Grouped and Hierarchical Model Selection through Composite Absolute Penalties

Peng Zhao, Guilherme Rocha, Bin Yu
Department of Statistics University of California, Berkeley, USA
{pengzhao, gvrocha, binyu}@stat.berkeley.edu

April 17, 2006

Abstract

Recently much attention has been devoted to model selection through regularization methods in regression and classification where features are selected by use of a penalty function (e.g. Lasso in Tibshirani, 1996). While the resulting sparsity leads to more interpretable models, one may want to further incorporate natural groupings or hierarchical structures present within the features.

Natural grouping arises in many situations. For gene expression data analysis, genes belonging to the same pathway might be viewed as a group. In ANOVA factor analysis, the dummy variables corresponding to the same factor form a natural group. For both cases, we want the features to be excluded and included in the estimated model together as a group. Furthermore, if interaction terms are to be considered in ANOVA, a natural hierarchy exists as the interaction term between two factors should only be included after the corresponding main effects. In other cases, as in the fitting of multi-resolution models such as wavelet regression, the hierarchy between bases on different resolution levels should be enforced, that is, the lower resolution base should be included before any higher resolution base in the same region.

Our goal is to obtain model estimates that approximate the true model while preserving such group or hierarchical structures. Assuming data is given in the form $(Y_i, X_i); i = 1, ..., n$, where $X_i \in \mathcal{X} \subset \mathbb{R}^d$ are explanatory variables and $Y_i \in \mathcal{Y}$ a response variable, also assuming the estimate for Y is of the form $f(X) \cdot \beta$, where $\beta \in \mathbb{R}^p$ are the model coefficients and $f : \mathcal{X} \to \mathcal{X}^* \subset \mathbb{R}^p$ the features, we obtain our model estimates by jointly minimizing a goodness of fitness criterion represented by a convex loss function $L(\beta, Y, X)$ and a suitably crafted CAP (Composite Absolute Penalty) penalty function. Such a framework fits within that of penalized regressions.

The CAP penalty function is constructed by first defining groups G_i, $i = 1, ..., k$ that reflect the natural structure among the features. A new vector is then formed by collecting the L_{γ_i} $(i = 1, \ldots, k)$ norm of the coefficients β_{G_i} associated with the features within each of the groups. These are the group-norms and they are allowed to differ from group to group. The CAP penalty is then defined to be the L_{γ_0} norm (the overall norm) of this new vector. By properly selecting the group-norms and the overall norm, selection of variables can be done in a grouped fashion (Grouped Lasso by Yuan and Lin (2006) and Blockwise Sparse Regression by Kim et al. (2006) are special cases of this penalty class). In addition, when the groups are defined to overlap, this construction of penalty provides a mechanism for expressing hierarchical relationships between the features.

When constructed with $\gamma_i \geq 1$, for $i = 0, \ldots, k$, the CAP penalty functions closely resemble proper norms and are proven to be convex which renders CAP computationally feasible. In this case, the BLASSO algorithm (Zhao and Yu, 2004) can be used to trace the regularization path. Particularly, in Least Squares Regressions, when the norms are restricted to combinations of L_1 and L_∞ norms, the regularization paths are piecewise linear. Therefore we provide LARS-fashioned (Efron et al., 2004) algorithms, which jump between the turning points of the piecewuse linear path, to compute the entire regularization path efficiently.

1 Introduction

Regularization has recently gained enormous attention in statistics and the field of machine learning due to the high dimensional nature of many current datasets. The high dimensionality could lead us to models that are very complex. This poses challenges in two most fundamental aspects of statistical modeling – prediction and interpretation. On one hand, it is inherently unstable to fit a model with a large number of parameters which leads to poor prediction performance. On the other hand, the estimated models are often too complex to reveal interesting aspects of data. Both of these challenges force us to regularize the estimation procedure to obtain more stable and interpretable model estimates.

Problems where the data dimension p is large in comparison to sample size n have become common over the recent years. Two examples are the analysis of micro-array data in Biology (Dudoit et al., 2003, e.g.) and cloud detection through analysis of satellite images composed of many sensory channels (Shi et al., 2004). In such cases, structural information within the data can be incorporated into the model estimation procedure to significantly reduce the actual complexity involved in the estimation procedure. Regularization methods provide a powerful yet versatile technique for doing so. They are utilized by many successful modern methods like Boosting (Freund and Schapire, 1997), Support Vector Machine (Vapnik, 1995) and Lasso (Tibshirani, 1996; Chen and Donoho, 1994; Chen et al., 2001). The regularization is, in some cases, imposed implicitly as in early stopping of Boosting (Bühlmann and Yu, 2003) or, in other cases, imposed explicitly through the use of a penalty function as in Lasso. Our approach falls into the latter category.

The main contribution of this paper is the introduction of the Composite Absolute Penalties (CAP) family of penalties that are convex, highly customizable and enable users to build their subjective knowledge o the data structure into the regularization procedure. It goes beyond the Lasso and encompasses group selecting penalties like GLasso in (Yuan and Lin, 2006) and similarly in (Kim et al., 2006) as a special case and extends it to hierarchical modeling. This inclusion of structural regularization significantly improves prediction as shown in our extensive simulations and in an application to arctic cloud detection based on multi-angle satellite images Shi et al. (2004).

In what follows, we let $Z = (Y, X)$ with $Y \in \mathbb{R}^n$ a response variable and $X \in \mathbb{R}^{n \times p}$, denote the observed data. The estimates defined by these penalized methods can be expressed by:

$$\hat{\beta} \quad = \arg\min_\beta \quad L(Z, \beta) + \lambda \cdot T(\beta)$$

where L is a loss function representing the goodness of fit of the model. Typical examples include log-likelihood functions, such as the squared error loss for ordinary least squares regression and logistic loss function, and the hinge loss in Support Vector Machines. T is a penalty function that enforces complexity (size of the parameters) and structural constraints (e.g. sparsity and group structure) on the estimates.

It can also be used as a way of incorporating side or prior information into estimation. The sources of side information are diverse and range from function smoothness in Smoothing Splines to distributional information on the predictor variables in the popular field of semi-supervised learning. The regularization parameter λ adjusts the trade off between fidelity to the observed data and reduction of the penalty. As the regularization parameter increases, the estimates become more constrained, therefore the variance of the estimates tend to decrease whereas the bias in the estimates tend to increase as the estimates become less faithful to the observed data. Except for special cases, choosing the regularization parameter is not trivial and usually requires computation of the entire regularization path – the set of regularized estimates corresponding to different values of λ's. We will present efficient algorithms that give the entire regularization path. For a subset of the CAP penalties, we also derive an unbiased estimate of the degrees of freedom to facilitate choosing the amount of regularization.

One of the early examples of the use of penalties within the estimation framework is the ridge regression (Hoerl and Kennard, 1970). In this work, a penalty on the squared norm of the coefficients of a linear regression is added to the Least Squares problem. As the penalty is smaller for estimates closer to the origin, the ridged estimates are "shrunken" from the Ordinary Least Squares (OLS) solutions. The authors show that an infinitesimal increase in the penalization parameter from the unpenalized estimates results in an improvement on the mean squared prediction error.

In more recent years, new penalties have been proposed to the ordinary least squares problem. The bridge regression (Frank and Friedman, 1993) generalizes the ridge in that the squared norm of the coefficients is substituted by a penalty T given by the L_γ-norm of the model coefficients, that is:

$$T(\beta) \quad = \quad \left[\sum_{j=1}^{p} |\beta_j|^\gamma \right]^{\frac{1}{\gamma}} = \|\beta\|_\gamma$$

The regularization path for the bridge estimate can vary a lot according to the value of γ. Intuitively, the behavior of regularization path can be understood in terms of the penalty contour plot. For $\gamma \leq 1$, the penalty function causes some of the regressors are set to zero due to the presence of acute corners along the axes in these penalties contour plots. For $1 < \gamma < \infty$, the estimates tend to fall on regions of high "curvature" of the penalty function. Hence, for $1 < \gamma < 2$, the sizes of the coefficients tend to be very different, while for $2 < \gamma \leq \infty$ the sizes of the coefficients tend to be more similar. In the particular case $\gamma = \infty$, some of the coefficients tend to be exactly the same along the regularization path as a result of the acute corners on the contour plot along diagonal directions. Figure 1 shows the regularization path for the bridge regressions for different values of γ using the diabetes data presented in Efron et al. (2004).

When $\gamma \downarrow 0$ in the bridge regression, the penalty function becomes the "L_0-norm" of the coefficients: that is, the count of the number of parameters in the regression model. This case is of interest as model selection criteria defined in an information theoretical framework such as the AIC (Akaike, 1973), BIC (Schwartz, 1978), AIC_C (Sugiura, 1978) and gMDL (Hansen and Yu, 2001) can be thought of as particular points along the bridge regularization path. In this context, AIC and BIC have $\lambda = 2$,$\lambda =$ and $\lambda = \log(n)$ respectively, while gMDL chooses λ based on the data trying to strike a balance between the two and AIC_C tunes λ to adjust AIC to take the sample size into account. One inconvenient of the L_0-penalty is the combinatorial

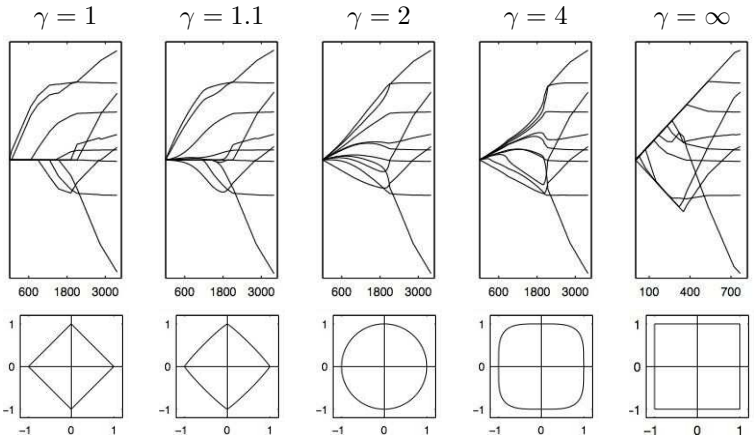


Figure 1: Regularization Paths of Bridge Regressions.
Upper Panel: Solution paths for different bridge parameters. From left to right: Lasso ($\gamma = 1$), near-Lasso ($\gamma = 1.1$), Ridge ($\gamma = 2$), over-Ridge ($\gamma = 4$), $\max(\gamma = \infty)$. The Y-axis has the range $[-800, 800]$. The X-axis for the left 4 plots is $\sum_i |\beta_i|$, the one for the 5th plot is $\max(|\beta_i|)$ because $\sum_i |\beta_i|$ is unsuitable. *Lower Panel*: The corresponding penalty equal contours of β_1 versus β_2 for $\|(\beta_1, \beta_2)\|_\gamma = 1$.

nature of the optimization problem, which causes the computational complexity of getting the estimates to grow exponentially on the number of regressors.

In that respect, the case of $\gamma = 1$ is of particular interest: while it still has the variable selection property, the optimization problem involved is convex. This represents a huge advantage in the computational point of view as it allows for the tools of convex optimization to be used in calculating the estimates (Boyd and Vandenberghe, 2004). This particularly case of bridge regression has deserved a lot of attention within the Statistics community, where it is popularly referred to as the Lasso (Tibshirani, 1996) as well as within the Signal Processing field, where it is more commonly referred to as basis pursuit (Chen and Donoho, 1994; Chen et al., 2001). Computationally efficient algorithms for tracing the regularization path for the $\gamma = 1$ case have been developed in recent years by Osborne et al. (2000) and Efron et al. (2004). One key property of the regularization path in this case is its piecewise linearity.

Even though the ability of the L_1 penalty to select variables in a model is a major advancement, some situations require additional structure on the selection procedure, especially when p is large. One such situation occurs in ANOVA regression models where some of the regressors are categorical. here, a factor is typically represented by a series of dummy variables. It is most desirable that the dummies corresponding to a factor be included into or excluded from the model simultaneously. The Blockwise Sparse Regression (Kim et al., 2006) and the GLasso (Yuan and Lin, 2006) and extensions (Meier et al., 2006)) provide ways of defining penalties that do grouped selection.

In other cases, the need exists for the variables to be added to the model in a particular ordering. For instance, in ANOVA models involving interaction among the factors, the statistician usually want to include higher order interaction between some terms once all lower order interactions involving those terms have been included to the model. In multi-resolution methods such as wavelet regression, it is desirable to only include a higher resolution term for a given region once the coarser terms involved in it have been added to

the model. The authors are not aware of any previous convex penalization method that has this ability.

The key idea in the construction of the CAP penalties is having different norms operating on the coefficients of different groups of variables – the group norms – and an overall norm that performs the selections across the different groups – the overall norm. Within each group of variables, the properties of the L_γ-norm regularizations paths presented above can be used to enforce different kinds of within-group relationship. Such relationships can also be understood to a certain extent through a Bayesian interpretation of the CAP penalties provided in Section 2.

To allow hierarchical selection, the groups can be constructed to overlap which, in conjunction with the properties of the use of L_γ-norms as penalties, cause the coefficients to become non-zero in specific orders.

In what concerns algorithms for tracing or approximating the regularization path, an important condition to be observed is convexity. We present sufficient conditions for convexity within the CAP family. For the convex members of the family, we propose the use of the BLasso (Zhao and Yu, 2004) as a means of approximating the regularization path for a CAP penalty in general. For some particular cases, very efficient algorithms are developed for tracing the regularization path exactly.

Even though cross-validation can be used for the selection of the regularization parameter, it suffers from some drawbacks. Firstly, it can be quite expensive from a computational standpoint. In addition, it is well suited for prediction problems but are not the tool of choice when data interpretation is the goal (Leng et al., 2004; Yang, 2003). We present an unbiased estimate for the number of degrees of freedom of the estimates along the regularization path for some particular cases of the CAP penalty. These results rely on a duality between the L_1 and L_∞ regularization and are based on the results by Zou et al. (2004).

The remainder of this paper is organized as follows. Section 2 present the CAP family of penalties. In addition to defining the CAP penalties, it includes a Bayesian interpretation for this penalties and results that guide the design of penalties for specific purposes. Section 3 provide a discussion on computational issues. It proves conditions for convexity and describes some of the algorithms involved in tracing the CAP regularization path. Section 4 presents unbiased estimates for the number of degrees of freedom for a subset of the CAP penalties. Simulation results are presented in section 5 and an application to a real data set is described in section 6. Section 7 concludes with a summary and a discussion on themes for future research.

2 The Composite Absolute Penalty (CAP) Family

In this section, we define the Composite Absolute Penalty (CAP) Family and explain the roles of the parameters involved in its construction. Specifically, we discuss how the group norms and the overall norm influence the CAP regularization path and how the overlapping of groups can lead to a hierarchical structure. After defining the CAP family of penalty functions, an interpretation of the Bayesian interpretation for the CAP penalties is provided. We then discuss some properties of bridge estimates that cause the grouping effects to take place. We end this section by describing the construction of CAP penalties for grouped and hierarchical variable selection.

2.1 Composite Absolute Penalties Definition

As the Composite Absolute Penalties (CAP) provide a framework to incorporate grouping or hierarchical information within the regression procedure, it assumes that information about the grouping and or order

of selection of the regressors is available a priori. Based on this prior information, K groups (denoted by $\mathcal{G}_k, k = 1, \ldots, K$) of regressors are formed and their respective coefficients are collected into K vectors. We shall refer to the vectors thus formed and their respective norms as:

$$\begin{aligned} \beta_{\mathcal{G}_k} &= (\beta_j)_{j \in \mathcal{G}_k}, k = 1, \ldots, K \\ N_k &= \|\beta_{\mathcal{G}_k}\|_{\gamma_k} \end{aligned}$$

Once $N_k, k = 1, \ldots, K$ are computed, they are collected in a new K-dimensional vector $\mathbf{N} = (N_1, \ldots, N_k)$ and using a pre-defined γ_0, the CAP penalty is computed by:

$$T(\beta) = \|\mathbf{N}\|_{\gamma_0}^{\gamma_0} = \sum_k |N_k|^{\gamma_0} \tag{1}$$

Once the CAP penalty is defined, its corresponding estimate is given as a function of the regularization parameter λ as:

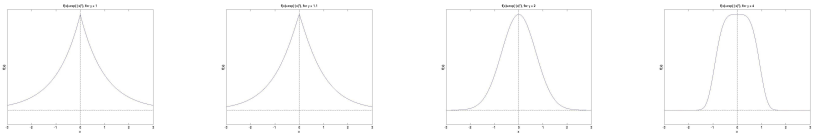
$$\hat{\beta}(\lambda) = \arg\min_{\beta} \sum_i L(Y_i, X_i, \beta) + \lambda \cdot T(\beta) \tag{2}$$

where L is a loss function as used described in the introduction above and T is a CAP penalty.

We now consider an interpretation for this family of functions.

2.2 A Bayesian Interpretation to CAP Penalties

When the loss function L corresponds to a log-likelihood function, a connection exists between penalized estimation and the use of Maximum a Posteriori (MAP) estimates. Letting L represent the log-likelihood of the data given the set of parameters β, T can be seen as the log of an *a priori* probability function and the penalized estimates can be thought of as the MAP estimates of the coefficients. This interpretation is helpful in understanding the role of the penalty function in the estimation procedure: it tends to favor solutions that are more likely under the prior. Within this idea, the ridge regression estimates can be thought of as assuming the error terms in the regression to have a Gaussian distribution given the coefficients in β and a Gaussian prior on β. The bridge estimates keep the Gaussian assumption on the data but use different priors according to the different γs. Figure 2 shows examples of different bridge priors. For $\gamma \leq 1$, the variable selection property may be thought of as arising from the kink at the origin for the corresponding densities.



(a) $\gamma = 1$ (b) $\gamma = 1.1$ (c) $\gamma = 2$ (d) $\gamma = 4$

Figure 2: Marginal prior densities on the coefficients for different values of γ

For the CAP penalties, this Bayesian interpretation corresponds to using the following a priori distribution assumption with density $g(\beta)$ given by:

$$g(\beta) = C_{\gamma_0,\gamma}^1 \exp\{-\sum_{k=1}^{K} (\|\beta_{\mathcal{G}_k}\|_{\gamma_k})^{\gamma_0}\} \tag{3}$$

where $C^1_{\gamma_0,\gamma}$ is a constant that causes $g(\beta)$ in (3) to integrate to 1. Even though (3) results in a well defined joint distribution for β, it does not provide much insight into what kind of structure CAP is promoting on the estimates at a first glance. A closer look, however, proves insightful.

The high-level view is that CAP priors operate on two levels. At the across-groups level, the components of the $\mathbf{N}$ vector of coefficients are independently identically distributed according to a density function f with $f_{\gamma_0}(x) \propto \exp(x^{\gamma_0})$. The intuitive role γ_0 plays operates on a group level in the same fashion as the bridge parameter: for $\gamma_0 \leq 1$ group sparsity is promoted in that some of the group norms N_k are set to zero; for $1 < \gamma_0 < 2$, dissimilarity across the group norms is encouraged, while $2 < \gamma_0 \leq \infty$ promotes similarity across group norms.

Once $\mathbf{N} = (\|\beta_{\mathcal{G}_1}\|_{\gamma_1}, \ldots, \|\beta_{\mathcal{G}_K}\|_{\gamma_K})$ has been sampled from f_{γ_0}, define the scaled coefficients $\frac{\tilde{\beta}_{\mathcal{G}_k}}{\|\tilde{\beta}_{\mathcal{G}_k}\|_{\gamma_k}}$. Under the assumption that the groups do not overlap, these scaled coefficients can be proven to be independently and uniformly distributed on the unit sphere defined by the L_{γ_k} norm. As a result, within the each group k, the smaller γ_k the more the coefficients of that group tend to concentrate closer to the coordinate axis, while the larger γ_k the more the coefficients concentrate along the diagonals.

This intuition about the CAP penalties for non-overlapping groups is made rigorous in lemma 1 and proposition 1 below:

Lemma 1 *Assuming $\tilde{\beta}$ follows the joint distribution (3) and that $\mathcal{G}_k \cap \mathcal{G}_{k'} = \emptyset$ whenever $k \neq k'$, the following holds:*

- *groups $\tilde{\beta}_{G_k}$ are independent w.r.t. each other;*
- *for any k the normalized group norm $\frac{\tilde{\beta}_{G_k}}{\|\tilde{\beta}_{G_k}\|_{\gamma_k}}$ is conditionally independent of $\|\tilde{\beta}_{G_k}\|_{\gamma_k}$;*
- *the distribution of $\|\tilde{\beta}_{G_k}\|_{\gamma_k}$ does not depend on γ_k;*
- *the distribution of $\frac{\tilde{\beta}_{G_k}}{\|\tilde{\beta}_{G_k}\|_{\gamma_k}}$ does not depend on γ_0.*

Lemma 1 indicates each group's norm and its normalized members can be regularized separately and independently using different γ_0 and γ_k. Formally, we have the followng theorem:

Theorem 1 *Suppose β^* and β^{**} are independent r.v. in $\mathbb{R}^k$, where*

$$\begin{aligned} \beta^*_j &\overset{i.i.d.}{\sim} C^2_{\gamma_0} \exp\{-x^{\gamma_0}\} \qquad \text{and} \\ \beta^{**}_k &\overset{i.i.d.}{\sim} C^2_{\gamma_k} \exp\{-x^{\gamma_k}\} \qquad \text{for } j \in G_i \text{ independently across groups} \end{aligned}$$

Then the following two relations hold:

$$\|\tilde{\beta}_{G_i}\|_{\gamma_i} \overset{d}{=} \|\beta^*\|_{\gamma_0}, \tag{4}$$

$$\frac{\tilde{\beta}_{G_i}}{\|\tilde{\beta}_{G_i}\|_{\gamma_i}} \overset{d}{=} \frac{\beta^{**}_{G_i}}{\|\tilde{\beta}^{**}_{G_i}\|_{\gamma_i}}. \tag{5}$$

The relationship in (4) tells us that the distributions of the components of the vector $\mathbf{N}$ have are independent and identically distributed. Furthermore, it tells us how the size of the each component behaves given γ_0. Hence, for $\gamma_0 \leq 1$, the spike in the density of β^* at zero promotes group selection.

The right hand side in (5) defines an uniform distribution over unit sphere for the L_{γ_k}-norm. Thus, the normalized coefficients of the k-th group are uniformly distributed over the L_{γ_k} unit sphere. This provides a formal justification for the fact that the higher γ_k the more the coefficients in group k tend to be similar.

2.3 Designing CAP penalties

The Bayesian interpretation to the CAP estimates provide justification to the notion that the CAP estimates operate on two different levels: an across-group level and a within-group level. Still, it does not provide conditions that ensure that the variables within a group are selected to or dropped from the model simultaneously. To get such conditions, recall the definition of bridge estimates for a fixed γ:

$$\hat{\beta} \quad = \quad \arg\min_{\beta} \left[L(Z, \beta) + \lambda \cdot \|\beta\|_{\gamma} \right]$$

For convex cases (i.e., $\gamma \geq 1$), the solution to the optimization problem is fully characterized by the Karush-Kuhn-Tucker (KKT) conditions:

$$\frac{\partial L}{\partial \beta_j} = -\lambda \frac{\partial \|\beta\|_{\gamma}}{\partial \beta_j} = -\lambda \cdot \text{sign}(\beta_j) \frac{|\beta_j|^{\gamma-1}}{\|\beta\|_{\gamma}^{\gamma-1}}, \qquad \text{for } j \text{ such that } \beta_j \neq 0 \tag{6}$$

$$|\frac{\partial L}{\partial \beta_j}| \leq \lambda |\frac{\partial \|\beta\|_{\gamma}}{\partial \beta_j}| = \lambda \frac{|\beta_j|^{\gamma-1}}{\|\beta\|_{\gamma}^{\gamma-1}}, \qquad \text{for } j \text{ such that } \beta_j = 0 \tag{7}$$

From these conditions, it is clear that, for $1 < \gamma \leq \infty$, the estimate $\hat{\beta}_j$ equals zero if and only if the condition $\frac{\partial L(Y_i, X_i, \hat{\beta})}{\partial \beta_j}|_{\beta_j=0} = 0$ is satisfied. The loss function and its gradient are data dependent. When the distribution of $Z_i = (X_i, Y_i)$ is continuous, the probability that $\frac{\partial L(Y_i, X_i, \hat{\beta})}{\partial \beta_j}|_{\beta_j=0} = 0$ is satisfied is zero. Thus, the solution is sparse with probability 0 when $1 < \gamma < \infty$.

When $\gamma = 1$, however, the right side of (7) becomes a constant dependent on λ. As a result, the coefficients that contribute less than a certain threshold to the loss reduction are set to zero.

From these two situations, we conclude that, setting $\gamma > 1$ will cause all variables to be kept our or included in the model simultaneously while, setting $\gamma = 1$, results in just a subset of the variables being selected to the model.

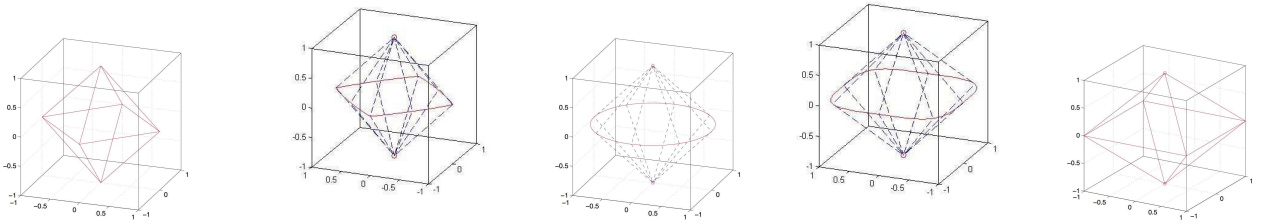
In the remainder of this subsection, we describe how to exploit these properties at the within-group and the across-groups levels to get grouped and hierarchical model selection. The designs below are meant to perform group selection and, thus, γ_0 is kept at one for the remainder of the paper.

2.3.1 CAP penalties for grouped selection

The goal in grouped model selection consists of letting the variables within a group in or out of the model simultaneously. From the Bayesian interpretation provided above, γ_0 should then be set to be 1. That will cause some of the terms of the norm vector of norms $\mathbf{N}$ to be set to zero and these groups are kept out of the model. Now, by setting $\gamma_k > 1$ for every group, the conditions for bridge estimates above ensure that with probability one, all variables within each group are included or excluded from the model simultaneously.

This definition of the CAP penalty provides not only for group selection: it allows the behavior of the coefficients within different groups to differ. In that sense, it is possible to have the coefficients in a group to encourage a restriction that all coefficients within a group are equal (by setting $\gamma_k = \infty$ in cases in group k if the effects of all variables in it are roughly of the same size) while not encouraging any particular direction for another group (by setting $\gamma_{\tilde{k}} = 2$ when no particular information on the relative effect sizes for variables in group $\tilde{k}$ is available). Following this principle, the Grouped Lasso penalty used by Yuan and Lin (2006) setting $\gamma_k = 2$ for all groups corresponds to the case where only the grouping information is used. As we will see in the simulation studies (section 5, grouping experiment 3), embedding extra information on the relative sizes of the groups may pay off in terms of the model error.

As in the bridge case, intuition about how the penalty operates can be derived from its contour plots. Figure 3 shows the contour plots for a simple case where the problem consists of choosing among three regressors with coefficients β_1, β_2 and β_3. We assume the variables 1 and 2 (with coefficients β_1 and β_2) form a group and variable 3 (coefficient β_3) forms a group of its own. The plots show how different levels of similarities are promoted by the use of different group-norms.



(a) $\gamma_0 = 1, \gamma = 1$ (Lasso) (b) $\gamma_0 = 1, \gamma = 1.1$ (c) $\gamma_0 = 1, \gamma = 2$ (d) $\gamma_0 = 1, \gamma = 4$ (e) $\gamma_0 = 1, \gamma = \infty$

Figure 3: Equal contour surfaces for different CAP penalties.
The X, Y and Z axes are β_1, β_2 and β_3 respectively. The solid lines indicate "sharp edges" of the surfaces, i.e. points where the CAP penalties are not continuously differentiable.

The solid lines in figure 3 plots correspond to points where the penalty is not differentiable. As in the Lasso, CAP estimates tend to concentrate on these points. When $\gamma = 1$, the CAP penalty reduces to the L_1 penalty case. In panel (a) we see the contour plot for this case. The grouping effect in this case is lost as variables within a group can come into the model on their own. This effect presents itself in panel (a) as a symmetry in the penalty contour plot. In panels (b) through (d), we see that setting γ_0 to 1 and $1 < \gamma \leq \infty$ causes the estimates to concentrate either on the "north" or "south" poles of the contours (group 2 composed of variable 3 is selected on its own) or on a L_γ unit sphere in the xy plane (in which case group 1 containing variables 1 and 2 is selected). As was the case in bridge, the higher γ, the more the coefficients in a group are similar. In panel (e) the limiting case where $\gamma = \infty$ is shown: the estimates within a group are encouraged to be exactly the same. In that case, even after the two groups are added to the model (low enough λ), the restriction $\beta_1 = \beta_2$ is still encouraged by the penalty function as shown by the solid lines along the xy diagonals.

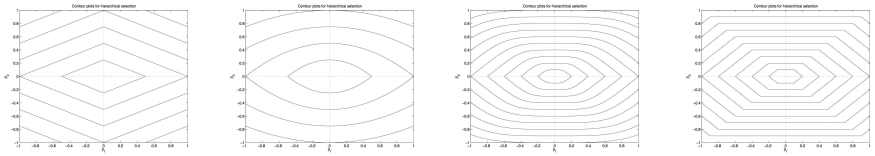
2.3.2 CAP penalties for hierarchical inclusion

In addition to grouping information, it is often the case that the analysis benefits from having variables entering the model is some prespecified order. Two such examples are: first, in the fitting of ANOVA models, it is usually the case that interactions between variables should only be included after its corresponding main effects are already in the model and; second, in the fitting of multiresolution models, one usually wants to prevent higher resolutions within a region to be added to the model before the coarser resolutions within the same area are added.

CAP penalties as defined above can be used to enforce the inclusion if variables in a model to take place in a given order by letting the groups overlap, that is, two different groups are allowed to contain the same variable. We start by considering a simple case and then extend the principle involved in the building of these penalties to tackle more interesting cases.

Consider a case where two variables X_1 and X_2 are to be selected in a specific order: variable 1 is to be included in the model before variable 2. To obtain this hierarchy, we define two groups $\mathcal{G}_1 = \{1, 2\}$ and $\mathcal{G}_2 = \{2\}$ and set $\gamma_0 = 1$, $\gamma_m > 1$ for $m = 1, 2$. That results in the penalty:

$$T(\beta) \quad = \quad \|(\beta_1, \beta_2)\|_{\gamma_1} + \|(\beta_2)\|_{\gamma_2} \tag{8}$$



(a) $\gamma_0 = 1, \gamma = 1$ (b) $\gamma_0 = 1, \gamma = 2$ (c) $\gamma_0 = 1, \gamma = 4$ (d) $\gamma_0 = 1, \gamma = \infty$

Figure 4: Contour plots for the penalty shown in (9).
It can be seen that for $\gamma > 1$ the coefficient β_2 tends to always be nonzero once feature 1 is added to the model.

The contour plots of this penalty function are shown in figure 4 for different values of $\gamma = \gamma_1$. As $\mathcal{G}_2$ contains only one variable, these contours are the same regardless of the value chosen for γ_2, since $\|\beta_2\|_{\gamma_2} = |\beta_2|$. The singularity points along the $\beta_2 = 0$ axis in panels (b) through (d) show that solutions with $\beta_1 \neq 0$ and $\beta_2 = 0$ tend to be encouraged by this penalty when $\gamma_1 > 1$. When $\gamma_1 = 1$, it is possible to have either $\beta_1 \neq 0$ and $\beta_2 = 0$ or $\beta_1 = 0$ and $\beta_2 \neq 0$ as the singularity points align along both axis. In this case, however, X_2 is less likely to be added to the model as it is more heavily penalized.

Algebraically, that fact follows from noticing that for $\gamma > 1$:

$$\frac{\partial}{\partial \beta_1} T(\beta) \quad = \quad \frac{\partial}{\partial \beta_1} \|\beta\|_\gamma = \text{sign}\,(\beta_1) \left(\frac{|\beta_1|}{\|\beta\|_\gamma} \right)^{(\gamma - 1)}$$

As a result, if $\beta_2 > 0$ and $\gamma > 1$, β_1 is locally not penalized at 0 and it will only stay at this point if the gradient of the loss function L is exactly zero for $\beta_1 = 0$. Unless the distribution of the gradient of the loss function has an atom at zero for β_1, $\beta_1 \neq 0$ with probability one.

The above steps can be used to prove a result that extends this simple setting to more interesting cases:

Theorem 2 *Suppose* $\gamma_0 = 1$ *and* $\gamma_k > 1, \forall k = 1, \ldots, K$ *and :*

- $\{2\} \subset \mathcal{G}_k \Rightarrow \{1\} \subset \mathcal{G}_k$ *for all* k *and*
- $\exists k^*$ *such that* $\{2\} \subset \mathcal{G}_{k^*}$ *and* $\{1\} \not\subset \mathcal{G}_{k^*}$

Then, $\frac{\partial}{\partial \beta_1} T(\beta) = 0$ *whenever* $\beta_2 \neq 0$ *and* $\beta_1 = 0$.

As a consequence, once β_2 enters the model, there is no increase in the penalty for moving β_1 away from zero locally. It follows that Theorem 2 provides a general recipe for defining groups resulting in variable selection following a specific ordering. One important case is when the hierarchical structure can be represented in the form of a graph. In the experimental section, two such examples are provided: wavelet regression and ANOVA with interaction terms. Figures 5 and 6 show the graphs for their respective hierarchical structures.

For these graphs, each group in the penalty is formed by a variable and all its descendants. As a result, there exists a group for each variable in the model. Nodes with no offsprings are penalized on their own. As before, the penalty for the m-th group is given by the L_{γ_m}-norm of the vector of coefficients corresponding to that group. The overall penalty is then formed by computing the L_{γ_0}-norm of the vector of group-norms. Setting $\gamma_0 = 1$, that leads to the following penalty function:

$$T(\beta) \quad = \quad \sum_{m=1}^{\text{nodes}} \| \left(\beta_m, \beta_{\text{descendants of } m}\right) \|_{\gamma_m} \tag{9}$$

3 Computation of the CAP Regularization Path

Once the CAP penalty is defined, computational methods are needed to get the CAP estimate for a data set. As a suitable value for the penalty parameter λ is not known beforehand, there is the need to be able to compute CAP estimates for various different values of λ efficiently. In this section, we consider the problem of tracing the entire regularization path for CAP estimates, while in the next we discuss the selection of a suitable λ. We point out, however, that these problems should ideally be tackled simultaneously: computing a criterion for evaluating λ as the path is traced allows some computational savings as it avoids the entire path to be calculated. That is the idea behind early stopping in boosting (e.g. Zhang and Yu, 2005).

Before describing the algorithms used to trace the CAP path, we tackle a common concern when dealing with optimization problems: ensuring the convexity of the objective function. Subsection 3.1 presents conditions ensuring that the CAP Lagrangian to be convex. Once that is done, section 3.2 discusses the use of the BLasso algorithm (Zhao and Yu, 2004) for approximating the CAP regularization path for a general convex CAP penalty. We notice that alternative algorithms may be used (e.g. Rosset, 2004). For some particular cases of grouped and hierarchical selection, the CAP regularization path can be proven to be piecewise linear (Rosset and Zhu, 2006). For some of these cases, algorithms that trace the CAP regularization path exact and efficiently are presented in 3.3.

3.1 Convexity of the CAP program

Convexity is an important property when dealing with optimization procedures as it ensures computational tractability to the problem (Boyd and Vandenberghe, 2004). It has enjoyed an increasing importance in

statistics as various statistical procedures are defined in terms of optimization problems (e.g. Bartlett et al., 2006, and references therein). The CAP estimates are no exception. In this subsection, we present a simple result that justifies the use of the algorithms for tracing the regularization path of the CAP estimates in what follows.

The key point for establishing the convexity for the CAP estimates consists of noticing that its properties closely parallel that of norms. The properties that causes proper norms to be convex are the scaling and the triangular inequality properties. Those properties are inherited from the norms used to define the CAP as stated in

Lemma 2 *If $\gamma_i \geq 1, \forall i = 0, ..., K$, then $T(\beta)$ as defined in (2) satisfies:*

1. *(Scaling): Let $\alpha \in \mathbb{R}$ and $\beta \in \mathbb{R}^p$. Then $T(\alpha \cdot \beta) = |\alpha| T(\beta)$;*
2. *(Triangular inequality): Let $\beta_1, \beta_2 \in \mathbb{R}^p$. Then $T(\beta_1 + \beta_2) \leq T(\beta_1) + T(\beta_2)$.*

Once these properties are proved, the following result follows easily:

Theorem 3 *If L is convex in β and $\gamma_i \geq 1, \forall i = 0, ..., K$, then the CAP objective function is convex in β.*

For cases where the the CAP objective function is known to be convex, the KKT conditions are known to be necessary and sufficient for optimality. Thus, algorithms that work by using the KKT conditions can be used for tracing the CAP regularization path. We describe some of them in the remainder of this section.

3.2 Using the BLasso

When both the penalty and the loss functions are convex, the BLasso algorithm (Zhao and Yu, 2004) provides an approximation for the regularization path of a regularization method. It operates similarly to boosting in that estimates are formed by taking steps in the model parameter space. Given its stepwise nature, it is similar to Forward Stagewise Fitting (FSF Efron et al., 2004). However, instead of solely relying on the gradients of the loss function to trace the path, this algorithm also takes the penalty function into account: before moving forward in any direction, a check is made on whether it is worth trading a little increase in the loss function for a reduction in the penalty. By introducing these *backward steps*, the BLasso can approximate the L_1-penalty regularization path arbitrarily close provided the step size is allowed to be reduced.

One great advantage of the BLasso algorithm is that, at each step, no computational effort on the inversion of matrices or the calculation of Hessian and gradients for the objective function. The price to pay for the computational simplicity is that the results are approximate. However, under convexity and differentiability of the objective function, the approximation can be made arbitrarily precise by reducing the size of the step taken at each iteration as shown in the original paper.

Another benefit of the BLasso algorithm is the ease of implementation of the algorithm. Similarly to boosting, once the loss and the penalty functions have been implemented, the BLasso can be adapted to solve the new problem with little effort.

In the experimental section below, the BLasso algorithm is used to approximate the regularization paths for different set ups of the CAP penalties. In certain cases, however, the regularization is known to be piecewise linear and more efficient algorithms can be used. That is where we turn our attention now.

3.3 L_1 and L_∞-norms and Piecewise Linearity

Even though the BLasso is versatile enough to handle a broad range of loss and penalty functions, in some particular cases, algorithms that compute the regularization path exactly and very fast can be developed. In this subsection, algorithms for tracing the CAP regularization path for the squared error loss for two specific penalties are proposed. In both cases, the property behind the possibility of constructing these algorithms is the piecewise linearity of the regularization path (Rosset and Zhu, 2006).

The two particular cases for which we develop specific algorithms have $\gamma_0 = 1$ and $\gamma_k = \infty, \forall k = 1, \ldots, K$. The first case we consider is that of nonoverlapping groups. In that case, the algorithm operates as the Lasso on the group level and as a L_∞-regularized regression within the group level. The second one deals with the particular case where the hierarchical structure can be represented as a graph (see figures 5 and 6). In this case, the algorithm can be interpreted as operating similarly to the nonoverlapping groups case, but the groups are defined dynamically along the path. The way in which the L_∞-regularization operates is a common thread in both cases so it is briefly described first.

The regularization path for L_∞: We now describe how to trace the regularization path for the L_∞-penalized squared error loss regression, as it will constitute a building block for the algorithms for grouped and hierarchical selection described below. The Lagrangian for this optimization problem defining the L_∞-regularized least squares estimate is given by:

$$\mathcal{L}(b, \lambda) = Y'Y + \left(\lambda \cdot \text{signs}(b) \cdot \mathbf{I}(\|b\|_\infty) - 2Y'X \right) b + b'X'Xb$$

where $\mathbf{I}(\|b\|_\infty)$ is a vector whose $j-th$ component equals $\text{sign}(b_j)$ if $|b_j| = \|b\|_\infty$ and 0 otherwise. Defining:

$$\mathcal{U}_\lambda = \{j : |b_j| < \|b\|_\infty\} \tag{10}$$

$$\mathcal{R}_\lambda = \{j : |b_j| = \|b\|_\infty\} \tag{11}$$

For simplicity, we define p_r and p_u to be the number of indices in $\mathcal{R}_\lambda$ and $\mathcal{U}_\lambda$ respectively and rearrange the columns in X so that $\mathcal{R}_\lambda$ and $\mathcal{U}_\lambda$ are such that $X = \left[\begin{array}{cc} (X_{\mathcal{R}_\lambda})_{n \times p_r} & (X_{\mathcal{U}_\lambda})_{n \times p_u} \end{array} \right]$. We then define:

$$(\mathcal{S}_\lambda)_{p_r \times 1} = \text{signs}\left(X'_{\mathcal{R}_\lambda} (Y - X\hat{\beta}_\lambda) \right) \tag{12}$$

$$(\mathcal{X})_{n \times (p_u+1)} = X \cdot \left[\begin{array}{cc} S_\lambda & \mathbf{0}_{p_r \times p_u} \\ \mathbf{0}_{p_u \times 1} & I_{p_u} \end{array} \right] \tag{13}$$

a necessary and sufficient condition for $\hat{\beta}_\lambda$ to be the L_∞ penalized estimate for λ is that the following conditions hold:

$$\hat{\beta}_\lambda = \left[\begin{array}{cc} S_\lambda & \mathbf{0}_{p_r \times p_u} \\ \mathbf{0}_{p_u \times 1} & I_{p_u} \end{array} \right]' \cdot \hat{\alpha}_\lambda \tag{14}$$

$$\mathcal{X}'\mathcal{X}\hat{\alpha}_\lambda = \mathcal{X}'Y - \lambda \cdot \text{sign}\left(Y - \mathcal{X}\hat{\alpha}_\lambda\right) \tag{15}$$

We notice that $X\hat{\beta}_\lambda = \mathcal{X}\hat{\alpha}_\lambda$ so the L_∞ penalized fit for λ is given equivalently by $X\hat{\beta}_\lambda$ and $\mathcal{X}\hat{\alpha}_\lambda$.

From these conditions, it is possible to interpret the estimates at a given λ as follows. There are two groups of coefficients. The first group ($\mathcal{R}_\lambda$) is that whose coefficients equal $\|\hat{\beta}_\lambda\|_\infty$ in absolute value. The

coefficients in the second group ($\mathcal{U}_\lambda$) away from the $\|\hat{\beta}_\lambda\|_\infty$ boundary and must be orthogonal to the current residuals, that is, must satisfy $X_j'(Y - X\hat{\beta}_\lambda) = 0$.

Using this interpretation of of the optimality conditions, it is possible to construct a path tracing algorithm. Starting from zero, all coefficients are moved by the same amount and with signs according to those of their correlations to the response. Once one of the correlation of one of the variables reaches zero, the corresponding coefficient is moved to the $\mathcal{U}_\lambda$ group. The direction of movement is then updated such that the variables in $\mathcal{U}_\lambda$ keep their zero correlation to the residuals. The remaining variables should move by the same amount according to the sign of their correlation with the residuals. The other condition to be observed is that, once a coefficient in the $\mathcal{U}_\lambda$ group touches the $\|\hat{\beta}_\lambda\|_\infty$ boundary, it should be let into the $\mathcal{R}_\lambda$ group.

It is interesting to point out that a duality exists between the Lasso problem and the L_∞-regularization problem. The correlations along the Lasso path behave similarly to the coefficients along the L_∞-regularized path and *vice-versa*. The correlations and the coefficients can be shown to be dual variables by expressing the L_∞-regularization problem (alt. the Lasso) as a quadratic programming problem and getting its dual to be the Lasso (alt. the L_∞-regularization problem). In Appendix A, the duality between these two problems is made precise.

Having dealt with the L_∞ penalized regression, we now move on to consider how to use it in the grouped selection case.

The algorithm for nonoverlapping groups, $\gamma_0 = 1$ and $\gamma_k = \infty, \forall k = 1, \ldots, K$. As was the case with the L_∞ regularization path, the path when $\gamma_0 = 1$ and $\gamma_k = \infty$, for all k is known to be piecewise linear (Rosset and Zhu, 2006). As mentioned before, this case can be interpreted as having a Lasso type of selection on the group level and an L_∞ penalization on the within group level.

We first define *group correlation* to a set of residuals as $c_k(\beta) = \|X_{\mathcal{G}_k}'(Y - X\beta)\|_1$. This group correlation plays the role of the variable correlation in the Lasso . We also define $\mathcal{A}_\lambda$ to be the set of active groups and $\mathcal{R}_{\lambda,k}$ to be the correspondent within each group of R_λ in the previous section. We define $S_{k,\lambda} = \text{sign}(X_{\mathcal{R}_{k,\lambda}}(Y - X\hat{\beta}_\lambda))$.

As in the Lasso, for $\lambda > \max_k \|X_{\mathcal{G}_k}'Y\|_1$, $\hat{\beta}_\lambda = 0$. Hence the algorithm starts by setting $\hat{\beta}_\lambda = 0$ for $\lambda = \max_k \|X_{\mathcal{G}_k}'Y\|_1$. At this point, the most correlated groups are added to A_λ. The estimates then move in a direction $\Delta\hat{\beta}$ such that:

- if $k \notin \mathcal{A}_\lambda$, then $\Delta\hat{\beta}_{\mathcal{G}_k}(\lambda) = 0$;
- for $k \in \mathcal{A}_\lambda$, $\Delta\hat{\beta}_{\mathcal{R}_{k,\lambda}} = \alpha_k \cdot S_\lambda$ and $X_{\mathcal{U}_{k,\lambda}}' \left(Y - X(\hat{\beta}(\lambda) + \Delta\hat{\beta})\right) = 0$, where α_k is chosen so that:

$$c_k(\hat{\beta}(\lambda) + \delta \cdot \Delta\hat{\beta}) \quad = \quad c_{k^*}(\hat{\beta}(\lambda) + \delta \cdot \Delta\hat{\beta}) \qquad \text{for all } k, k^* \in \mathcal{A}_\lambda$$

The sets $\mathcal{A}_\lambda$, $\mathcal{R}_{k,\lambda}$ and $\mathcal{U}_{k,\lambda}$ are kept constant between breakpoints. Once the direction of movement is determined, it is necessary to determine for how long to move. We choose δ to be the least positive number such that one of the following conditions holds:

- $c_{k^*}(\hat{\beta}_\lambda + \delta \cdot \Delta\hat{\beta}) = c_k(\hat{\beta}_\lambda + \delta \cdot \Delta\hat{\beta})$ for some $k^* \notin \mathcal{A}_\lambda$ and $k \in \mathcal{A}_\lambda$, in this case any $k^* \notin \mathcal{A}_\lambda$ for which the condition holds is added to $\mathcal{A}_\lambda$;

- $\|\hat{\beta}_{\mathcal{G}_k}(\lambda) + \delta \cdot \Delta\hat{\beta}\|_\infty = 0$ for some $k \in \mathcal{A}_\lambda$, in this case, any k satisfying the condition is removed from $\mathcal{A}_\lambda$;
- For some $m \in \mathcal{G}_k$ with $k \in \mathcal{A}_\lambda$, $X'_m \left(Y - X(\hat{\beta}(\lambda) + \delta \cdot \Delta\hat{\beta})\right) = 0$, in which case any m satisfying the condition is moved from $\mathcal{R}_{k,\lambda}$ to $\mathcal{U}_{k,\lambda}$ of its group;
- For some $m \in \mathcal{G}_k \cap \mathcal{U}_{k,\lambda}$ with $k \in \mathcal{A}_\lambda$, $|\hat{\beta}_m(\lambda) + \delta \cdot \Delta\hat{\beta}_m| = \|\hat{\beta}_{\mathcal{G}_k}(\lambda) + \delta \cdot \Delta\hat{\beta}_{\mathcal{G}_k}\|_\infty$, in which case any m satisfying the condition is moved from $\mathcal{U}_{k,\lambda}$ to $\mathcal{R}_{k,\lambda}$;

Having defined how to move from one breakpoint to another and identified the conditions that characterize the breakpoints and what actions need to be taken once they are reached, the algorithm for grouped selection with $\gamma_0 = 1$ and $\gamma_k = \infty$ for all k is fully described.

The algorithm presented here is only adequate for the squared error loss function. However, they can be extended for loss functions corresponding to members of the GLM family by adapting the contributions from (Park and Hastie, 2006). In these situations, the regularization path is no longer piecewise linear, but piecewise smooth. The algorithm is in general terms similar to the one for the squared error loss in which a quadratic approximation for the loss function is used. In this approximation, the gradient and the Hessian play the role of the correlations and the matrix $(X'X)$ respectively. The λ corresponding to the next "event" can then be approximated and its corresponding solution computed. If indeed an event takes place at this new λ, the active and orthogonal sets are updated accordingly. If no event takes place at the current point, the Hessian and gradient are updated and the approximation recomputed.

The algorithm for nested groups, $\gamma_0 = 1$ and $\gamma_k = \infty, \forall k = 1, \ldots, K$. In this section, we consider an algorithm for tracing the exact regularization path for hierarchical selection for the squared error loss function, $\gamma_0 = 1$, $\gamma_k = \infty$ for all k. The KKT conditions in this case reveal that optimality in this case is somewhat similar to proceeding as in the nonoverlapping groups case. The difference is that the groups now change dynamically along the path. The algorithm is complicated to be described in detail, here we give a high level description, readers who are interested in implementing the algorithm can refer to the code available at `http://www.stat.berkeley.edu/~pengzhao/CAP/`.

The algorithm is started by forming groups so that the following conditions hold:

- Each group consists of a sub-tree
- Interpreting each of the subtrees formed as a supernode, the tree formed by these supernodes must satisfy the condition that the average correlation between $Y - X\hat{\beta}(\lambda)$ and X's within the supernode is higher than that of all its supernode descendants.

Once these groups are formed, the optimality conditions are preserved by moving the coefficients in the root group as this is the group with the highest average correlation. The algorithms proceeds moving these coefficients until a breakpoint is reached. The breakpoints are characterized by:

- If the average correlation between $Y - X\hat{\beta}(\lambda)$ and a subtree $\mathcal{G}_a$ matches that of the supernode that contains the subtree, then $\mathcal{G}_a$ splits into a new supernode.
- If is a is a subtree whose corresponding supernode is a descendent of the supernode corresponding to subtree $\mathcal{G}_b$ and $\|\hat{\beta}_{\mathcal{G}_a,\lambda}\|_\infty$ becomes equal to $\|\hat{\beta}_{\mathcal{G}_b,\lambda}\|_\infty$, then combine groups $\mathcal{G}_a$ and $\mathcal{G}_b$ into a new supenode.

- If the coefficients for a supernode are zero and the average correlation of its elements to $Y - X\hat{\beta}(\lambda)$ becomes as large as its parent in the tree of supernodes, then the two supernodes should be merged.

As we now have means of tracing the regularization path for the CAP family, we now turn our attention to the problem of selecting the regularization parameter.

4 Choosing the regularization parameter for CAP penalties

Once the CAP penalty has been defined and its regularization path computed, one must make a decision on which estimate along the path to choose. At this point, the goal of the analysis must be taken into account as it is rarely possible that the same choice is optimal for prediction and structural inference purposes simultaneously (Yang, 2003).

If prediction is the goal of the model, cross-validation provides one way of selecting the regularization parameter. In this context, M-fold cross-validation works by splitting the available sample into M different groups. In each fold, the regularization path is traced by keeping the data points of one group out. The prediction errors are then computed on this out-of-sample data. The procedure is repeated for each group and the mean squared prediction error (MSPE) is computed over the M paths aligned according to the value of the penalty function T. Once an optimal $\tilde{T}$ is chosen, the path for the entire data is traced and the estimate with $T(\hat{\beta}(\lambda)) = \tilde{T}$ is chosen. We refer the reader to Efron and Tibshirani (1994) for more on cross-validation.

One drawback in using cross-validation is the computational effort involved in it: we now need to trace the regularization path for M different data sets. As a result, even when prediction is the goal of the analysis, it may be desirable to use an information criterion suitable for this purpose such as the AIC (Akaike, 1973) and AIC_C (Sugiura, 1978).

Another problem with cross-validation arises when the goal of the analysis is to unveil the structure among the variables in the model . In that situation, cross-validation is known not to be consistent in model selection (e.g. Leng et al., 2004). In this situation, other information criteria such as the BIC (Schwartz, 1978) provide a more suitable procedure for selecting a model from the path.

These two drawbacks with cross validation – namely: its computational intensiveness and its inability to choose structural models consistently – motivate the search for an unbiased estimate of the number of degrees of freedom of each estimate. Once these are available, different information criteria can be used to select a model according to the its intended use.

In this section, unbiased estimates for the number of degrees of freedom for some members of the CAP family are provided: namely the case of nonoverlapping groups with $\gamma_0 = 1$ and $\gamma_k = \infty, \forall k = 1, \ldots, K$ and squared error loss. These estimates are born from an adaptation of the results of Zou et al. (2004) on the degrees of freedom of the Lasso. We start off by considering the case of L_∞ regularized regression as it provides some insight into the more general case. The result is then extended to the case of nonoverlapping groups with $\gamma_0 = 1$ and $\gamma_k = \infty, \forall k = 1, \ldots, K$. Results for the general CAP penalty will be the theme of future research.

Number of degrees of freedom for the L_∞-penalized regression: As is the case with the L_1-norm, the L_∞-norm has some kinks on its contour plots. That leads to similarities between the two procedures.

The two procedures cause some restrictions to be enforced in higher penalized regions. While the L_1-penalty tends to set some of the coefficients to zero, the L_∞-penalty tend to restricted the coefficients to be the same. That provides the intuition for the estimates of the number of degrees of freedom in this setting.

An interpretation for the results from Zou et al. (2004) is that an unbiased estimate for the number of degrees of freedom is given by the number of regressors minus the number of enforced restrictions on a point along the path. The same is true for the L_∞-penalized regression, yielding the following estimate for the number of degrees of freedom along the L_∞-regularization path:

$$\hat{\mathrm{df}}(\lambda) \quad = \quad |\mathcal{U}_\lambda| + 1$$

where $\mathcal{U}_\lambda$ is as defined in section 3 above.

The proof of this fact follows the same steps as the one in Zou et al. (2004). It first restricts attention on a set $\mathcal{K}_\lambda \subset \mathbb{R}^n$ such that $y \in \mathcal{K}_\lambda$ ensures that λ is not a breakpoint. Within this set, conditions that ensure the conditions on the fit for Stein's lemma to hold are proven. The results are then extended for all $y \in \mathbb{R}^n$. A more detailed proof is shown in Appendix A.

Number of degrees of freedom for nonoverlapping groups with $\gamma_0 = 1$ and $\gamma_k = \infty, \forall k = 1, \ldots, K$: In this case, the penalty function can be thought of as promoting restrictions on two levels: on the higher level each group tends to have its coefficients set to zero and; within the group level, the coefficients tend to be set to the same value exactly. As a result, the number of degrees of freedom can again be estimated by subtracting the number of enforced restrictions from the total number of regressors. An alternative interpretation is that groups that are kept out at a given λ do not contribute any degrees of freedom for that estimate and selected groups only contribute their number of free parameters as in the L_∞ regularization case. That leads to the following expression for the number of degrees of freedom for the CAP:

$$\hat{\mathrm{df}}(\lambda) \quad = \quad \sum_{k \in \mathcal{A}_\lambda} (|\mathcal{U}_{k,\lambda}| + 1) = |\mathcal{A}_\lambda| + \sum_{k \in \mathcal{A}_\lambda} |\mathcal{U}_{k,\lambda}|$$

A formal proof for this estimate is provided in Appendix A. The proof follows roughly the same steps as the ones used in the proof for the L_1 and L_∞ case are present.

As all elements involved in getting a CAP estimate have been considered, we now move on to reviewing how the method perform in a series of experiments.

5 Simulation results

In the previous sections we have defined the CAP estimates, provided algorithms for computing its regularization path and discussed how to select the regularization parameter. Here we examine CAP's performance on a series of simulated examples. The section is divided into two parts: in the first, the performance of the CAP estimates for grouped selection is evaluated, while the second part is devoted to assessing the CAP performance for hierarchical variable selection.

The estimates will be evaluated according to two criteria: the model error defined by

$$ME \quad = \quad (\hat{\beta}_\lambda - \beta)' \mathrm{E}\,(X'X)\,(\hat{\beta}_\lambda - \beta)$$

and the sparsity of the model as measured by the number of nonzero variables selected. For the grouping examples an estimate of the number of degrees of freedom is also presented when available, that is, whenever $\gamma_k = \gamma, \ldots, K$. For the grouping examples, the number of degrees of freedom in the case where $\gamma = \infty$ are also reported and used to select CAP estimates based on different information criteria: BIC and AIC_C. We choose to report the BIC due to its goodness in selecting model consistently and the AIC_C due to its ability to pick good models for prediction in samples of reduced size.

Within the grouped selection subsection, two types of group information are considered: in the first, the group structure arises from a clear cut relationship among the variables (they are dummies representing the same categorical variable); in the second, the grouping information originates from some side information about the predicting variables. While in the first case there is no uncertainty about what grouping structure to use, in the second case we also consider what happens when the group structure is misspecified.

The hierarchical selection subsection also consider two cases: we first review the performance for CAP estimates when fitting wavelets to a signal and then consider the fitting of an ANOVA model with interactions between the factors. The main difference between these two cases refer to the graph structure of the hierarchy involved: in the first case, the groups are nested and, as a result, the efficient algorithm presented in Section 3 can be used when $\gamma_k = \infty, \forall k = 1, \ldots, K$; in the second case, the loopy nature of the hierarchy graph precludes the use of such algorithm.

For each of the simulated examples, we consider two different estimates: the *oracle* and the *validated* estimate. The oracle estimate is the one estimate along the path that minimizes the model error. It is not achievable in practice, but it serves as a lower bound on the performance for a given procedure. The validated estimate is obtained by hold-out validation. Both choices are made by considering the regularization path in 1,000 points equally spaced according to $T(\beta)$. If the estimates at this points are not on the path, they are computed by linearly interpolating the available points on the path. When an estimate for the degrees of freedom is available, the information criteria based selection only uses the training set which gives some advantage to the validated estimates when a comparison between the two is made.

In the cases presented next, we use the penalties presented in section 2 and hence keep $\gamma_0 = 1$ throughout. We leave the properties of penalties with $\gamma_0 \neq 1$ for future investigation. For each simulated example, we look at the cases $\gamma_k = \gamma$ for all $k = 1, \ldots, K$ and $\gamma \in \{2, 4, \infty\}$. In all cases, we consider the comparison to the Lasso. For the grouped cases, the Lasso estimate coincides with setting $\gamma = 1$.

In all examples that follow, the true process generating the data is of the form:

$$Y = X\beta + \sigma\varepsilon$$

with $\varepsilon \sim \mathcal{N}(0, I)$. According to the case, X will have some structure of which the CAP estimates try to take advantage.

We now provide an overview of the results from the different experiments. The experimental setup is explained in detail in the two subsections that follows. The detailed results are listed in Tables 3 to 10.

Grouping experiments results: In the grouping experiments, the inclusion of the group structure proves beneficial in terms of model error as for all simulated cases where $\gamma > 1$ there is a significant reduction in the model error both in terms of the oracle and the hold-out validated estimates. In cases where all the coefficients in a group are the same, the higher the coefficient γ the better the results in terms of model

error. In the example where the coefficients are not exactly equal but still of similar sizes, the use of $\gamma = 4$ has resulted in a reduced model error. We notice that even the extreme case $\gamma = \infty$ has a slight edge in terms of model error over the $\gamma = 2$ case (GLasso, Yuan and Lin, 2006).

In what concerns the number of variables added, what we see is that more variables are included in the CAP estimates with $\gamma > 1$ than when $\gamma = 1$ for both the oracle and the hold-out validated estimates. That is due to the grouping effect of the penalty: once a group is added to the model all its variables are picked. The average number of degrees of freedom for $\gamma = \infty$ suggests that, even though the number of included variables is larger than that for the Lasso, the within group regularization prevents the number of effective parameters from increasing rapidly.

Still regarding the grouping experiments, when the groups are correctly specified, the estimates obtained by using BIC have a number of variables which are on average close to the true number of variables in the model. This is not the case under the group misspecification considered: noisy variables are added to significant groups. Under these circumstances, the average size of the model selected is close to true number of variables in the model plus the number of noisy variables mistakingly added to the significant groups. The number of degrees of freedom for the model selected by BIC is also higher for the case where the groups are misspecified than for the correct group specification. That can be explained by the fact that the noisy variables are added to the $\mathcal{U}_{k,\lambda}$ sets of their respective groups early on the regularization path. We also notice that the BIC estimates result in higher model errors than the oracle, the hold-out validated and the AIC_C estimates especially in the cases where the sample size was severely limited.

The use of AIC_C has yielded estimates with model errors comparable to the ones resulting from using the oracle selection and cross-validation. In what concerns the selection of a correct specification for the model, it seems to have a worse performance than that of the BIC as the number of variables selected is on average much higher than the number of variables in the true model. That holds true both when the groups are correctly specified and when they are not.

Finally, for the specific form of misspecification considered, the use of the misspecified group information still resulted in advantages in terms of model error over the Lasso. However, misspeciying the groups causes some of these gains to be lost in the comparison to the Lasso ($\gamma = 1$ case). Additionally, if the model is not aimed at prediction, the group misspecification may prove more harmful as suggested by its influence on the number of variables selected by BIC.

Hierarchical wavelet fitting results: The results for the wavelet tree example are shown in Table 9. In terms of model error, the use of the hierarchical structure has results in a reduction in terms of model whenever $\gamma > 1$ in the comparison to the Lasso. The number of terms selected by the oracle and holdout validated estimates is larger than that selected by their corresponding estimates using the Lasso. However, as was the case with the group selection, this larger number of terms added to the model does not necessarily imply a larger number of degrees of freedom for the estimates as the CAP estimates are constrained to obey the hierarchical structure coded in the penalty function. Further research aiming at defining an estimate for the degrees of freedom in this case is desired to confirm this point. That would also open the way for procedures consistent in terms of model selection.

Hierarchical fitting for ANOVA: We evaluate the performance of hierarchical CAP penalties in the ANOVA case in a regression involving ten independently distributed standard normal variables and their respective interactions. Out of the ten variables, only four are present in the true model. Five cases are considered and they differ mainly in the relative size between the main and interaction effects. The coefficients of the four relevant variables and respective interactions are shown in table 1: in the first case, there are no interaction terms at all; for the second and third cases, the interaction terms are of moderate size as compared to the main effects; and in the last two cases, the main and interaction effects are about the same size.

The results for the ANOVA case suggest that the hierarchical procedure performs better when the interaction effects are of moderate size when compared to the main effects. In these situations (cases 1, 2 and 3), the resulting model error is significantly better than that of the Lasso. For larger interaction effects, however, the use of this family of CAP penalties results in slightly larger model errors than using the Lasso. In these situations, given the large effect of the interaction terms, it may not be optimal to keep the interaction terms from entering the model before the main effects as in some of the cases considered (e.g. β_3 and β_{13} in case 5) the interaction terms have larger coefficients than the main effects.

In what concerns the number of terms selected to the models, the hierarchical selection resulted again in less sparse models. Again, we point out that this does not necessarily translates into estimates with a higher number of degrees of freedom. As was the case with the estimates for the wavelet tree case, the CAP estimates here are constrained to satisfy the hierarchy represented in the penalty function. Further research towards an estimate of the number of degrees of freedom of the CAP estimates is called for as it may help clarify this point and allow for the use of criteria resulting in consistent model selection.

5.1 Grouping experiments

In this subsection we describe the experimental setup for evaluating the performance of CAP estimates in problems where a natural grouping exist. In the first example, the grouping arises from the fact that the different regressors are actually dummy variables used to represent categorical variables. In this situation, there is no uncertainty about the grouping to be used. In the last two examples, the grouping arises from some side information about the regressors: the fact that some of them correspond to different noisy measurements of the same underlying variable. In such a situation, there may be uncertainty about the right group structure to use. Thus, we simulate the CAP estimates for a correct and an incorrect grouping specification.

First grouping experiment: This first grouping case is taken from Yuan and Lin (2006). Define $Z_1, \ldots, Z_{15}$ to be normally distributed each with mean zero, variance one and $\mathrm{cov}(Z_i, Z_j) = \sigma_{ij} = 0.5^{|i-j|}$. The categorical factors $\tilde{Z}_i$ are then defined by:

$$\tilde{Z}_i = \begin{cases} 0, & \text{if } Z_i < \Phi^{-1}(\frac{1}{3}) \\ 1, & \text{if } Z_i > \Phi^{-1}(\frac{2}{3}) \\ 2, & \text{if } \Phi^{-1}(\frac{1}{3}) \leq Z_i \leq \Phi^{-1}(\frac{2}{3})) \end{cases}$$

And X is defined to be a set of dummy variables with $\left(X_{2\cdot(j-1)+1}, X_{2j}\right) = \left(\mathbf{I}(\tilde{Z}_j = 1), \mathbf{I}(\tilde{Z}_j = 2)\right)$. The true set of coefficients is:

$$\beta \quad = \quad \left(1.8, -1.2, 0, 0, 1, 0.5, 0, 0, 1, 1, \underbrace{0, \ldots, 0}_{20}\right)$$

The noise level is set to $\sigma = 1.476$ resulting in a signal-to-noise ratio (SNR) of 1.8. The train and validation sample sizes are respectively 50 and 25. There is no uncertainty about the grouping and it is set to be $\mathcal{G}_k = \{2k-1, 2k\}$, for $k = 1, \ldots, 15$.

Second grouping experiment: This example is taken from Zou and Hastie (2005). Here the group structure arises as some of the regressors corresponds to repeated measurements of some hidden variable Z:

$$\begin{aligned} X_i &= Z_1 + \varepsilon_i^x, \qquad Z_1 \sim \mathcal{N}(0,1), \qquad i = 1, \ldots, 5 \\ X_i &= Z_2 + \varepsilon_i^x, \qquad Z_2 \sim \mathcal{N}(0,1), \qquad i = 6, \ldots, 10 \\ X_i &= Z_3 + \varepsilon_i^x, \qquad Z_3 \sim \mathcal{N}(0,1), \qquad i = 11, \ldots, 15 \\ X_i &\sim \mathcal{N}(0,1), \qquad X_i \text{i.i.d}, \qquad i = 16, \ldots, 40 \end{aligned}$$

The true model is characterized by the coefficients:

$$\beta \quad = \quad \left(\underbrace{3, \ldots, 3}_{15}, \underbrace{0, \ldots, 0}_{25}\right)$$

and the noise level is set to $\sigma = 15$, yielding $SNR = 3.00$). The results are reported for two difference training sample sizes: 50 and 500. In both situations, the validation sample size is kept at 50. As in this sort of situation there can be uncertainty about the right grouping to use, we consider two alternative groupings:

Correct grouping	**Incorrect grouping**
$\mathcal{G}_1 = \{1, \ldots, 5\}$	$\mathcal{G}_1 = \{1, \ldots, 5, 16\}$
$\mathcal{G}_2 = \{6, \ldots, 10\}$	$\mathcal{G}_2 = \{6, \ldots, 10, 17\}$
$\mathcal{G}_3 = \{11, \ldots, 15\}$	$\mathcal{G}_3 = \{11, \ldots, 15\}$
$\mathcal{G}_k = \{k\}$, for $k = 16, \ldots, 40$	$\mathcal{G}_k = \{k\}$, for $k = 18, \ldots, 40$

Third grouping experiment: This example is similar to the second one but the number of replicates is larger and the groups are less similar to each other. Letting $w = 0.05$ and $\varepsilon^x \sim \mathcal{N}(0,1)$ and $\text{cov}(\varepsilon_i^x, \varepsilon_j^x) = 0.5^{|i-j|}$, X is given by:

$$\begin{aligned} X_i &= w \cdot Z_1 + \sqrt{1-w^2} \cdot \varepsilon_i^x, \qquad Z_1 \sim \mathcal{N}(0,1), \qquad i = 1, \ldots, 10 \\ X_i &= w \cdot Z_2 + \sqrt{1-w^2} \cdot \varepsilon_i^x, \qquad Z_2 \sim \mathcal{N}(0,1), \qquad i = 11, \ldots, 20 \\ X_i &= w \cdot Z_3 + \sqrt{1-w^2} \cdot \varepsilon_i^x, \qquad Z_3 \sim \mathcal{N}(0,1), \qquad i = 21, \ldots, 30 \\ X_i &= w \cdot Z_4 + \sqrt{1-w^2} \cdot \varepsilon_i^x, \qquad Z_4 \sim \mathcal{N}(0,1), \qquad i = 31, \ldots, 40 \\ X_i &= w \cdot Z_5 + \sqrt{1-w^2} \cdot \varepsilon_i^x, \qquad Z_5 \sim \mathcal{N}(0,1), \qquad i = 41, \ldots, 50 \end{aligned}$$

The true coefficients are given by:

$$\beta \quad = \quad \left(\underbrace{7, \ldots, 7}_{10}, \underbrace{2, \ldots, 2}_{10}, \underbrace{1, \ldots, 1}_{10}, \underbrace{0, \ldots, 0}_{20}\right)$$

The noise level is set at $\sigma = 19.22$ corresponding to a SNR=4.0. The training and validation sets contains 450 and 50 observations respectively. As above, it is interesting to look at the effect of group misspecification on the results:

Correct grouping	**Incorrect grouping**
$\mathcal{G}_1 = \{1, \ldots, 10\}$	$\mathcal{G}_1 = \{1, \ldots, 10, 31\}$
$\mathcal{G}_2 = \{11, \ldots, 20\}$	$\mathcal{G}_2 = \{11, \ldots, 20, 41\}$
$\mathcal{G}_3 = \{21, \ldots, 30\}$	$\mathcal{G}_3 = \{21, \ldots, 30\}$
$\mathcal{G}_4 = \{31, \ldots, 40\}$	$\mathcal{G}_4 = \{32, \ldots, 40\}$
$\mathcal{G}_5 = \{41, \ldots, 50\}$	$\mathcal{G}_5 = \{42, \ldots, 50\}$

5.2 Hierarchical selection of variables

In this subsection, we describe two situations under which hierarchical CAP penalties described in Section 2 will be evaluated. Two different cases are considered: the selection of variables in a multiresolution model (wavelet regression) and in an ANOVA model with interaction terms.

Hierarchical selection in Wavelet Regression: In this example, the true signal is given by a linear combination of Haar wavelets at different resolution levels. Letting Z_{ij} denote the Haar wavelet at the j-th position of level i, we have:

$$Z_{ij}(t) \quad = \quad \begin{cases} -1, & \text{if } t \in (\frac{j}{2^{i+1}}, \frac{j+1}{2^{i+1}}) \\ 1, & \text{if } t \in (\frac{j+1}{2^{i+1}}, \frac{j+2}{2^{i+1}}) \\ 0, & \text{otherwise} \end{cases}$$

The design matrix X is formed by sequentially adding the wavelets on a given level to the right of the current X starting from the coarsest level and an empty design. The true coefficients are given by:

$$\beta = \left(\underbrace{15}_{\text{level 0}} , \underbrace{7, 8}_{\text{level 1}} , \underbrace{-4, 6, 0, 0}_{\text{level 2}}, \underbrace{0, 0, 1, 2, 0, 0, 0, 0}_{\text{level 3}} \right)$$

Figure 5 shows a graphical representation of the hierarchy to be enforced. The shaded nodes correspond to components that are present in the observed signal this example.

We set $\sigma = 17.18$ so the signal to noise ratio is 1. The training and test set contain 4 and 1 observations for each of the 16 positions in which the signal is observed. Following the recipe laid out by (9) in section 2, we define the grouping by:

Group 1: All nodes			
Group 2: $\beta_{10}, \beta_{20}, \beta_{21}, \beta_{30}, \beta_{31}, \beta_{32}, \beta_{33}$		Group 3: $\beta_{11}, \beta_{22}, \beta_{23}, \beta_{34}, \beta_{35}, \beta_{36}, \beta_{37}$	
Group 4: $\beta_{20}, \beta_{30}, \beta_{31}$	Group 5: $\beta_{21}, \beta_{32}, \beta_{33}$	Group 6: $\beta_{22}, \beta_{34}, \beta_{35}$	Group 7: $\beta_{23}, \beta_{36}, \beta_{37}$
Group $8 + k$: β_{3k}, for $k = 0, \ldots, 15$			

Hierarchical selection in ANOVA models: We now consider another example where it is desirable that the inclusion of variables in a model happen according to a prespecified hierarchy. We consider an ANOVA regression where ten candidate regressors and respective interactions are included in a regression.

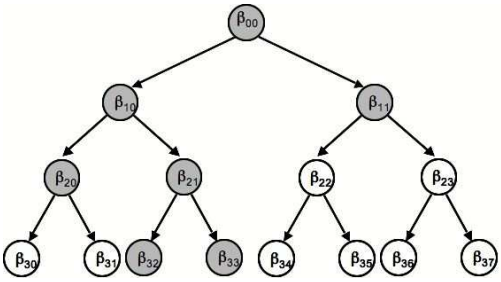


Figure 5: Hierarchy tree for the wavelet example (shaded nodes are the ones in the true model)

Each of the ten candidate variable is independently normally distributed with mean zero and variance 1. The true model only include four of these variables and some of its interactions. We consider five different cases with coefficients as shown below:

	x_1	x_2	x_3	x_4	x_1x_2	x_1x_3	x_1x_4	x_2x_3	x_2x_4	x_3x_4	σ	SNR
Case 1	7	2	1	1	0	0	0	0	0	0	3.7081	4.0
Case 2	7	2	1	1	0.5	0	0	0.1	0.1	0	3,7353	4.0
Case 3	7	2	1	1	1.0	0	0	0.5	0.4	0.1	3.8490	4.0
Case 4	7	2	1	1	5	0	0	4	2	0	6.8920	4.0
Case 5	7	2	1	1	7	7	7	2	2	1	11.4346	4.0

Table 1: Coefficients for the ANOVA experiment

These cases are intended to study the performance of the hierarchical penalties in cases where the relative size of the main effects and interaction effects vary over a broad range. All terms not shown in the table have no effect on the response variable. In all cases, the training and validation sets had respectively 200 and 100 observations.

Figure 6 depicts the hierarchy imagined in this case in a reduced problem with only four variables and respective interactions. The groups are defined following the recipe prescribed by (9) in section 2 resulting in 55 groups. The first ten groups each contains one of the regressors and all interaction terms where they appear. The remaining forty five groups are composed each by one of the interaction terms.

Contrary to happens in the tree example, the hierarchy graph now have loops which precludes the use of the efficient algorithm used for the $\gamma = \infty$ in the wavelet regression example. As a result, the BLasso algorithm was used throughout to trace the regularization paths except for the Lasso case.

6 CAP for Cloud Detection in Arctic Areas

In this section, we present the use of CAP in the problem of detecting clouds on arctic regions based on radiance measurements provided by the Multi-angle Imaging SpectroRadiometer (MISR) onboard NASA's Terra satellite. Clouds play a major role in Earth's climate. They can help cool down Earth's climate

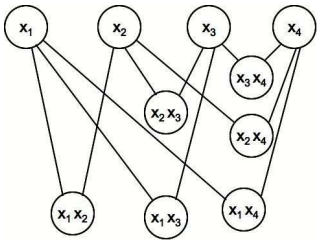


Figure 6: Hierarchy for the ANOVA example

by reflecting part of the incoming solar radiation back to the space, but can also contribute to a warmer climate by reflecting the some of the radiation emitted by the planet surface. The prediction of the future climate depends on a better understanding of these interactions which in turn require efficient algorithms for detecting clouds over polar regions (Shi et al., 2004).

Our goal here will be to classify into cloud or clear each 4×4 cluster of pixels in a satellite picture taken by the MISR. The data used is part of the data considered in Shi et al. (2004). The reader is referred to this reference for further details on the data and on this classification problem. Before we describe the use of CAP in this situation and the penalties used, details on the data are provided.

The data corresponds to two images collected by the MISR sensor over a region in the arctic cap. Each data point corresponds to a 4×4 cluster of pixels on each of these images. Two groups of variables are considered. The variables that can be used in the classification can be divided into two groups:

- raw MISR radiance measurements: these are radiance measurements collected by cameras pointed in five different angles in the forward direction: 70.5^o, 60^o 45.6^o 26.1^o and 0.0^o, referred to as DF, CF, Bf, AF and AN respectively;
- Three features developed by Shi et al. (2004) that factors in expert knowledge:
  - SD – the standard deviation of the radiance measured on the AN camera (based on a 8×8 cluster of pixels centered on the cluster to be classified);
  - CORR – the mean of the correlation between the radiance measured in the AN and BF cameras and those measured on the AN and AF angle;and
  - NDAI – the the Normalized Differential, which measures the forward scattering property of the scene,

The data consist of two pictures from a region in the arctic cap at two different instants. In the data used here, an expert label is available for each data point to be classified. The number of observations available for images 1 and 2 are respectively $68,306$ and $82,148$. For each picture we consider 4 replications of the following procedure: 300 data points are randomly selected to be the training set; the CAP path is traced for each of the cases described below; ten fold cross-validation is then used for selecting one CAP estimate and; the remaining points are used to compute the out-of-sample error rate. The sample size for each fitting is kept small to mimic the effects of high-dimensionality.

We define two groups in this case:

- group 1: a group of five variables containing the radiance measurements reported by MISR;

- group 2: a group with the three handcrafted variables developed by Shi et. al (2004).

Five different CAP penalties are considered. In all of them, $\gamma_0 = 1$. The five different penalties are:

- Penalty 1: $\gamma_k = \gamma = 1$ for all groups (Lasso);
- Penalty 2: $\gamma_k = \gamma = 2$ for all groups;
- Penalty 3: $\gamma_k = \gamma = 4$ for all groups;
- Penalty 4: $\gamma_k = \gamma = \infty$ for all groups;
- Penalty 5: $\gamma_1 = \infty$, $\gamma_2 = 2$;

The first penalty corresponds to variable selection as performed by the lasso in the context of logistic regression (Park and Hastie, 2006). Penalties 2, 3 and 4 consider the group information and encourage similarity among the coefficients of a group equally. Penalty 5 corresponds to a prior idea that the values of the radiance at the different angles are expected to influence the predicted probability by the same amount while not encouraging any particular level of similarity on the effects of the three handcrafted variables.

As the number of variables in each group is different, we make an adjustment during the normalization phase. First, each column is normalized to have mean zero and the same variance. Then, the variables in group 1 and 2 are multiplied by $\sqrt{5}$ and $\sqrt{3}$ respectively. That would make the total group correlation to be equal in the two groups in the case in which all variables have the same correlation to the response.

The mean error rate was used within cross-validation as the criterion to select a regularization parameter. As was the case in the simulations, each path in the cross-validation sample was computed (using interpolation when necessary) at 1,000 equally spaced points according to $T(\beta)$. The threshold for classifying a cluster as a cloud was set to 0.5.

The results are evaluated in terms of the out of sample error rate and are presented in table 2. We can see that different penalties perform better for the two different images considered. For the first image, it is advantageous to encourage different levels of similarities for the different groups. For the second image, the Lasso and penalty 5 perform similarly and better than the other penalties.

These results suggest that it is desirable to have a flexibility on how much the coefficients are encouraged to be similar.

	Image 1	Image 2
Penalty 1	0.1153 ± 0.0098	0.0278 ± 0.0020
Penalty 2	0.1081 ± 0.0041	0.0928 ± 0.0099
Penalty 3	0.1191 ± 0.0137	0.0801 ± 0.0032
Penalty 4	0.0963 ± 0.0037	0.0401 ± 0.0092
Penalty 5	0.0950 ± 0.0038	0.0300 ± 0.0032

Table 2: Results for using CAP on the cloud data set

7 Discussion

Based on the results from both the simulations and the experiment on real data , we conclude that using CAP penalties to incorporate structure among the covariates into the estimation procedure result in significant gains in prediction performance. This proves CAP to be both conceptually interesting and a practically useful regularization tool. In addition, the flexibility that CAP provided for adjusting the penalties to further regularize the relative sizes of the variables in a group is proved to be useful in the experiments.

From a computational standpoint, the CAP penalties that yielded piecewise linear regularization paths have clear advantages, as the algorithms in these cases run in speeds comparable to those of the LARS algorithm for the Lasso path. The algorithms developed here are also interesting from the methodological point of view as they provide further understanding of how the CAP penalties operate.

It is also of particular interests that even though the CAP estimates tended to be less sparse than those resulting from the Lasso, they make use of less degrees of freedom as the parameters corresponding to variables in an added group are still subject to the regularization imposed by the group norms. In that respect, the L_1 and L_∞-penalized regularization path may be used as starting points for methods that dynamically decide restrictions to impose on the estimates along the regularization path.

Finally, since Lasso suffer from instability under strong colinearity (Zou and Hastie, 2005; Zhao and Yu, 2006), the practice of grouping highly correlated variables together may prove beneficial in term of both prediction and model selection.

To formalize and theoretically study these ideas point towards our future work.

A Appendix: DFs for the L_∞-regularized regression

In this appendix, we provide a more detailed proof of the extension of the results on the number of degrees of freedom for the Lasso for the L_∞-norm regularized regression.

We first prove that the L_∞-penalized regression optimization problem is dual to the Lasso problem. We start by rewriting the optimization problem in the standard form of a quadratic program:

$$\begin{array}{rcll} \hat{\beta}(t) & = & \arg\min_b & \frac{1}{2}(Y - X\beta)'(Y - X\beta) \\ & & \text{st} & b_i \leq t, \qquad \forall i = 1, \ldots, p \\ & & & -b_i \leq t, \qquad \forall i = 1, \ldots, p \end{array}$$

Following Boyd and Vandenberghe (2004), define:

$$\begin{aligned} P(c) &= X'X \\ q(c) &= -X'Y + \sum_{i=1}^{p} c_i e_i - \sum_{j=1}^{p} c_{j+p} e_j \\ &= -X'Y + \mathbf{c}_1 - \mathbf{c}_2 \\ r(c) &= Y'Y + \sum_{i=1}^{p} c_i t - \sum_{j=1}^{p} c_{j+p} t \\ &= Y'Y + t \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} \begin{bmatrix} \mathbf{1}_p \\ -\mathbf{1}_p \end{bmatrix} \end{aligned}$$

so the dual problem is given by:

$$\begin{array}{ll} \min_c & \frac{1}{2}\left[\left(-X'Y+\mathbf{c}_1-\mathbf{c}_2\right)\left(X'X^{-1}\right)\left(-X'Y+\mathbf{c}_1-\mathbf{c}_2\right)\right]+Y'Y+t\left[\begin{array}{cc}\mathbf{c}_1 & \mathbf{c}_2\end{array}\right]\left[\begin{array}{c}\mathbf{1}_p\\ -\mathbf{1}_p\end{array}\right] \\ \text{st} & c_i\geq 0, \forall i=1,\ldots,2p \end{array}$$

Using some algebra, it can be shown that for each $t \geq 0$ there exists $K \geq 0$ that makes this problem equivalent to:

$$\begin{array}{ll} \min_c & \frac{1}{2}c'(X'X)^{-1}c-c'(X'X)^{-1}\left(X'Y\right) \\ \text{st} & \|c\|_1 \leq K \end{array}$$

which establishes the duality between the Lasso problem and the L_∞-regularized least squares regression. A little further analysis reveals that the dual parameters in $\mathbf{c}$ correspond to the correlations along the L_∞ path. It follows that the results that hold for the coefficients in the L_1 path must hold for the correlations along the L_∞ path and *vice-versa*.

Thus, letting the CAP fit be denoted by:

$$\hat{\mu}(\lambda, y) = X\hat{\beta}(\lambda, y).$$

the following three facts follow from the results of Efron et al. (2004) and Zou et al. (2004) to the Lasso.

Fact 1 *For each λ, there exists a set $\mathcal{K}_\lambda$ such that:*

- *$\mathcal{K}_\lambda$ is a the union of a finite collection of hyperplanes;*
- *if $Y \in \mathcal{C}_\lambda = \mathbb{R}^n - \mathcal{K}_\lambda$, then there is no break point at the regularization path at λ.*

Fact 2 *$\hat{\beta}(\lambda, y)$ is a continuous function of y for all λ.*

Fact 3 *If $y \in \mathcal{C}_\lambda$, then the sets $\mathcal{R}_\lambda$ and $\mathcal{U}_\lambda$ are locally invariant.*

From these three facts, we can prove:

Lemma 3 *For a fixed $\lambda \geq 0$, the fit $\hat{\mu}(\lambda, y) = X\hat{\beta}(\lambda, y)$ is uniformly Lipschitz on the set $\mathcal{C}_\lambda$ with:*

$$\|X\hat{\beta}(\lambda, y+\Delta y) - X\hat{\beta}(\lambda, y)\| \leq \|\Delta y\|, \qquad \textit{for sufficiently small } \Delta y$$

and:

$$\nabla \cdot \hat{\mu}(\lambda, y) = |\mathcal{U}_\lambda| + 1$$

Proof Lemma 3: We first recall that, in terms of the definitions in (10) through (15), $\hat{\mu}(\lambda, y) = \mathcal{X}\hat{\alpha}(\lambda, Y)$. Furthermore, from the optimality conditions for the L_∞ penalty we have that:

$$\begin{array}{rcl} \left(\mathcal{X}'\mathcal{X}\right)\hat{\alpha}(\lambda, Y) & = & \mathcal{X}'Y - \lambda \cdot \text{sign}\left(Y - \mathcal{X}\hat{\alpha}(\lambda, Y)\right) \\ & \text{and} & \\ \left(\mathcal{X}'\mathcal{X}\right)\hat{\alpha}(\lambda, Y+\Delta Y) & = & \mathcal{X}'(Y+\Delta Y) - \lambda \cdot \text{sign}\left(Y+\Delta Y - \mathcal{X}\hat{\alpha}(\lambda, Y+\Delta Y)\right) \end{array}$$

As $Y \in \mathcal{C}_\lambda$, ΔY can be chosen small enough so that the signs of the correlation of each index $k \in \mathcal{R}_\lambda$ is preserved. Thus subtracting the two equations above:

$$\left(\mathcal{X}'\mathcal{X}\right)\left(\hat{\alpha}(\lambda, Y + \Delta Y) - \hat{\alpha}(\lambda, Y)\right) \quad = \quad \mathcal{X}'\Delta Y$$

It follows that:

$$\hat{\mu}(\lambda, Y + \Delta Y) - \hat{\mu}(\lambda, Y) \quad = \quad \mathcal{X}\left(\mathcal{X}'\mathcal{X}\right)^{-1}\mathcal{X}'\Delta Y$$

Hence, the fit $\hat{\mu}(\lambda, Y)$ for a $Y \notin \mathcal{C}_\lambda$ behaves locally as a projection on a fixed subspace characterized by $\mathcal{R}_\lambda$ and $\mathcal{U}_\lambda$. As projections have eigenvalues bounded by 1, it follows that:

$$\|X\hat{\beta}(\lambda, y + \Delta y) - X\hat{\beta}(\lambda, y)\| \leq \|\Delta y\|, \qquad \text{for sufficiently small } \Delta y$$

Furthermore, we have that $\nabla \cdot \hat{\mu}(\lambda, Y) = \mathrm{tr}(\mathcal{X}\left(\mathcal{X}'\mathcal{X}\right)^{-1}\mathcal{X}') = |\mathcal{U}_\lambda| + 1$ from standard results for projection matrices.

□

As in Zou et al. (2004), the results from Lemma 3 yield that the fit $\hat{\mu}(\lambda, y)$ is uniformly Lipschitz in $\mathbb{R}^n$ as the closure of $\mathcal{C}_\lambda$ is the entire $\mathbb{R}^n$ space.

Theorem 4 *The L_∞ fit $\hat{\mu}_\lambda(y)$ is uniformly Lipschitz for all λ. The degrees of freedom of $\hat{\mu}_\lambda(y)$ equal the expectation of the number of elements in the unrestricted set plus 1, that is:*

$$df(\lambda) \quad = \quad E\left[|\mathcal{U}_\lambda|\right] + 1$$

Proof of Theorem 4: From propositon 3 above and subsequent comment, we have that $\hat{\mu}_\lambda(y)$ is uniformly Lipschitz in $\mathbb{R}^n$. The expression for the number of degrees of freedom then follows from Stein's lemma and the expression for the formula for the divergent shown above. □

The proof for the case of nonoverlapping groups follow the same steps. It is not presented in details as a detailed proof is lengthy and not very insightful. We present a summary of the steps involved.

The first part of the proof consists of establishing facts 1 through 3 for the nonoverlapping case. Similarly to what happens to the Lasso, fact 1 is proven by noticing that for each of the four possible events characterizing a breakpoint $Y \in \mathbb{R}^n$ belongs to a hyperplane. As before, y must belong to a union of these hyperplanes so that λ is a breakpoint. Fact 3 is established by noticing that the sets $\mathcal{A}_\lambda$ and $\mathcal{R}_{k,\lambda}, \forall k = 1, \ldots, K$ are invariant in between breakpoints. Fact 2 follows from the fact that the CAP objective function is continuous in both λ and Y and convex.

Once these three facts are established, all is needed is noticing that, except for the amount of shrinkage imposed to the coefficients which does not depend on Y, the CAP fit behaves as a projection onto a subspace whose dimension is given by the number of "free" parameters at that point of the path. The result then follows from arguments similar to the ones used for standard linear estimates.

B Appendix: Proofs

Proof Lemma 1: Since (3) can be factorized into

$$C^1_{\gamma_0,\gamma} \exp\{-\sum_{i=1}^{m_0}(\|\tilde{\beta}_{G_i}\|_{\gamma_i})^{\gamma_0}\} = \prod_{i=1}^{m_0} C^2_{\gamma_0,\gamma_i} \exp\{-\|\tilde{\beta}_{G_i}\|^{\gamma_0}_{\gamma_i}\} \tag{16}$$

for some normalization constant $C^2_{\gamma_0,\gamma_i}$ from integrating $\exp\{-\|\beta_{G_i}\|^{\gamma_0}_{\gamma_i}\}$. The independence between β_{G_i} for different i is obvious.

Turning to the independence between $\|\tilde{\beta}_{G_i}\|_{\gamma_i}$ and $\frac{\tilde{\beta}_{G_i}}{\|\tilde{\beta}_{G_i}\|_{\gamma_i}}$, without loss of generality we let $i = 1$ and assume $G_1 = 1, ..., m_1$. We need to be a little careful since $\frac{\tilde{\beta}_{G_1}}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}}$ is defined on the surface of the unit L_{γ_1} ball $\mathcal{B}_{\gamma_1} = \{x \in R^{m_1}; \|x\|_{\gamma_1} = 1\}$ rather than the whole R^{m_1}. Now consider substituding $\tilde{\beta}$ with the spherical coordinates (r, θ):

$$\begin{aligned}
\tilde{\beta}_1 &= r\cos\theta_1 \\
\tilde{\beta}_2 &= r\sin\theta_1\cos\theta_2 \\
&\vdots \\
\tilde{\beta}_{m_1} &= r\sin\theta_1\sin\theta_2 ... \sin\theta_{m_1-1}
\end{aligned}$$

where the Jacobian of the substitution is

$$J_{\tilde{\beta}}(r, \theta) = r^{m_1-1}\sin^{m_1-2}\theta_1 ... \sin\theta_{m_1-2}$$

Since only the separatability is needed for the proof, we simplify the notation by writting

$$\tilde{\beta}_j = r f_j(\theta) \ , j = 1, ..., m_1, \tag{17}$$

$$J_{\tilde{\beta}}(r, \theta) = r^{m_1-1} g(\theta) \tag{18}$$

and $\mathbf{f}(\theta) = (f_1(\theta), ..., f_{m_1}(\theta))$

By (17), we get

$$\|\tilde{\beta}_{G_1}\|_{\gamma_1} = r\|\mathbf{f}(\theta)\|_{\gamma_1} \tag{19}$$

$$\frac{\tilde{\beta}_j}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}} = \frac{f_j(\theta)}{\|\mathbf{f}(\theta)\|_{\gamma_1}} \tag{20}$$

Do another substitution of r by $s = \|\tilde{\beta}_{G_i}\|_{\gamma_i}$. By (19), the Jacobian is simply $\|\mathbf{f}(\theta)\|^{-1}_{\gamma_1}$ which leads to:

$$J_{\tilde{\beta}}(s, \theta) = s^{m_1-1}\|\mathbf{f}(\theta)\|^{-m_1}_{\gamma_1} g(\theta) \tag{21}$$

(20) also implies a bijection between $\frac{\tilde{\beta}_j}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}}$ and θ that does not involve r_0 or s.

Combining these results, now take (lebesgue) measurable sets $A \in [0, +\infty)$ and $B \in \mathcal{B}_{\gamma_1}$. The event $\frac{\tilde{\beta}_{G_1}}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}} \in B$ can also be defined by $\theta \in B'$ almost surely for approporiate B'. Therefore we have

$$P(\|\tilde{\beta}_{G_1}\|_{\gamma_1} \in A, \frac{\tilde{\beta}_{G_1}}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}} \in B)$$

$$
\begin{aligned}
&\propto \int_{\|\tilde{\beta}_{G_i}\|_{\gamma_1}\in A, \frac{\tilde{\beta}_j}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}}\in B} \exp\{-\|\tilde{\beta}_{G_i}\|_{\gamma_1}^{\gamma_0}\} d\tilde{\beta} \\
&= \int_{s\in A,\theta\in B'} \exp\{-s^{\gamma_0}\} s^{m_1-1} \|\mathbf{f}(\theta)\|_{\gamma_1}^{-m_1} g(\theta) ds d\theta \\
&= [\int_A s^{m_1-1} \exp\{-s^{\gamma_0}\} ds][\int_{B'} \|\mathbf{f}(\theta)\|_{\gamma_1}^{-m_1} g(\theta) d\theta] \qquad (22)
\end{aligned}
$$

(22) implies both the independence between $\|\tilde{\beta}_{G_1}\|_{\gamma_1}$ and $\frac{\tilde{\beta}_{G_1}}{\|\tilde{\beta}_{G_1}\|_{\gamma_1}}$ and the fact that their distributions do not depend on γ_1 and γ_0 respectively. □

Proof Theorem 1: Theorem 1 is a direct consequence of Lemma 1. Since the distribution of $\|\tilde{\beta}_{G_i}\|_{\gamma_i}$ does not depend on γ_i, setting $\gamma_i = \gamma_0$ does not change the distribution. This leads to a joint distribution

$$
\beta^* \propto \exp\{-\sum_{i=1}^{p} (\beta_i^*)^{\gamma_0}\}, \qquad (23)
$$

which immediately implies (4).

Similarly, for the distribution of $\frac{\tilde{\beta}_{G_i}}{\|\tilde{\beta}_{G_i}\|_{\gamma_i}}$, set $\gamma_0 = \gamma_i$ then (5) follows immediately. □

Proof Lemma 2:

1. Let α be a scalar. For each group, we know from the properties of a norm that:

$$
N_k(\alpha\beta) \quad = \quad \alpha N_k(\beta)
$$

Hence $N(\alpha\beta) = \alpha N(\beta)$ and from the property of norms again, we have:

$$
P(\alpha\beta) = \|N(\alpha\beta)\|_{\gamma_0} \quad = \quad |\alpha| \|N(\beta)\|_{\gamma_0} = |\alpha| P(\beta)
$$

2. Let β_1 and β_2 be two distinct values for the parameter β. From the triangular inequality applied to each group-norm, we have that:

$$
N_k(\beta_1 + \beta_2) \quad \leq \quad N_k(\beta_1) + N_k(\beta_2), \qquad \forall k = 1, \ldots, K
$$

It follows that:

$$
\sum_k (N_k(\beta_1 + \beta_2))^{\gamma_0} \quad \leq \quad \sum_k (N_k(\beta_1) + N_k(\beta_2))^{\gamma_0}
$$

and hence:

$$
T(\beta_1 + \beta_2) \leq \|N(\beta_1) + N(\beta_2)\|_{\gamma_0} \leq \|N(\beta_1)\|_{\gamma_0} + \|N(\beta_2)\|_{\gamma_0} = T(\beta_1) + T(\beta_2)
$$

where the second inequality follows from the triangular inequality for general norms.

□

Proof Theorem 3: Let $\theta \in [0,1]$ and compute $\beta^* = \theta\beta_1 + (1-\theta)\beta_2$. It follows that:

$$\begin{aligned} P(\beta^*) &= P\left(\theta\beta_1 + (1-\theta)\beta_2\right) \\ &\leq P\left(\theta\beta_1\right) + P\left((1-\theta)\beta_2\right) \\ &= \theta P\left(\beta_1\right) + (1-\theta)P\left(\beta_2\right) \end{aligned}$$

and hence T is convex. Since L and T are convex and $\lambda \geq 0$, the result follows from the fact that the sum of convex functions is itself convex. □

References

H. Akaike. Information theory and an extension of the maximum likelihood principle. *Proc. 2nd International Symposium on nformation Theory*, pages 267–281, 1973.

P. Bartlett, M. Jordan, and J. McAuliffe. Convexity, classification and risk bounds. *JASA*, 101:138–156, 2006. URL `http://www.cs.berkeley.edu/ jordan/papers/bartlett-jordan-mcauliffe.ps`.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Peter Bühlmann and Bin Yu. Boosting with the l2 loss: Regression and classification. *J. Amer. Statist. Assoc.*, 98:324–340, 2003.

S. Chen and D. Donoho. Basis pursuit. Technical report, Department of Statistics, Stanford University, 1994. URL `http://stat.stanford.edu/ donoho/Reports/1994/asilomar.ps.Z`.

S.S. Chen, D. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 2001.

S. Dudoit, M. J. van der Laan, S. Keles, A. M. Molinaro, S. E. Sinisi, , and S. L. Teng. Loss-based estimation with cross-validation: Applications to microarray data analysis. *SIGKDD Explorations*, 5(2):56–68, 2003.

B. Efron and R. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1994.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 35:407–499, 2004.

I. E. Frank and J. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35: 109–148, 1993.

Y. Freund and R. Schapire. A decision theoretic generalization of online learning and an application to boosting. *Journal Computer and System Sciences*, 1997.

M. Hansen and B. Yu. Model selection and the principle of mdl. *J. Amer. Statist. Assoc.*, 96:746–774, 2001. URL `http://www.stat.berkeley.edu/users/binyu/ps/mdl.ps`.

A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation of nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

Y. Kim, J. Kim, and Y. Kim. Blockwise sparse regression. *Statistica Sinica*, 16(2), 2006.

C. Leng, Y. Lin, and G. Wahba. A note on the lasso and related procedures in model selection. Technical report, Department of Statistics, University of Wisconsin, 2004. URL `http://www.stat.wisc.edu/ wahba/ftp1/`.

L. Meier, S. van der Geer, and Peter Bühlmann. The group lasso for logistic regression. 2006.

M. Osborne, B. Presnell, and B. Turlach. A new approach to variable selection in least square problems. *IMA Journal of Numeric Analysis*, 20:389–404, 2000.

Mee-Young Park and Trevor Hastie. An l1 regularization-path algorithm for generalized linear models. Technical report, Department of Statistics, Stanford University, 2006. URL `http://www-stat.stanford.edu/ hastie/Papers/glmpath.pdf`.

S. Rosset. Tracking curved regularized optimization solution paths. In *NIPS*, 2004. URL `http://www-stat.stanford.edu/ saharon/papers/path.pdf`.

S. Rosset and J. Zhu. Piecewise linear regularized solution paths. Technical report, Department of Statistics, University of Michigan, 2006. URL `http://www-stat.stanford.edu/ saharon/papers/piecewise-revised.pdf`.

Gideon Schwartz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.

Tao Shi, Bin Yu, Eugene E. Clothiaux, and Amy J. Braverman. Cloud detection over snow and ice using misr data. Technical report, Department of Statistics, UC Berkeley, 2004. URL `http://www.stat.berkeley.edu/users/binyu/ps/cloud.pdf`.

N. Sugiura. Further analysis of the data by akaike's information criterion and finite corrections. *Communications in Statistics*, A7:13–26, 1978.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

Yuhong Yang. Can the strengths of aic and bic be shared? *Biometrika*, 101:937–950, 2003.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68(1):49–67, 2006. URL `http://www.stat.wisc.edu/public/ftp/yilin/tr1095.pdf`.

T. Zhang and B. Yu. Boosting with early stopping consistency and convergence. *Annals of Statistics*, 33: 1538–1579, 2005.

P. Zhao and B. Yu. Boosted lasso. Technical report, Department of Statistics, UC Berkeley, 2004. URL `http://www.stat.berkeley.edu/users/binyu/ps/blasso.ps`.

Peng Zhao and Bin Yu. On model selection consistency of lasso. Technical report, UC Berkeley, Department of Statistics, 2006. URL `http://www.stat.berkeley.edu/users/binyu/ps/LassoConsi.pdf`.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67(2):301–320, 2005.

Hui Zou, Trevor Hastie, and Rob Tibshirani. On the "degrees of freedom" of the lasso. Technical report, Department of Statistics, Stanford University, 2004. URL `http://www-stat.stanford.edu/ hastie/Papers/dflasso.pdf`.

			Oracle				CV			
	OLS	Null model	1	2	4	∞	1	2	4	∞
Model	3.83	2.09	0.596	0.474	0.437	0.441	0.765	0.628	0.608	0.614
Error	(1.592)	(0.088)	(0.193)	(0.160)	(0.157)	(0.182)	(0.396)	(0.343)	(0.352)	(0.390)
# coefs	30.00	0.00	8.680	10.440	9.520	14.080	10.080	11.520	12.920	15.920
	(0.000)	(0.000)	(2.577)	(3.709)	(3.938)	(4.949)	(7.088)	(7.827)	(7.719)	(8.031)

Table 3: Results for the first grouping experiment: oracle and CV

			Oracle		CV		BIC		AIC_C	
	OLS	Null model	1	∞	1	∞	1	∞	1	∞
Model	3.83	2.09	0.596	0.441	0.765	0.614	1.125	0.718	0.764	0.590
Error	(1.592)	(0.088)	(0.193)	(0.182)	(0.396)	(0.390)	(0.574)	(0.232)	(0.428)	(0.330)
# coefs	30.00	0.00	8.680	14.080	10.080	15.920	1.360	2.240	7.520	8.160
	(0.000)	(0.000)	(2.577)	(4.949)	(7.088)	(8.031)	(0.907)	(0.879)	(4.389)	(5.684)
# $\hat{\text{dfs}}$	30.00	0.00	8.680	8.360	10.080	10.720	1.360	1.120	7.520	4.560
	(0.000)	(0.000)	(2.577)	(3.264)	(7.088)	(7.329)	(0.907)	(0.440)	(4.389)	(3.630)

Table 4: Results for the first grouping experiment: using the estimates of the number of degrees of freedom for model selection

				Oracle				CV			
		OLS	Null model	1	2	4	∞	1	2	4	∞
Case 1	Model	1101.3	697.0	55.636	25.503	20.008	17.183	60.767	29.616	23.966	20.393
	Error	(591.99)	(29.98)	(30.352)	(15.080)	(11.776)	(10.402)	(32.353)	(16.779)	(14.191)	(11.643)
	# coefs	40.00	0.00	11.720	17.920	17.080	16.400	11.480	17.920	17.360	16.800
		(0.000)	(0.000)	(2.923)	(1.579)	(1.382)	(1.258)	(3.405)	(2.871)	(2.396)	(2.309)
Case 2	Model	1101.3	697.0	55.636	28.993	25.204	23.467	60.767	32.779	28.363	26.985
	Errors	(591.99)	(29.98)	(30.352)	(16.122)	(13.100)	(12.557)	(32.353)	(16.361)	(14.232)	(15.451)
	# coefs	40.00	0.00	11.720	19.120	18.440	18.080	11.480	19.320	18.840	18.760
		(0.000)	(0.000)	(2.923)	(1.301)	(1.227)	(0.997)	(3.405)	(2.249)	(2.230)	(2.587)
Case 3	Model	20.10	678.40	6.230	2.706	2.310	2.141	9.419	6.458	6.955	5.654
	Error	(4.88)	(3.11)	(2.538)	(1.863)	(1.572)	(1.555)	(4.772)	(4.519)	(5.365)	(4.421)
	# coefs	40.00	0.00	15.200	17.640	16.840	16.360	17.320	21.960	21.880	20.880
		(0.000)	(0.000)	(3.055)	(1.846)	(1.344)	(1.381)	(7.347)	(7.613)	(7.496)	(7.305)
Case 4	Model	20.10	678.40	6.230	3.342	3.155	3.046	9.419	6.770	7.073	6.477
	Error	(4.88)	(3.11)	(2.538)	(1.941)	(1.750)	(1.668)	(4.772)	(4.609)	(4.391)	(4.440)
	# coefs	40.00	0.00	15.200	19.120	18.520	18.360	17.320	23.160	23.120	22.600
		(0.000)	(0.000)	(3.055)	(1.943)	(1.503)	(1.469)	(7.347)	(6.421)	(6.833)	(6.795)

Table 5: Results for the second grouping experiment: oracle and CV
Case 1: Small Sample with Correct Grouping. Case 2: Small Sample with Incorrect Grouping. Case 3: Large Sample with Correct Grouping. Case 4: Large Sample with Incorrect Grouping.

				Oracle		CV		BIC		AIC_C	
		OLS	Null model	1	∞	1	∞	1	∞	1	∞
Case 1	Model Error	1101.3	697.0	55.636	17.183	60.767	20.393	144.733	20.273	71.078	26.880
		(591.99)	(29.98)	(30.352)	(10.402)	(32.353)	(11.643)	(172.170)	(13.110)	(35.240)	(17.661)
	# coefs	40.00	0.00	11.720	16.400	11.480	16.800	4.200	15.040	8.880	16.760
		(0.000)	(0.000)	(2.923)	(1.258)	(3.405)	(2.309)	(1.683)	(0.200)	(3.800)	(2.260)
	# $\hat{df}$s	40.00	0.00	11.720	4.400	11.480	4.800	4.200	3.040	8.880	4.760
		(0.000)	(0.000)	(2.923)	(1.258)	(3.405)	(2.309)	(1.683)	(0.200)	(3.800)	(2.260)
Case 2	Model Error	1101.3	697.0	55.636	23.467	60.767	26.985	144.733	32.494	71.078	38.126
		(591.99)	(29.98)	(30.352)	(12.557)	(32.353)	(15.451)	(172.170)	(27.601)	(35.240)	(21.576)
	# coefs	40.00	0.00	11.720	18.080	11.480	18.760	4.200	17.040	8.880	19.160
		(0.000)	(0.000)	(2.923)	(0.997)	(3.405)	(2.587)	(1.683)	(0.200)	(3.800)	(2.882)
	# $\hat{df}$s	40.00	0.00	11.720	5.680	11.480	6.360	4.200	4.320	8.880	6.680
		(0.000)	(0.000)	(2.923)	(1.145)	(3.405)	(2.481)	(1.683)	(0.900)	(3.800)	(3.092)
Case 3	Model Error	20.100	678.399	6.230	2.141	9.419	5.654	9.791	2.431	7.150	3.201
		(4.880)	(3.109)	(2.538)	(1.555)	(4.772)	(4.421)	(4.711)	(1.635)	(3.286)	(3.119)
	# coefs	40.000	0.000	15.200	16.360	17.320	20.880	8.520	15.000	14.560	16.960
		(0.000)	(0.000)	(3.055)	(1.381)	(7.347)	(7.305)	(1.475)	(0.000)	(4.243)	(3.600)
	# $\hat{df}$'s	40.000	0.000	15.200	4.360	17.320	9.760	8.520	3.000	14.560	5.000
		(0.000)	(0.000)	(3.055)	(1.381)	(7.347)	(8.791)	(1.475)	(0.000)	(4.243)	(3.686)
Case 4	Model Error	20.100	678.399	6.230	3.046	9.419	6.477	9.791	3.356	7.150	4.523
		(4.880)	(3.109)	(2.538)	(1.668)	(4.772)	(4.440)	(4.711)	(1.770)	(3.286)	(3.653)
	# coefs	40.000	0.000	15.200	18.360	17.320	22.600	8.520	17.000	14.560	19.720
		(0.000)	(0.000)	(3.055)	(1.469)	(7.347)	(6.795)	(1.475)	(0.000)	(4.243)	(3.835)
	# $\hat{df}$'s	40.000	0.000	15.200	6.360	17.320	11.520	8.520	5.000	14.560	7.800
		(0.000)	(0.000)	(3.055)	(1.469)	(7.347)	(8.317)	(1.475)	(0.000)	(4.243)	(3.990)

Table 6: Results for the second grouping experiment: using the estimates of the number of degrees of freedom Case 1: Small Sample with Correct Grouping. Case 2: Small Sample with Incorrect Grouping. Case 3: Large Sample with Correct Grouping. Case 4: Large Sample with Incorrect Grouping.

				Oracle				CV			
		OLS	Null model	1	2	4	∞	1	2	4	∞
Case 1	Model Error	50.9	1482.0	31.775	19.610	14.221	10.613	35.764	23.788	19.545	17.128
		(11.20)	(5.03)	(5.706)	(4.545)	(4.285)	(4.134)	(7.508)	(7.503)	(8.663)	(9.193)
	# coefs	50.00	0.00	35.160	38.320	38.160	45.600	37.200	40.120	37.960	44.800
		(0.00)	(0.00)	(2.968)	(5.105)	(4.384)	(5.831)	(7.528)	(6.604)	(6.024)	(7.141)
Case 2	Model Errors	50.9	1482.0	31.775	20.426	15.993	12.234	35.764	24.979	20.760	19.214
		(11.20)	(5.03)	(5.706)	(4.648)	(4.561)	(4.460)	(7.508)	(7.214)	(8.566)	(9.288)
	# coefs	50.00	0.00	35.160	37.600	38.080	43.520	37.200	41.160	38.680	44.240
		(0.00)	(0.00)	(2.968)	(3.926)	(4.406)	(6.104)	(7.528)	(6.176)	(6.606)	(8.166)

Table 7: Results for the third grouping experiment: oracle and CV Case 1: Correct Grouping. Case 2: Incorrect Grouping.

				Oracle		CV		BIC		AIC_C	
		OLS	Null model	1	∞	1	∞	1	∞	1	∞
Case 1	Model Error	50.9	1482.0	31.775	10.613	35.764	17.128	74.184	21.621	34.107	13.606
		(11.20)	(5.03)	(5.706)	(4.134)	(7.508)	(9.193)	(32.452)	(10.337)	(6.100)	(4.935)
	# coefs	50.00	0.00	35.160	45.600	37.200	44.800	21.080	30.000	32.000	44.400
		(0.000)	(0.000)	(2.968)	(5.831)	(7.528)	(7.141)	(3.390)	(0.000)	(4.416)	(7.681)
	# $\hat{\text{dfs}}$	50.00	0.00	35.160	13.200	37.200	20.280	21.080	3.880	32.000	13.400
		(0.000)	(0.000)	(2.968)	(2.972)	(7.528)	(13.337)	(3.390)	(0.927)	(4.416)	(6.090)
Case 2	Model Error	50.9	1482.0	31.775	12.234	35.764	19.214	74.184	21.595	34.107	15.391
		(11.20)	(5.03)	(5.706)	(4.460)	(7.508)	(9.288)	(32.452)	(9.123)	(6.100)	(5.610)
	# coefs	50.00	0.00	35.160	43.520	37.200	44.240	21.080	32.000	32.000	42.440
		(0.000)	(0.000)	(2.968)	(6.104)	(7.528)	(8.166)	(3.390)	(0.000)	(4.416)	(7.200)
	# $\hat{\text{dfs}}$	50.00	0.00	35.160	14.560	37.200	23.920	21.080	5.840	32.000	13.800
		(0.000)	(0.000)	(2.968)	(2.959)	(7.528)	(13.269)	(3.390)	(1.248)	(4.416)	(6.238)

Table 8: Results for the third grouping experiment: using the estimates of the number of degrees of freedom Case 1: Correct Grouping. Case 2: Incorrect Grouping.

			Oracle				CV			
	OLS	Null model	1	2	4	∞	1	2	4	∞
Model Error	66.88	299.68	42.350	30.035	31.886	34.227	50.997	35.065	37.700	40.649
	(27.08)	(6.81)	(25.008)	(16.792)	(17.575)	(18.826)	(30.426)	(17.521)	(19.839)	(22.219)
$ coefs.	15.00	0.00	9.120	9.800	9.720	9.720	8.960	9.760	9.240	9.080
	(0.00)	(0.00)	(2.147)	(2.141)	(2.112)	(2.390)	(3.541)	(2.712)	(2.833)	(3.161)

Table 9: Results for the wavelet tree example

				Oracle				CV			
		OLS	Null model	1	2	4	∞	1	2	4	∞
Case 1	Model Error	18.32	55.25	1.600	1.001	1.142	1.171	2.104	1.251	1.317	1.447
		(5.65)	(0.30)	(0.640)	(0.471)	(0.474)	(0.494)	(0.964)	(0.639)	(0.623)	(0.827)
	# coefs	55.00	0.00	9.920	15.520	16.600	17.440	14.600	17.120	16.640	19.400
		(0.000)	(0.000)	(1.801)	(5.554)	(6.083)	(5.938)	(6.014)	(7.656)	(6.993)	(8.495)
Case 2	Model Error	18.59	56.08	2.092	1.355	1.496	1.533	2.741	1.765	1.952	1.860
		(5.73)	(0.29)	(0.730)	(0.517)	(0.547)	(0.537)	(1.561)	(1.320)	(1.468)	(1.231)
	# coefs	55.00	0.00	11.960	17.960	17.720	17.840	17.040	20.440	19.840	20.000
		(0.000)	(0.000)	(2.574)	(5.272)	(5.601)	(6.459)	(8.023)	(9.074)	(8.459)	(8.578)
Case 3	Model Error	19.73	59.54	3.112	2.157	2.193	2.126	3.943	2.627	2.664	2.544
		(6.08)	(0.29)	(1.201)	(0.868)	(0.895)	(0.874)	(2.004)	(1.576)	(1.699)	(1.541)
	# coefs	55.00	0.00	14.840	20.600	20.680	22.200	20.440	24.040	23.360	23.320
		(0.000)	(0.000)	(2.688)	(6.252)	(5.528)	(5.723)	(8.216)	(7.231)	(8.020)	(8.285)
Case 4	Model Error	63.27	190.83	11.110	12.097	12.547	11.916	13.063	13.606	13.983	13.839
		(19.50)	(0.74)	(5.594)	(5.622)	(5.648)	(5.513)	(6.903)	(7.025)	(7.125)	(7.902)
	# coefs	55.00	0.00	15.960	26.440	26.360	26.360	19.840	27.400	27.000	28.400
		(0.000)	(0.000)	(3.221)	(5.687)	(6.632)	(6.231)	(6.780)	(8.578)	(8.185)	(9.046)
Case 5	Model Error	174.17	524.52	37.354	40.799	43.658	39.574	39.304	45.770	48.576	42.454
		(53.68)	(1.67)	(20.169)	(21.249)	(22.138)	(20.562)	(20.277)	(24.460)	(25.665)	(21.179)
	# coefs	55.00	0.00	19.080	31.200	28.720	28.360	20.320	28.720	26.240	27.280
		(0.000)	(0.000)	(3.651)	(4.655)	(5.489)	(5.122)	(5.194)	(8.101)	(9.623)	(7.662)

Table 10: Results for the ANOVA hierarchical example