the Lasso is not the sparsest model and (e) the smooth LASSO is much more parsimonious than the LASSO. These are consistent findings in our work.

5 Discussion

There is, apparently, a highly impressive literature on Lasso methods. It is, however, predicated on model selection based on main effects models. In the presence of interactions, Lasso methods will often fail to produce scientific models. It has been argued that group Lasso methods provide one answer to this problem, but they require multiple tuning parameters, one for each class of interactions anticipated in the final solution. Accordingly, they are prohibitively computationally expensive. Other authors have argued for weak hierarchy (Bien et al, 2013). Their arguments are not compelling
and difficult to implement. Moreover, it is well known that the Lasso lacks the *oracle property* and the results in Table 2 confirm this. However, the results suggest that this may not be the case for the Smooth Lasso, a finding which requires further investigation. To our knowledge the problem of false estimation has not previously been reported. All these issues raise serious questions about the usefulness of Lasso methods for model selection.

**Acknowledgments:** We acknowledge the help and support of Professor Victor Kiri, of FVJK consultants, UK. The second author works on the MI-MOMICS project in the Centre of Biostatistics in Manchester University, UK.

**References**


