Penalized and Constrained Optimization: An Application to High-Dimensional Website Advertising

Gareth M. James, Courtney Paulson, and Paat Rusmevichientong *

*Gareth M. James is the E. Morgan Stanley Chair in Business Administration and a Professor of Data Sciences and Operations, University of Southern California Marshall School of Business, Los Angeles, CA 90089 (e-mail: gareth@marshall.usc.edu); Courtney Paulson is an Assistant Professor of Decision, Operations, and Information Technology, University of Maryland Smith School of Business, College Park, MD 20742 (e-mail: cpaulson@rhsmith.umd.edu); and Paat Rusmevichientong is a Professor of Data Sciences and Operations, University of Southern California Marshall School of Business, Los Angeles, CA 90089 (e-mail: rusmevic@marshall.usc.edu).

Abstract

Firms are increasingly transitioning advertising budgets to Internet display campaigns, but this transition poses new challenges. These campaigns use numerous potential metrics for success (e.g., *reach* or *clickthrough rate*), and because each website represents a separate advertising opportunity, this is also an inherently high-dimensional problem. Further, advertisers often have constraints they wish to place on their campaign, such as targeting specific sub-populations or websites. These challenges require a method flexible enough to accommodate thousands of websites, as well as numerous metrics and campaign constraints. Motivated by this application, we consider the general constrained high-dimensional problem, where the parameters satisfy linear constraints. We develop the *Penalized and Constrained* optimization method (PAC) to compute the solution path for high-dimensional, linearly-constrained criteria. PAC is extremely general; in addition to internet advertising, we show it encompasses many other potential applications, like portfolio estimation, monotone curve estimation, and the generalized lasso. Computing the PAC coefficient path poses technical challenges, but we develop an efficient algorithm over a grid of tuning parameters. Through extensive simulations, we show PAC performs well. Finally, we apply PAC to a proprietary dataset in an exemplar Internet advertising case study and demonstrate its superiority over existing methods in this practical setting.

Keywords: PAC; Constrained Problems; Internet Advertising; High-Dimensional Optimization; Constrained Lasso

1. Introduction

This paper is inspired by problems in the online advertising industry. In 2012, U.S. digital advertising spending totaled 37 billion dollars, of which Internet display advertising accounted for 40%. This percentage is expected to continue to grow, outpacing paid search ad spending (eMarketer, 2012) which until recently has been the forefront of Internet advertising marketing research[1]. Nevertheless, with millions of potential websites (HBR, 2015), firms face considerable challenges in deciding where to place their online display ads. As

[1]Internet display advertising refers to static advertisements shown on specific websites, while search advertising refers to ads shown on the search result pages. This paper focuses on Internet display advertising.

the number of ad buying opportunities proliferates, advertising automation has become inevitable. Unfortunately, advertising on websites entails a variety of challenges not present in traditional media such as newspapers or magazines, particularly in the wide variations in cost and traffic across websites as well as the sheer number of websites. One reason for the difficulty of optimizing budget allocations over p websites is that simply considering all possible combinations of websites involves 2^p possible subsets, a computationally infeasible NP-hard problem.

An additional complication arises because firms often wish to optimize a given marketing metric subject to a set of constraints on the allocation of the advertising budget. For example, imagine a firm is developing an advertising campaign to promote a new NCAA sports mobile app. The firm knows its target audience visits sports update websites, e.g. ESPN or Yahoo Sports. Because of this the firm might wish to allocate, for example, 50% of its advertising budget to these sports websites, since the firm knows it will reach more target consumers at these sites. The campaign may also be designed to constrain the advertising to target consumers from certain demographics such as age group, income level, geographic region, family status, etc.

Most state-of-the-art marketing methods can only optimize these metrics on the order of 10 websites, and we are not aware of any that can incorporate a set of linear constraints. However, the following optimization problem is more computationally tractable:

$$\arg\min_{\boldsymbol{\beta}} g(\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1, \tag{1}$$

where g a well-behaved convex loss function and $\boldsymbol{\beta} \in \mathbb{R}^p$ is a vector of coefficients over which we wish to optimize. Recently Paulson et al. (2018) demonstrated that, by choosing $g(\boldsymbol{\beta})$ to represent various marketing metrics, (1) could be used to efficiently optimize Internet campaigns involving thousands of websites. Optimization problems of this form have also been extensively studied in the statistical literature. For example, if $g(\boldsymbol{\beta}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$, then (1) reduces to the lasso (Tibshirani, 1996). Alternatively, when $g(\boldsymbol{\beta})$ represents the log likelihood function for a generalized linear model (GLM), then (1) implements a GLM extension of the lasso. However, as Paulson et al. (2018) shows, this generalized form is extremely versatile and can be applied in a variety of settings beyond fitting standard statistical models. Numerous algorithms have been developed for solving (1) for various instances of $g(\boldsymbol{\beta})$, including LARS (Efron et al., 2004), the alternating direction method of multipliers (ADMM)

algorithm (Boyd et al., 2010), DASSO (James et al., 2009a), bundled regularization gradient methods (Teo et al., 2010), and coordinate descent methods such as block descent (de Leeuw, 1994; Xu and Yin, 2013) or coordinate-wise steepest descent (Lange, 2012). See also Lange et al. (2014) who provide a very comprehensive overview of penalized regularization methods for general linear models and their algorithms, including using coordinate descent (Friedman et al., 2007; Wu and Lange, 2008).

While Paulson et al. (2018) provides an important step forward in handling high-dimensional marketing problems, it fails to address the important practical issue of imposing constraints on website allocations. Hence, in this article we consider an extension of (1) to the constrained setting:

$$\arg\min_{\boldsymbol{\beta}} g(\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1 \quad \text{subject to} \quad \mathbf{C}\boldsymbol{\beta} \leq \mathbf{b}, \tag{2}$$

where $\mathbf{C} \in \mathbb{R}^{m\times p}$ is a predefined constraint matrix and $\mathbf{b} \in \mathbb{R}^m$ is the corresponding predefined constraint vector. We refer to this problem as *Penalized And Constrained* (PAC) optimization. The PAC problem appears similar to the more standard setting in (1). However, the addition of m linear constraints turns out to significantly increase both the range of scenarios in which (2) is relevant and also the difficulty of optimizing the criterion. There has been some previous work on optimizing constrained criteria of this form, mostly in the setting where g is a quadratic sum of squares term, in which case (2) produces a constrained lasso fit. The constrained lasso can be fit using standard quadratic programming (Frank and Wolfe, 1956; Floudas and Visweswaran, 1995). However, just as with the standard lasso, this approach is inefficient when the solution needs to be computed over a wide range of possible values for λ. The LARS path algorithm provided an efficient optimization approach for the lasso, but it can not be used in the constrained setting. In a recent paper, Gaines et al. (2018) develop an analogous path algorithm to LARS but using linear constraints. The same paper also proposes an ADMM approach to this problem, while He (2011) develops a somewhat different path algorithm. However, this work all assumes a quadratic loss function and can not be easily extended to the more general setting, such as where g represents a marketing metric as in Paulson et al. (2018).

Main Contributions: This paper makes three important contributions. First we illustrate a few of the wide range of applications where optimization problems of the form given by (2) are applicable. In particular, we show that PAC has applications in fitting smooth

monotone functions and portfolio optimizaiton. We also demonstrate that the generalized lasso problem (Tibshirani and Taylor, 2011) is a special case of our PAC formulation.

Second we present an efficient algorithm for solving PAC that generates a sequence of solutions over λ. By exploiting the structure of the linear constraints, our method solves a standard lasso problem with appropriately-defined vector $\boldsymbol{Y}$ and matrix $\mathbf{X}$. Because our algorithm can be applied in conjunction with standard lasso optimization methods, our approach is both simple to implement and much faster than standard algorithms such as quadratic programming.

Third, we apply our method to solve an important marketing problem. Namely, how to optimize different metrics across thousands of Internet websites subject to various budget constraints. We provide an extensive case study of this problem using a unique and proprietary comScore Media Metrix dataset (described in Section 3.2) anonymously recording daily webpage usage information from a panel of 100,000 Internet users. Our case study illustrates how PAC can be used to efficiently optimize either the *reach*, the fraction of customers who are exposed to a given ad at least one time during a specified campaign, or *clickthrough rate*, the fraction of customers who click on a given ad, subject to a set of real-world constraints. Most importantly, we show how the linear constraints in the PAC formulation can be used to target consumers with certain demographic profiles, such as age group, income level, geographic region, family status, etc. Our analysis indicates that our method can improve the clickthrough rate of a target segment by over 100% when compared to existing approaches in the literature. To our knowledge, this is the first non-propriety method for solving such problems in a real-world setting.

The rest of this article is structured as follows. In Section 2, we illustrate some applications where (2) is applicable. Section 3 demonstrates that the constrained marketing problem can be formulated as a PAC optimization. We present our algorithm for solving PAC in Section 4. Several simulation studies are provided in Section 5 to demonstrate the validity of the proposed methodology. In Section 6 we discuss our case study using the comScore Media Metrix data. Finally, Section 7 provides a brief conclusion.

2. Applications of PAC

Several well-known real world applications can be formulated as PAC optimization problems. We briefly discuss a few examples here.

Example 1 (Monotone Curve Fitting)**.** Consider the problem of fitting a smooth function, $h(x)$, to a set of observations $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, subject to the constraint that h must be monotone. Model $h(x)$ as $\mathbf{B}(x_i)^T\boldsymbol{\beta}$, where $\mathbf{B}(x_i)$ is a high-dimensional flexible basis function such as a spline basis. Then this problem can be addressed using the PAC methodology by minimizing $g(\boldsymbol{\beta}) = \sum_{i=1}^{n}(y_i - \mathbf{B}(x_i)^T\boldsymbol{\beta})^2$ subject to $\mathbf{C}\boldsymbol{\beta} \leq \mathbf{0}$, where the l^{th} row of $\mathbf{C}$ is the derivative $\mathbf{B}'(u_l)$ of the basis functions evaluated at u_ℓ for a fine grid of points, $u_1, \ldots, u_m$, over the range of x. Enforcing this constraint ensures that the derivative of h is non-positive, so h will be monotone decreasing.

Example 2 (Portolio Optimization)**.** Portfolio optimization is another well-known problem of interest which turns out to fit the PAC setting. Suppose we have p random assets indexed by $1, 2, \ldots, p$ whose covariance matrix is denoted by $\boldsymbol{\Sigma}$. Markowitz (1952, 1959) developed the seminal framework for mean-variance analysis. In particular his approach involved choosing asset weights $\boldsymbol{\beta}$ to minimize the portfolio risk $R(\boldsymbol{\beta}) = \boldsymbol{\beta}^T\boldsymbol{\Sigma}\boldsymbol{\beta}$ subject to $\boldsymbol{\beta}^T\mathbf{1} = 1$. One often also wishes to impose additional constraints on $\boldsymbol{\beta}$ to control the expected return of the portfolio, the allocations among sectors or industries, or the exposures to certain known risk factors.

In practice $\boldsymbol{\Sigma}$ is unobserved so must be estimated using the sample covariance matrix, $\hat{\boldsymbol{\Sigma}}$. However, it has been well documented in the finance literature that when p is large, which is the norm in real-world applications, minimizing $\hat{R}(\boldsymbol{\beta}) = \boldsymbol{\beta}^T\hat{\boldsymbol{\Sigma}}\boldsymbol{\beta}$ gives poor estimates for $\boldsymbol{\beta}$. One possible solution involves regularizing $\hat{\boldsymbol{\Sigma}}$, but more recently attention has focused on directly penalizing or constraining the weights, an analogous approach to penalizing the coefficients in a regression setting. Fan et al. (2012) adopted this framework by minimizing $\hat{R}(\boldsymbol{\beta})$ subject to $\boldsymbol{\beta}^T\mathbf{1} = 1$ and $\|\boldsymbol{\beta}\|_1 \leq c$, where c is a tuning parameter. It is not hard to verify that this optimization problem can be expressed in the form of (2), where $\mathbf{C}$ has at least one row (to constrain $\boldsymbol{\beta}$ to sum to one) but may also have additional rows if we place constraints on the expected return, industry weightings, etc. Hence, implementing PAC with $g(\boldsymbol{\beta}) = \hat{R}(\boldsymbol{\beta})$ allows us to solve the constrained and regularized portfolio optimization problem.

Example 3 (Generalized Lasso)**.** Another application of PAC involves the generalized lasso problem (Tibshirani and Taylor, 2011):

$$\arg\min_{\boldsymbol{\theta}} \frac{1}{2} \left\| \boldsymbol{Y} - \tilde{\mathbf{X}}\boldsymbol{\theta} \right\|_2^2 + \lambda \left\| \boldsymbol{D}\boldsymbol{\theta} \right\|_1, \tag{3}$$

where $\boldsymbol{D} \in \mathbb{R}^{r \times p}$. When $\mathsf{rank}(\boldsymbol{D}) = r$, and thus $r \leq p$, Tibshirani and Taylor (2011) show that the generalized lasso can be converted to the classical lasso problem. However, if $r > p$ then such a reformulation is not possible. Lemma 1 shows that when $r > p$ and $\boldsymbol{D}$ is full column rank, then there is an interesting connection between the generalized lasso and PAC.

Lemma 1 (Generalized Lasso is a Special Case of PAC)**.** *If $r > p$ and $\mathsf{rank}(\boldsymbol{D}) = p$ then there exist matrices $\mathbf{A}, \mathbf{C}$ and $\mathbf{X}$ such that, for all values of λ, the solution to (3) is equal to $\boldsymbol{\theta} = \mathbf{A}\boldsymbol{\beta}$, where $\boldsymbol{\beta}$ is given by:*

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2} \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \quad \textit{subject to} \quad \mathbf{C}\boldsymbol{\beta} = \mathbf{0}.$$

The proof of Lemma 1 is provided in Appendix A. Hence, any problem that falls into the generalized lasso paradigm can be solved as a PAC problem with $g(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$ and $\mathbf{b} = \mathbf{0}$. Tibshirani and Taylor (2011) further demonstrate that a variety of common statistical methods can be formulated as special cases of the generalized lasso. Some examples include: the fused lasso (Tibshirani et al., 2005), polynomial trend filtering (where one penalizes discrete differences to produce smooth piecewise polynomial curves), wavelet smoothing, and the FLiRTI method (James et al., 2009b). She (2010) also considers a similar criterion to (3) and discusses special cases such as the "clustered lasso". Lemma 1 shows that all of these various approaches can be solved using PAC.

3. Internet Media Campaigns and the comScore Data

In Section 3.1 we demonstrate that our target application, the constrained large-scale Internet media selection problem, can be formulated as a PAC optimization problem. We also include an overview of the data used for the case study in Section 3.2.

3.1 Internet Media Campaigns

Although PAC can be applied in many settings, our main focus in this paper is the application to Internet media campaigns. These campaigns have traditionally focused on two common

Internet advertising goals: maximizing ad reach or maximizing clickthrough rate (CTR) across a given advertising campaign, subject to some maximum allowed budget B. As discussed in Section 1, *reach* is defined as the fraction of customers who are exposed to a given ad at least one time, while *clickthrough rate* is defined as the fraction of customers who click on an ad. However, modeling either of these first requires defining the functions themselves, as there is no standard reach or CTR function for Internet media.

CTR is a natural extension of reach, since before an Internet user can click on an ad, he or she must first be exposed to it. Thus the two functions are naturally related and in fact can be formulated together. We assume that we have a collection of p websites indexed by $j = 1, \ldots, p$ where we can potentially show our ads. Let $\gamma_j = \frac{1}{\tau_j \text{CPM}_j}$, where CPM_j is the cost per thousand impressions at website j, and τ_j is the expected total number of pages viewed (in thousands) at the jth website during the course of the ad campaign. Then γ_j corresponds to the fraction of all ads purchased at website j for every dollar spent by the campaign. Thus, if β_j is the dollar budget allocated to website j, then $\gamma_j \beta_j$ represents the probability of an ad appearing to a user on a visit to website j.

Though this is the definition of an ad reaching a user, we can take this one step further to incorporate clickthrough rate. Let q_j represent the conditional probability that users click on an ad given that the ad has appeared to them at website j. In practice q_j can be obtained either directly from past click logs (e.g. Dave and Varma, 2010) or estimated in numerous ways if historical data is not available (e.g. Immorlica et al., 2005). Once the q_j value for each webpage is known, we can incorporate these website-specific values into the computation of the CTR values as follows. If the probability an ad appears to the user is $\gamma_j \beta_j$, and users additionally have a probability q_j of clicking on an ad at website j given the ad has appeared to them, the unconditional probability of a user viewing the ad and clicking through it is $\gamma_j \beta_j q_j$.

To develop our complete optimization function, consider that because ad appearances are independent of users and previous visits, each time a user views website j, he or she has the same probability of failing to click through the ad (i.e. $1 - \gamma_j \beta_j q_j$). Hence, if user i views each website a total of z_{ij} times, then the probability he or she fails to click on the ad at least once over all p websites is $\prod_{j=1}^{p} (1 - \beta_j \gamma_j q_j)^{z_{ij}}$. If we average this over all n users

in our data, we have our CTR function, $g(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{p}(1-\beta_j\gamma_j q_j)^{z_{ij}}$ where the sum of money spent across the p websites $\sum_j \beta_j$ must be less than our total allowed budget B, and spending at all websites must be nonnegative. Thus maximizing CTR can be formalized as

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{p}(1-\beta_j\gamma_j q_j)^{z_{ij}} + \lambda\|\beta\|_1, \tag{4}$$

where λ is a tuning parameter with a one-to-one correspondence to our budget B. This is an instance of (1). Further, our nonnegativity constraint on the β_j values is ultimately unnecessary, since our solution will never optimize by setting β_j to a negative value. The minimization of the function ensures β_j will be, at minimum, zero; a negative β_j would actually increase the value of the objective function.

Our case study in Section 6 goes through the above application in detail for a potential Norwegian Cruise Lines (NCL) marketing campaign, including examples of the optimization for both reach and CTR. The case study demonstrates the use of several constraints that firms often wish to incorporate into their campaigns, such as allocating a certain percentage of budget to a given set of websites or maximizing CTR subject to reaching certain demographics. We are able to show that, not only does PAC demonstrate measurable, statistical improvements over existing methods, but it also incorporates constraints directly in the problem formulation, which to the best of our knowledge, no existing method can currently incorporate.

3.2 comScore Internet Browsing Data

The NCL online marketing campaign case study considered in this paper is implemented using a subset of the 2011 comScore Media Metrix data. comScore's data is a commercial, proprietary data set purchased through comScore and accessed through the Wharton Research Data Service (www.wrds.upenn.edu). comScore records daily webpage usage information from a panel of 100,000 Internet users, whose behavior is recorded anonymously by individual computer. Using these comScore by-computer records, we construct a matrix of all websites visited and the number of times each computer visited each website (and how many webpages were viewed at each visit) during a particular time period. This comScore data is commonly utilized in the marketing literature when Internet visitation is considered (e.g., Danaher, 2007; Liaukonyte et al., 2015; Montgomery et al., 2004; Park and Fader,

Category	CPM	No. Websites	Avg. Visits	Category	CPM	No. Websites	Avg. Visits
Community	2.10	24	5942	Online Shop	2.52	29	4563
E-mail	0.94	6	5321	Photos	1.08	6	5194
Entertainment	4.75	100	3349	Portal	2.60	36	45660
Fileshare	1.08	22	6670	Retail	2.52	49	6672
Gaming	2.68	75	3677	Service	2.52	16	8503
General News	6.14	12	4945	Social Network	0.56	25	10837
Information	2.52	58	6816	Sports	6.29	13	4227
Newspaper	6.99	12	2400	Travel	2.52	17	2304

Table 1: Number of websites in each of the 16 website categories for the January 2011 500-website filtered comScore data

2004; Paulson et al., 2018). The comScore Internet browsing data is analogous to the more commonly-known Nielsen ratings for television; like Nielsen, comScore collects not only the websites visited by users but also household demographic information such as income level, geographic area, size of the household, etc. Thus researchers can use the data to get an overall sense of Internet browsing both at individual levels (by particular machine) and at higher group levels (by demographic characteristics or website visitation). For our case study, this data is supplemented by comScore Inc.'s Media Metrix data from May 2010 (Lipsman, 2010) to provide average advertising costs (given as CPMs, or cost per thousand ad impressions) for each website by grouping the websites into common website categories.

The data used in the NCL case study utilizes website visits by the 100,000 comScore users during January 2011. To create a manageable dataset, we manually identify the 500 most-visited websites in January 2011 which also supported Internet display ads. We choose this time period for our case study to mimic a hypothetical yearly promotion for Norwegian Cruise Line's "wave season," which runs from January to March (with the bulk of advertising taking place in January). Thus, our filtered data contains a record of every computer which visited at least one of the 500 most-visited websites at least once (48,628 users) during the month of January. The NCL case study ultimately uses a matrix of 48,628 comScore users by 500 websites, where the matrix entries are the total number of webpages viewed by each user at each website during the month of January.

Table 1 provides the categorical makeup of the 500 websites in the January 2011 data set. It includes the sixteen broad categories of websites presented by Lipsman (2010): Social Networking, Portals, Entertainment, E-mail, Community, General News, Sports, Newspapers, Online Gaming, Photos, Filesharing, Information, Online Shopping, Retail, Service, and Travel. The CPM columns are the average CPM values provided for that website category from the Media Metrix data, while the Number of Websites columns provide the total number of websites in each category, and the Average Visits column provides the average number of visits during January 2011 to a website in that category by our comScore users.[2] Note that for simplicity, the CPM values given in Table 1 are taken from comScore Inc.'s Media Metrix May 2010 data, but in practice firms would likely have already obtained actual average CPMs for each individual website from previously collected data or directly from the advertiser.

4. Methodology and Algorithm

In this section we develop our PAC optimization algorithm using the following three steps. First, we use Taylor's Theorem to approximate $g(\boldsymbol{\beta})$ using a quadratic term. Second, we incorporate the linear coefficient constraint into the objective function, and finally we minimize the new, unconstrained criterion.

Given a current parameter estimate $\tilde{\boldsymbol{\beta}}$, our objective function can be approximated by $g(\boldsymbol{\beta}) \approx g(\tilde{\boldsymbol{\beta}}) + \tilde{\mathbf{d}}^T(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}}) + \frac{1}{2}(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})^T\tilde{\mathbf{H}}(\boldsymbol{\beta} - \tilde{\boldsymbol{\beta}})$, where $\tilde{\mathbf{H}}$ and $\tilde{\mathbf{d}}$ are respectively the Hessian and gradient of g at $\tilde{\boldsymbol{\beta}}$. Let $\mathbf{X} = \mathbf{D}^{1/2}\mathbf{U}^T$ and $\boldsymbol{Y} = \mathbf{X}(\tilde{\boldsymbol{\beta}} - \tilde{\mathbf{H}}^{-1}\tilde{\mathbf{d}})$ where $\tilde{\mathbf{H}} = \mathbf{U}\mathbf{D}\mathbf{U}^T$ represents the singular value decomposition of the Hessian. Then it is not hard to show that, up to an irrelevant additive constant, $g(\boldsymbol{\beta})$ is approximated by $\frac{1}{2}\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|^2$. Hence, we can approximate (2) by

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1 \quad \text{subject to} \quad \mathbf{C}\boldsymbol{\beta} = \mathbf{b}, \tag{5}$$

[2]For further details on the relationships among categories, see Table 4 in Appendix B for an overview of viewership correlations within and across each of the sixteen website categories during January 2011.

a constrained version of the standard lasso[3]. Thus solving (5), updating $\tilde{\boldsymbol{\beta}}$ with the new solution, and iterating, will solve (2) in a similar fashion to the so-called *iterative reweighted least squares algorithm* for fitting GLMs.

Unfortunately, even though many algorithms exist to fit the lasso, the constraint on the coefficients in (5) makes it difficult to directly solve. However, we can reformulate (5) as an unconstrained optimization problem. Let $\mathcal{A}$ represent an index set of size m corresponding to a subset of $\boldsymbol{\beta}$ and let $\mathbf{X}_{\mathcal{A}}$ and $\mathbf{X}_{\bar{\mathcal{A}}}$ respectively represent the columns of $\mathbf{X}$ corresponding to $\mathcal{A}$ and the complement of $\mathcal{A}$.[4] Further define $\boldsymbol{\beta}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right)$ and

$$\boldsymbol{\beta}_{\bar{\mathcal{A}}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2}\|\boldsymbol{Y}^* - \mathbf{X}^*\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 + \lambda\|\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}\right)\|_1, \tag{6}$$

where $\boldsymbol{Y}^* = \boldsymbol{Y} - \mathbf{X}_{\mathcal{A}}\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{b}$, $\mathbf{X}^* = \mathbf{X}_{\bar{\mathcal{A}}} - \mathbf{X}_{\mathcal{A}}\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}$. In this setting $\boldsymbol{\beta}_{\mathcal{A}}$ represents the m constrained coefficients, and $\boldsymbol{\beta}_{\bar{\mathcal{A}}}$ the $p - m$ remaining unconstrained coefficients. Then, we have the following lemma.

Lemma 2. *For any index set $\mathcal{A}$ such that $\mathbf{C}_{\mathcal{A}}$ is non-singular, the solution to (5) is given by $\boldsymbol{\beta} = (\boldsymbol{\beta}_{\mathcal{A}}{}^T, \boldsymbol{\beta}_{\bar{\mathcal{A}}}{}^T)^T$.*

Solving (6) still poses a significant challenge, because the final term in the criterion is non-separable in the coefficients so standard optimization approaches, such as coordinate descent, will fail. Fortunately, an alternative, more tractable criterion can be used to compute $\boldsymbol{\beta}_{\bar{\mathcal{A}}}$. For a given index set $\mathcal{A}$ and m-dimensional vector $\mathbf{s}$, define $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$ by:

$$\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2}\|\tilde{\boldsymbol{Y}} - \mathbf{X}^*\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 \ , \tag{7}$$

where $\tilde{\boldsymbol{Y}} = \boldsymbol{Y}^* + \lambda\mathbf{X}^-\left(\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)^T\mathbf{s}$, and $\mathbf{X}^-$ is a matrix such that $\mathbf{X}^{*T}\mathbf{X}^- = \mathbf{I}$. Equation (7) is a much simpler criterion to solve as it is a standard lasso objective function which can be optimized using a variety of techniques. We discuss some additional implementation details in handling this reformulation in Appendix D.

Then, Lemma 3 shows that, provided we are careful in our choice of $\mathcal{A}$ and $\mathbf{s}$, solving (7) will provide a solution to (5).

[3]To simplify the presentation of our algorithm we have assumed equality constraints in (5). However, by introducing slack variables, the same basic approach can be used to optimize over inequality constraints. See Appendix C for further details.

[4]To reduce notation we assume without loss of generality that the elements of $\boldsymbol{\beta}$ are ordered so that the first m correspond to $\mathcal{A}$.

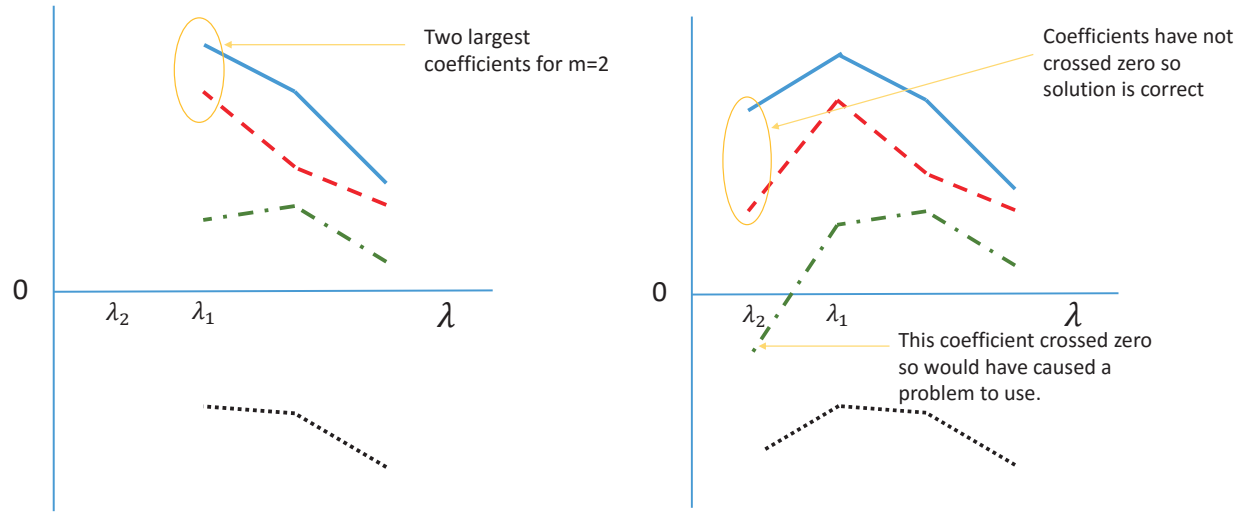

Figure 1: A simple illustration of the PAC algorithm with $p = 4$ variables and $m = 2$ constraints.

Lemma 3. *For any index set $\mathcal{A}$, it will be the case that $\boldsymbol{\beta}_{\bar{\mathcal{A}}} = \boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$ provided*

$$\mathbf{s} = sign\left(\boldsymbol{\beta}_{\mathcal{A},\mathbf{s}}\right), \tag{8}$$

where $\boldsymbol{\beta}_{\mathcal{A},\mathbf{s}} = \mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}\right)$. Hence, the solution to (5) is given by $\boldsymbol{\beta} = (\boldsymbol{\beta}_{\mathcal{A},\mathbf{s}}{}^T, \boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}{}^T)^T$.[5]

The proofs of Lemmas 2 and 3 are provided in Appendix E. There is a simple intuition behind Lemma 3. The difficulty in computing (6) lies in the non-differentiability (and non-separability) of the second ℓ_1 penalty. However, if (8) holds, then for any $\boldsymbol{\theta}$ close to $\boldsymbol{\beta}_{\bar{\mathcal{A}}}$, $\|\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}\right)\|_1 = \mathbf{s}^T\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}\right)$. Thus we can replace the ℓ_1 penalty by a differentiable term which no longer needs to be separable.

Of course the key to this approach is to select $\mathcal{A}$ and $\mathbf{s}$ such that (8) holds, which appears challenging given that $\mathbf{s}$ is a function of the unknown solution. However, choosing $\mathcal{A}$ and $\mathbf{s}$ turns out to be relatively simple in practice. Consider Figure 1, which illustrates our approach on a toy example involving $p = 4$ coefficients (the four colored lines), and $m = 2$ constraints. We generate the PAC solution over a decreasing grid of values for λ and the left-hand plot illustrates the solution up to $\lambda = \lambda_1$. To compute the PAC coefficients at $\lambda = \lambda_2$, we select $\mathcal{A}$ corresponding to the $m = 2$ largest coefficients in absolute terms (in this

[5] sign($\mathbf{a}$) is a vector of the same dimension as $\mathbf{a}$ with the i^{th} element equal to 1 or -1 depending on the sign of a_i.

case blue solid and red dashed) and set $\mathbf{s}$ equal to their current signs (both positive here). Thus, $\boldsymbol{\beta}_{\mathcal{A}}$ corresponds to the blue and red coefficients, while $\boldsymbol{\beta}_{\bar{\mathcal{A}}}$ represents the remaining two coefficients. In the right-hand plot we have computed the solution at λ_2 using (7). Since the blue and red coefficients are still positive, one can immediately observe that (8) holds, so we have the correct solution.

Crucially we use the fact that the coefficient paths are continuous in λ so, provided the step size from λ_1 to λ_2 is small enough, we are guaranteed that the signs of the largest m coefficients will remain the same. If our step size is too large, then it is possible that one of the coefficients in $\mathcal{A}$ may change sign. For example, the right-hand plot in Figure 1 shows that if we had selected the green dash-dot coefficient in $\mathcal{A}$, then the sign would have switched between λ_1 and λ_2. However, in such a situation one immediately observes that the solution is incorrect, and the correct solution can then be computed by choosing a smaller step size in λ. In this case a step size half as large would have allowed the sign of the green coefficient to remain positive. It is important to note that $\mathcal{A}$ will change for each step, so we are free to update the index set with the coefficients that are least likely to switch signs, i.e. those furthest from zero. In practice, provided the step size is not too large, this approach works well, with very few instances of sign changes. Algorithm 1 formally summarizes the PAC approach for solving (5).

Algorithm 1 PAC with Equality Constraints

1. Initialize $\boldsymbol{\beta}_0$ by solving (5) using $\lambda_0 = \lambda_{\max}$.
2. At step k select $\mathcal{A}_k$ and $\mathbf{s}_k$ using the largest m elements of $|\boldsymbol{\beta}_{k-1}|$ and set $\lambda_k \leftarrow 10^{-\alpha}\lambda_{k-1}$, where $\alpha > 0$ controls the step size.
3. Compute $\boldsymbol{\beta}_{\bar{\mathcal{A}}_k,\mathbf{s}_k}$ by solving (7). Let $\boldsymbol{\beta}_{\mathcal{A}_k,\mathbf{s}_k} = \mathbf{C}_{\mathcal{A}_k}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}_k}\boldsymbol{\beta}_{\bar{\mathcal{A}}_k,\mathbf{s}_k}\right)$.
4. If (8) holds then set $\boldsymbol{\beta}_k = \begin{bmatrix} \boldsymbol{\beta}_{\mathcal{A}_k,\mathbf{s}_k} \\ \boldsymbol{\beta}_{\bar{\mathcal{A}}_k,\mathbf{s}_k} \end{bmatrix}$, $k \leftarrow k+1$ and return to 2.
5. If (8) does not hold then one of the largest m elements of $\boldsymbol{\beta}_{k-1}$ has changed sign so our step size was too large. Hence, set $\lambda_k \leftarrow \lambda_{k-1} - \frac{1}{2}(\lambda_{k-1} - \lambda_k)$ and return to 3.
6. Iterate until $\lambda_k < \lambda_{\min}$.

Step 3 of the algorithm is the main computational component, but $\boldsymbol{\beta}_{\bar{\mathcal{A}}_k,\mathbf{s}_k}$ is easy to compute because (7) is just a standard lasso criterion, so we can use any one of a number of optimization tools. The initial solution, $\boldsymbol{\beta}_0$, can be computed by noting that as $\lambda \to \infty$ the solution to (5) will be

$$\arg\min_{\boldsymbol{\beta}} \|\boldsymbol{\beta}\|_1 \quad \text{such that} \quad \mathbf{C}\boldsymbol{\beta} = \mathbf{b}, \tag{9}$$

which is a linear programming problem that can be efficiently solved using standard algorithms. We also implement a reversed version of this algorithm where we first set $\lambda_0 = \lambda_{\min}$, compute $\boldsymbol{\beta}_0$ as the solution to a quadratic programming problem, and then increase λ at each step until $\lambda_k > \lambda_{\max}$. We discuss some additional implementation details in Appendix D. This approach can be extended in much the same way for inequality constraints by incorporating slack variables. See Appendix C for details.

5. Simulation Studies

In this section, we present simulation results to compare PAC's performance relative to unconstrained lasso fits. We choose the lasso due to its versatility, particularly in handling high-dimensional problems, as well as its widespread use in statistical modeling. Thus the results presented here correspond to data generated from a standard Gaussian linear regression with $g(\boldsymbol{\beta}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$. (For further comparisons with data generated from a binomial logistic regression model with $g(\boldsymbol{\beta})$ equal to the corresponding loglikelihood, see Appendix F). In Section 5.1 we show that, when the true underlying parameters satisfy equality constraints, PAC can yield significant improvements in prediction accuracy over unconstrained methods. In addition, Section 5.2 shows that these improvements are robust in the sense that, even when the true parameters violate some of the constraints, PAC still yields superior estimates. Finally, we demonstrate the computational efficiency of the PAC algorithm relative to a quadratic programming implementation in Section 5.3.

5.1 PAC Comparison to Existing Lasso Methods

To demonstrate the use of PAC in practice, we consider six simulation settings: three different combinations of observations (n) and predictors (p), corresponding to both classical and high-dimensional problems, and two different correlation structures, $\rho_{jk} = 0$ and $\rho_{jk} = 0.5^{|j-k|}$ (where ρ_{jk} is the correlation between the jth and kth variables). The training data sets were

produced using a random design matrix generated from a standard normal distribution. For each setting we randomly generated a training set, fit each method to the data, and computed the error over a test set of $N = 10,000$ observations, where the error metric used is the root mean squared error: RMSE $= \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{Y}_i - E(Y_i|X_i)\right)^2}$. This process was repeated 100 times for each of the six settings.

In all cases, the m-by-p constraint matrix $\mathbf{C}$ and the constraint vector $\mathbf{b}$ were randomly generated from a normal distribution. The true coefficient vector $\boldsymbol{\beta}^*$ was produced by first generating $\boldsymbol{\beta}^*_{\bar{\mathcal{A}}}$ using 5 non-zero random uniform components and $p - m - 5$ zero entries and then computing $\boldsymbol{\beta}^*_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}}^{-1}(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}^*_{\bar{\mathcal{A}}})$. Note that this process resulted in $\boldsymbol{\beta}^*$ having at most $m + 5$ non-zero entries and ensured that the constraints held for the true coefficient vector. For each set of simulations, the optimal value of λ was chosen by minimizing error on a separate validation set, which was independently generated using the same parameters as for the corresponding training data.

For each method we explored three combinations of n, p, and m: a low-dimensional setting with few constraints ($n = 100, p = 50$ and $m = 5$), a higher-dimensional problem with few constraints ($n = 50, p = 500$ and $m = 10$), and a high-dimensional problem with more constraints ($n = 50, p = 100$ and $m = 30$). The test error values for the six resulting settings are displayed in Table 2. For each method, we compared results from four different approaches: the standard unconstrained but penalized fit, i.e. the lasso as given in (1) (Friedman et al., 2010), PAC, the relaxed lasso, and the relaxed PAC. The latter two methods use a two-step approach in an attempt to reduce the overshrinkage problem commonly exhibited by the ℓ_1 penalty. In the first step, the given method is used to select a candidate set of predictors. In the second step, the final model is produced using an unshrunk ordinary least squares fit on the variables selected in the first step. The relaxed PAC coefficients are still optimized subject to the linear constraints.

Even in the first setting, with a low value for m, PAC shows highly statistically significant improvements over the unconstrained methods. Both relaxed methods display lower error rates than their unrelaxed counterparts, and the correlated design structure does not change the relative rankings of the four approaches. As one would expect, in the second setting, given the low ratio of m relative to p, PAC only shows small improvements over its uncon-

	ρ	Lasso	PAC	Relaxed Lasso	Relaxed PAC
$n = 100, p = 50$	0	0.59(0.01)	0.52(0.01)	0.45(0.01)	0.30(0.01)
$m = 5$	$0.5^{\|i-j\|}$	0.63(0.01)	0.49(0.01)	0.57(0.02)	0.35(0.01)
$n = 50, p = 500$	0	3.38(0.07)	3.33(0.09)	3.27(0.08)	3.16(0.10)
$m = 10$	$0.5^{\|i-j\|}$	2.58(0.07)	2.33(0.09)	2.44(0.07)	2.09(0.09)
$n = 50, p = 100$	0	6.59(0.07)	1.19(0.03)	6.75(0.08)	0.96(0.03)
$m = 60$	$0.5^{\|i-j\|}$	6.51(0.07)	1.31(0.04)	6.66(0.09)	0.98(0.03)

Table 2: Average RMSE over 100 training data sets, for four lasso methods tested in three different simulation settings and two different correlation structures. The numbers in parentheses are standard errors.

strained counterparts. However, this setting shows the PAC algorithm is still efficient enough to optimize the constrained criterion even for large data sets and very high-dimensional data. The final setting is more favorable to PAC, because m is much larger, and thus there is the potential to produce significantly more accurate regression coefficients by correctly incorporating the constraints. However, this is also a computationally difficult setting for PAC, because a large value of m causes the coefficient paths to be highly variable. Nevertheless, the large improvements in accuracy for both PAC and relaxed PAC demonstrate that our algorithm is quite capable of dealing with this added complexity.

5.2 Violations of Constraints

The results presented in the previous section all correspond to an ideal situation where the true regression coefficients exactly match the equality constraints. Here, we also investigate the sensitivity of PAC to deviations of the regression coefficients from the assumed constraints. In particular we generate the true regression coefficients according to

$$\mathbf{C}\boldsymbol{\beta}^* = (\mathbf{1} + \mathbf{u}) \cdot \mathbf{b}, \tag{10}$$

where $\mathbf{u} = (u_1, \ldots, u_m)$, $u_l \sim \text{Unif}(0, a)$ for $l = 1, \ldots m$, and the vector product is taken pointwise. The PAC and relaxed PAC were then fit using the usual (but in this case incorrect) constraint, $\mathbf{C}\boldsymbol{\beta} = \mathbf{b}$.

Table 3 reports the new RMSE values for three Gaussian settings under the $\rho = 0$ correlation structure, corresponding to the three settings of Table 2. Again, the first two settings are used for demonstration purposes to show PAC performs well even in standard

	a	Lasso	PAC	Relaxed Lasso	Relaxed PAC
$n = 100$	0.25	0.59(0.01)	0.52(0.01)	0.44(0.01)	0.31(0.01)
$p = 50$	0.50	0.59(0.01)	0.53(0.01)	0.44(0.01)	0.33(0.01)
$m = 5$	0.75	0.59(0.01)	0.54(0.01)	0.44(0.01)	0.36(0.01)
	1.00	0.59(0.01)	0.55(0.01)	0.44(0.01)	0.39(0.01)
$n = 50$	0.25	3.35(0.07)	3.31(0.09)	3.27(0.08)	3.13(0.10)
$p = 500$	0.50	3.39(0.07)	3.34(0.09)	3.31(0.09)	3.17(0.10)
$m = 10$	0.75	3.35(0.07)	3.30(0.09)	3.29(0.08)	3.09(0.10)
	1.00	3.33(0.07)	3.30(0.09)	3.25(0.08)	3.09(0.10)
$n = 50$	0.25	6.59(0.07)	1.20(0.03)	6.72(0.08)	0.97(0.03)
$p = 100$	0.50	6.60(0.07)	1.21(0.03)	6.73(0.08)	0.98(0.03)
$m = 60$	0.75	6.59(0.07)	1.26(0.03)	6.75(0.08)	1.03(0.03)
	1.00	6.61(0.07)	1.29(0.03)	6.77(0.08)	1.06(0.03)

Table 3: Average RMSE over 100 training data sets in three different simulation settings using the $\rho = 0$ correlation structure. The numbers in parentheses are standard errors. The true regression coefficients were generated according to (10).

or very high-dimensional settings, while the last is a setting with a very large number of constraints to demonstrate robustness even when $n < m$. We tested four values for a: $0.25, 0.50, 0.75$ and 1.00. The largest value of a corresponds to a 50% average error in the constraint. The results suggest that PAC and relaxed PAC are surprisingly robust to random violations in the constraints. While both methods deteriorated slightly as a increased, they were still both superior to their unconstrained counterparts for all values of a and all settings.

5.3 Efficiency of PAC Algorithm

In this section we demonstrate the efficiency of the PAC algorithm relative to a standard quadratic programming solution. Quadratic programming provides an excellent comparison since, as shown in the preceding section, PAC relies on approximating $g(\beta)$ with a sum of squares term. In addition, quadratic programming is a well-established option for optimizing constrained problems and can even be used in high-dimensional settings like the ones proposed in Section 5.1.

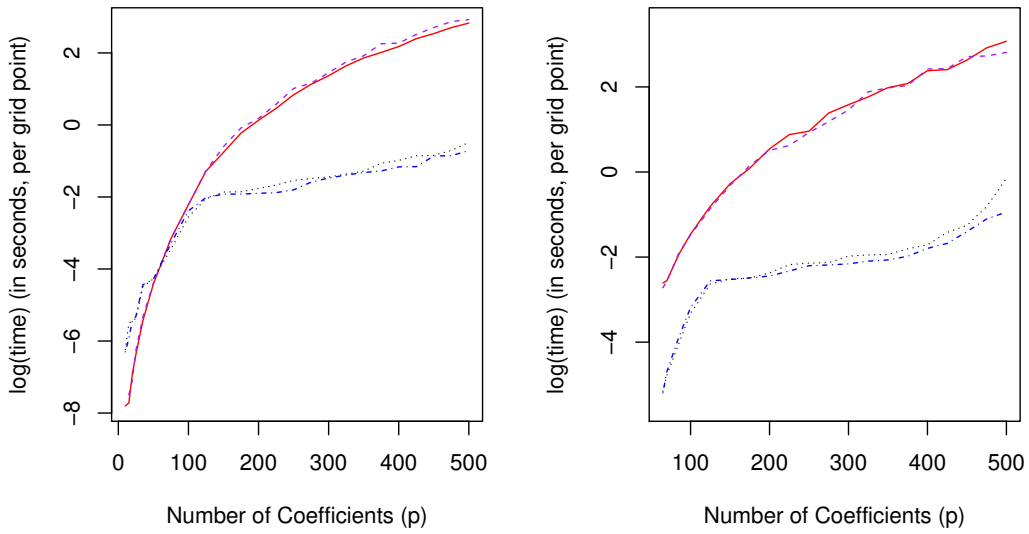

Figure 2: Plots of average time per lambda value (each grid point where a solution was calculated), on a logarithmic scale, for solutions over a range of p in two settings: our first setting, $n = 100$ and $m = 5$ (left) and our third setting, $n = 50$ and $m = 60$ (right). The quadratic solution is given both for data with no correlation structure (solid red line) as well as data with the correlation structure of Table 2 (dashed purple line); likewise PAC is also given with no correlation in the data (dotted black line) and with correlation (dotted-dashed blue line).

Figure 2 shows how computational efficiency dramatically increases for PAC relative to quadratic programming as the number of coefficients p increases.[6] Here, two general settings are plotted: (1) the first setting of Table 2, where $n = 100$ and $m = 5$ to demonstrate a low-constraint problem, and (2) the third setting of Table 2, where $n = 50$ and $m = 60$ to demonstrate a higher-constraint problem. Further, we also consider the two correlation structures to the data used in Table 2. In all cases, Figure 2 demonstrates that an increase in predictors can dramatically increase computational time for quadratic programming. While computation time increases for PAC as well, it is not nearly as dramatic. Thus PAC represents an efficient method to optimize constrained problems on increasingly large scales.

[6]To measure computational efficiency between PAC and quadratic programming, both were implemented in R on a personal laptop computer using a 2.59 GHz i7 processor.

6. Case Study: Cruise Line Internet Marketing Campaign

In this section, we apply PAC to an exemplar real-world case study for Norwegian Cruise Lines (NCL). Each year, the cruise industry advertises for its annual "wave season," a promotional cruise period which begins in January. NCL is among the cruise lines that participate heavily in wave season (Satchell, 2011). Because consumers who are interested in booking a cruise often use travel aggregation sites like Orbitz and Priceline to compare offerings across multiple options, and cruise lines frequently want to make the sales known to potential customers without sacrificing clickthrough to their websites, this case study is ideal for demonstrating PAC subject to various constraints. Since the wave season sale begins in January, we consider the comScore data from January 2011 to approximate an NCL advertising campaign. In Section 6.1 we demonstrate PAC in comparison to other possible approaches when NCL wishes to maximize reach subject to constraints. Section 6.2 demonstrates PAC in the setting in which NCL wishes to maximize clickthrough rate.

6.1 Internet Media Metric 1: Maximizing Reach

For real-life advertising campaigns, firms attempt to leverage business insights in order to improve their advertising campaigns by reaching target customers. Although NCL does want to reach as many potential cruisers as possible, they also know which characteristics make a consumer more likely to purchase a cruise. For example, because consumers who are interested in booking a cruise often use travel aggregation sites like Orbitz and Priceline to compare offerings, NCL will reach more likely customers at these websites. Because of this, NCL may want to allocate at least a minimum amount of budget (say, 20%) to a set of major aggregate travel websites. This induces a constraint on the optimization; NCL wishes to optimize total overall reach, but subject to 20% of budget being spent at the set of aggregate travel websites. In our January 2011 comScore data, we have eight major aggregate travel websites.

Formally, if firms have a subset S of websites on which they know they want to advertise and thus dedicate a minimum proportion of their budget to this subset, this fits very naturally into our constraint matrix setup by defining $\mathbf{C}_S^T\boldsymbol{\beta} \geq b_S B$, where $\mathbf{C}_S$ defines the websites in the subset S, and b_S is the proportion of budget the firm wishes to allocate to the subset S.

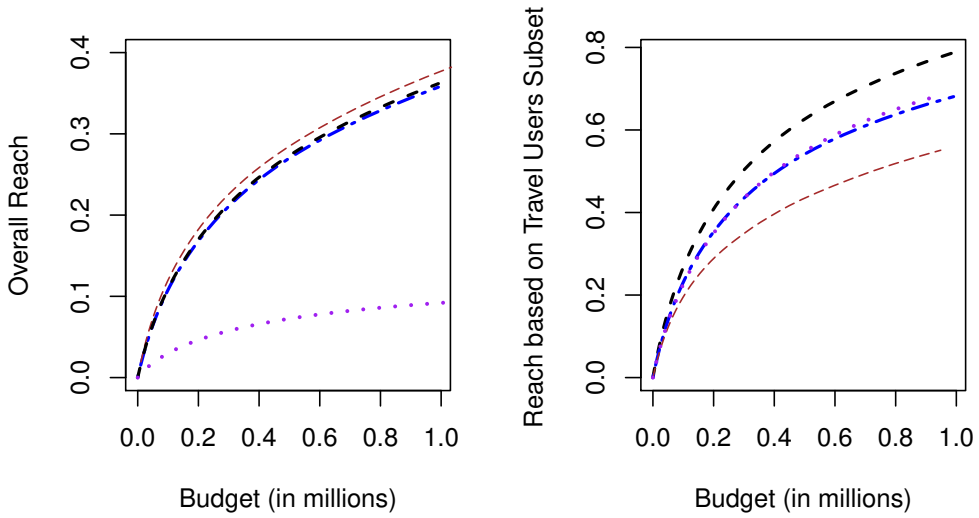

Figure 3: Plots of the reach calculated for 500 websites on the full data set (left) and the subset of travel users (right) using four methods: PAC with constraints (thick dashed black), PAC with no constraints (thin dashed brown), ELMSO with constraints (thick dotted-dashed blue), and cost adjusted allocation (dotted purple).

Figure 3 shows the results for reach as a function of budget, both on the overall data set and on the target users, the ones who have visited at least one of the eight aggregate travel websites. Here, we compare both the constrained and unconstrained PAC to two naive methods: equal budget allocation across the eight travel websites and cost-adjusted allocation across those websites. In addition, we compare to ELMSO (Paulson et al., 2018), which optimizes reach based on modeling views of Internet ads as a Poisson arrival process. In this way, ELMSO is similar to an unconstrained PAC, except the latter method assumes a binomial process rather than Poisson. We implement a constrained version of ELMSO. While PAC can handle the 20% minimum budget allocation directly through a single constraint (where $\mathbf{C}_S$ identifies the aggregate travel sites and $\mathbf{b} = 0.20B$), ELMSO cannot implement constraints of this form. Instead, ELMSO places a minimum budget, 2.5%, at each of the aggregate travel websites, thus ensuring at least 20% of the budget overall is allocated to these sites.

As Figure 3 shows, once constraints are introduced, PAC consistently outperforms ELMSO and the naive methods. Because the PAC optimization incorporates the budget allocation constraint directly, it has more flexibility in allocating across the subset of websites. ELMSO is forced to allocate a minimum to each website, whether that website is preferred over others or not. Most importantly, however, on the target subset of users (those who visit travel websites), constrained PAC very clearly outperforms all other methods, but overall reach is relatively unchanged between the constrained and unconstrained PAC methods. This means NCL is reaching its target customers at the aggregate sites without sacrificing much overall reach. By contrast, the naive allocation methods actually slightly outperforms the constrained ELMSO on the aggregate travel users' subset. PAC provides an option to maximize reach over the target consumer base without losing other potential customers at the non-aggregate travel websites.

6.2 Internet Media Metric 2: Maximizing Clickthrough

In this section, we consider an alternative performance metric: allocating budget to maximize clickthrough, as described in Section 3.1. Here, NCL wishes to maximize the number of people who click on their ad subject to a given budget. Clickthrough rates (CTR) are a more recent area of interest in the marketing literature, and as such have been far less explored than the traditional reach setup.

6.2.1 Clickthrough Rate

To implement this analysis, we compute CTR using the binomial formulation in (4). We use MediaMind's 2011-2012 Global Benchmarks Report (MediaMind 2012) to estimate q_j, the probability that a user clicks on an ad at website j given it is shown to them. This report provides average display ad clickthrough rates by industry for 2011-2012. Thus, we first group the websites by industry, then use the industry average for q_j. In practice, advertisers would have specific values for q_j and would update these throughout the campaign.

We first consider a campaign analogous to the one in Section 6.1 above, where instead of maximizing reach subject to a constraint on the subset of aggregate travel websites, NCL wishes to maximize CTR subject to the same budget constraint. As shown in Section 3.1, PAC does this directly, but we are not aware of any other publicly available method that

can maximize CTR on a large-scale problem such as a 500-website optimization. However, while ELMSO is designed for reach only, we can modify the reach criterion to incorporate a CTR parameter by multiplying the probability of an ad appearance by the probability a user will click on it (our CTR parameter, q_j). This is not directly a CTR optimization, since CTR is defined as the proportion of users who click on an ad at least once and thus does not fit neatly into a Poisson arrival process definition, but in the absence of other analogous methods, it works well for comparison purposes.

Figure 4 shows CTR as a function of budget, both on the overall data set and on the target users who have visited the aggregate travel websites. Again, we compare both constrained and unconstrained PAC to the two naive methods: equal budget allocation across the eight aggregate travel websites and cost-adjusted allocation across these websites. In addition, we again compare to the constrained implementation of the ELMSO CTR proxy. The results are qualitatively very similar to those in Figure 3, with PAC still outperforming the other approaches. Overall clickthrough is much lower than reach, as expected since only a few users who see the ad will click on it, but for the subset of aggregate travel site visitors, CTR is almost double that of the overall advertising campaign.

6.2.2 Clickthrough Rate subject to Multiple Constraints

Here we examine a setting involving optimizing CTR subject to multiple different constraints. Suppose that NCL wishes to target a particular subset of consumers H by ensuring that these consumers receive K times the average views relative to those not in H. PAC can incorporate this constraint using:

$$\frac{1}{n_H}\sum_{i\in H}\sum_{j=1}^{p} z_{ij}\gamma_j\beta_j \geq K\frac{1}{n-n_H}\sum_{i\notin H}\sum_{j=1}^{p} z_{ij}\gamma_j\beta_j, \tag{11}$$

where n_H is the number of people in the target group, and $z_{ij}\gamma_j\beta_j$ represents the expected number of ad appearances to person i at website j (since z_{ij} is the number of times person i views pages at website j, and $\gamma_j\beta_j$ is the probability the ad appears to user i at website j on any given visit). As a specific application of (11), in 2011 NCL created special single-occupancy rooms to appeal to solo cruise travelers, a niche which had been previously unexplored by the cruise industry. Historically, cruise lines had focused on double-occupancy rooms, requiring solo travelers to room with a stranger or incur the cost of booking a room

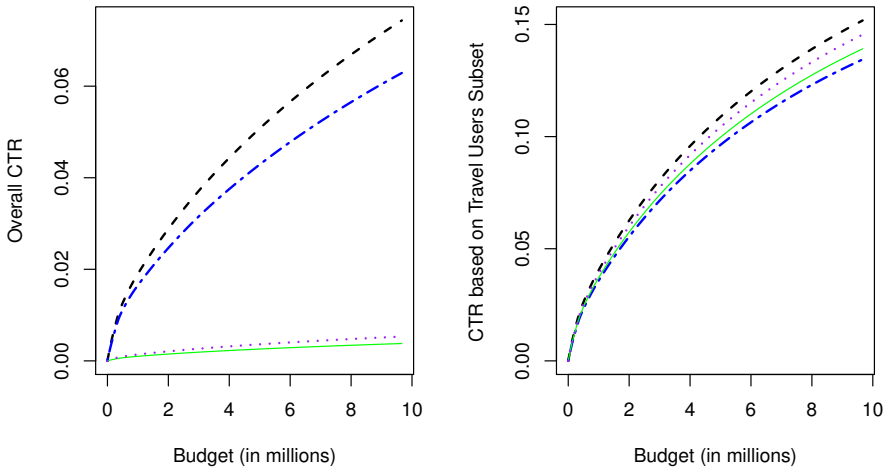

Figure 4: Plots of the clickthrough rate for 500 websites on (left) the full data set and (right) the subset of travel users using four methods: PAC with constraints (thick dashed black), constrained ELMSO proxy (thick dotted-dashed blue), cost adjusted allocation (dotted purple), and equal allocation (solid green).

for two people (Clements, 2013; CruiseCritic, 2017). Capitalizing on this niche can be extremely valuable; Cruise Lines International Association expected 23 million solo travelers in 2015 in North America alone (Ambroziak, 2015), and solo travelers accounted for 24% of total travelers in 2015 (Post, 2017). Hence, NCL might wish to optimzie CTR subject to (11) with $K = 2$ and H chosen to include households without children (because solo travelers necessarily travel without children).

In addition, NCL could have several other constraints. For example, solo cruise line travelers typically fall into an age range of 30-59 with an income of \$35,000 to \$70,000 (Clements, 2013). Hence, we constrain our optimization to ensure twice as many average views come from the target group (that is, single-person households in the 30-59 age range with incomes between \$35,000 and \$70,000) as from all others. To illustrate a geographical constraint, we also constrain average views of those from the "West" region of the US to be at least as large as those from other parts of the country. Further, we add six additional constraints, corresponding to the seven income levels provided by comScore, to ensure average ad views at a higher income level are always at least as large as average ad views at a lower income level.

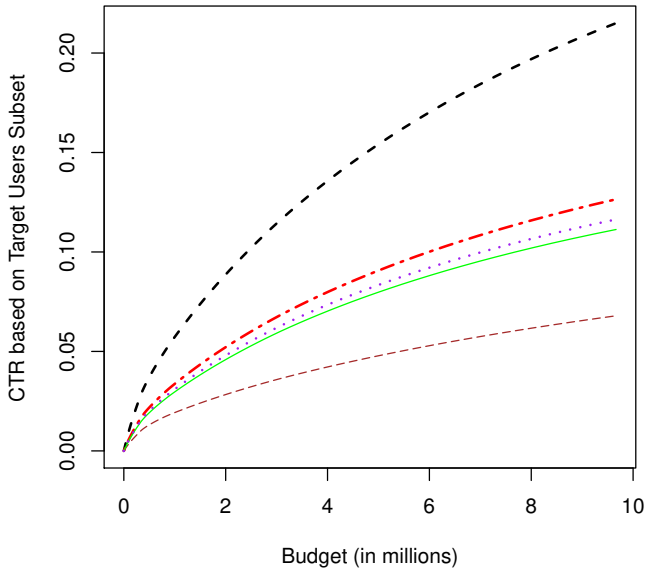

Figure 5: Clickthrough rates on travel users aged 30-59 with income between $35,000 and $70,000 using five methods: unconstrained PAC (dashed brown), constrained PAC (thick dashed black), constrained PAC using only the travel users (dotted-dashed red), allocation by cost across travel websites (dotted purple), and equal allocation across travel websites (solid green).

Finally, we constrain our optimization to force 20% of the budget to our aggregate travel websites, as we previously considered. This ultimately results in ten constraints, though firms would often include many more such target groups.

Figure 5 shows CTR as a function of budget on the target users who have visited the aggregate travel websites, are between the ages of 30 and 59, and have annual incomes between $35,000 and $70,000. Again, we compare both constrained and unconstrained PAC to the two naive methods: equal allocation across the eight travel websites and cost-adjusted allocation across these websites. Finally, we implement the previous constrained PAC method, which only includes the single constraint of 20% of the budget to aggregate travel websites. ELMSO can not be implemented with these constraints. We see that both versions of the constrained PAC provide the highest CTR on this target group, with the most highly-constrained fit generating by far the largest jump.

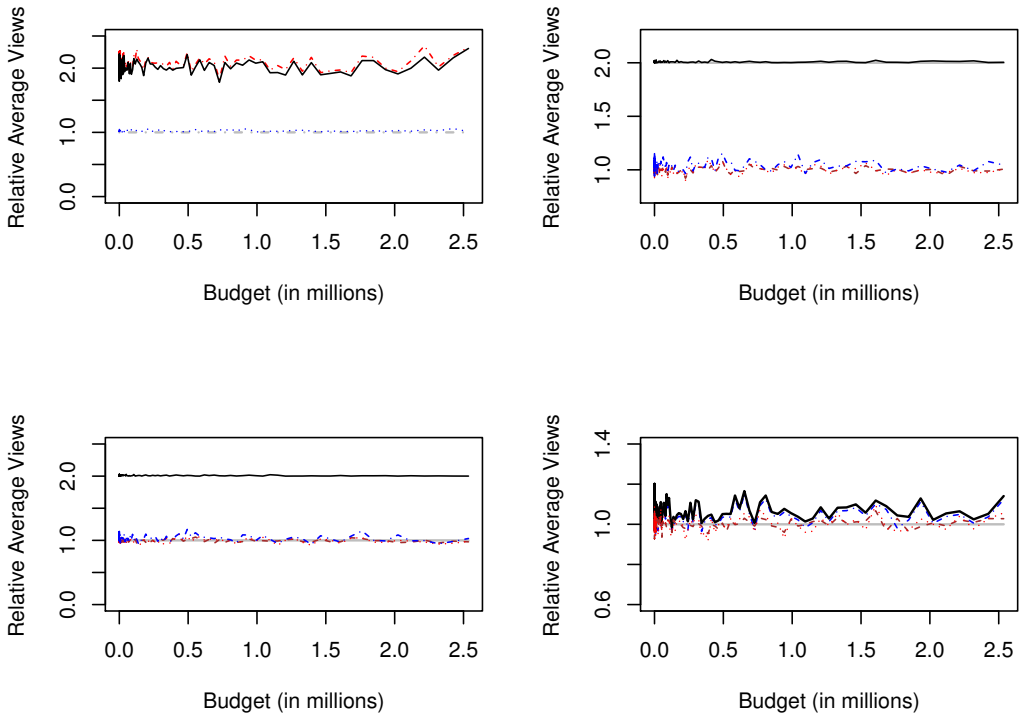

Figure 6: Plots of the ratio of average ad views between target demographic groups. A solid gray line is plotted at 1 for comparison.

Figure 6 demonstrates that PAC is effectively enforcing the various constraints that we considered. In the upper left corner, the plot shows the ratio for our constrained PAC optimization of average ad views by users in successive income ranges. Each line represents a comparison between a given income bracket and the *lowest* income bracket ("less than $15,000"); for example, the dotted blue line is the ratio of average ad views in the "$25,000 to $34,999" income bracket relative to the "less than $15,000" income bracket. Without constraints, we would expect all lines to show a ratio of 1, indicating no randomly-chosen member of a particular income group is more likely to view the ad than any other. However, because we have forced successively higher income brackets to have more average ad views than the previous bracket, we see an increase in the ratio. Most notably, a jump occurs at the "$35,000 to $49,999" (solid black line) income group, because the additional constraint for our target group begins to take effect at incomes above $35,000. We have also plotted

the highest income bracket ("above $100,000") with a dotted-dashed red line. As expected this line never falls below the "$35,000 to $49,999" bracket.

The top right plot shows a comparison between constrained PAC (solid black) and unconstrained PAC (dashed brown), constrained ELMSO (dotted-dashed blue), and unconstrained ELMSO (dotted red) in the ratio of average ad views by single-person household users in the 30-59 age range with incomes between $35,000 and $70,000, relative to all other users. The bottom left plot is identical to the top right except it provides the ratio of average ad views by users in the households without children compared to households with children. Recall in both cases our constraint forced the optimization to allocate twice as many average ad views in the target group as in any other. In both plots the black line sitting consistently at 2 demonstrates the constraint is holding for PAC, while the other methods all hover around a ratio of 1.

Finally, the bottom right plot matches the previous two plots showing the ratio of average ad views by users in the "West" region relative to all other users. Recall our constraint here forced the optimization to ensure at least as many average ad views in this group as in any other group. As the figure demonstrates, the constrained PAC method stays consistently above a ratio of 1, while the other methods all vary around a ratio of 1.

7. Conclusion

In this paper we have illustrated a few of the wide range of statistical applications for the PAC formulation and developed a computationally efficient path algorithm for computing its solutions. Our simulation results show the PAC estimates generally outperform the unconstrained estimates, not only when the constraints hold exactly, but also when there is some reasonable error in the assumption on the constraints. Furthermore, we show PAC can easily be used in practice to accommodate real-world considerations, particularly in the case of Internet advertising budget allocation problems. We demonstrate via our exemplar case study that PAC actually presents a significant advantage over current methods, which cannot handle constraints directly. PAC can handle multiple linear constraints with no additional ad hoc optimization requirements, and it is not limited to a single optimization criterion. Firms can run campaigns to maximize reach, clickthrough rate, or any other metric for which they have a known function. We are currently exploring other applications of PAC.

A. Proof of Lemma 1

Since $\boldsymbol{D}$ has full column rank, by reordering the rows if necessary, we can write $\boldsymbol{D}$ as

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{D}_1 \\ \boldsymbol{D}_2 \end{bmatrix}$$

where $\boldsymbol{D}_1 \in \mathbb{R}^{p\times p}$ is an invertible matrix and $\boldsymbol{D}_2 \in \mathbb{R}^{r-p\times p}$. Then,

$$\begin{aligned} \frac{1}{2}\left\|\boldsymbol{Y} - \tilde{\mathbf{X}}\boldsymbol{\theta}\right\|_2^2 + \lambda\left\|\boldsymbol{D}\boldsymbol{\theta}\right\|_1 &= \frac{1}{2}\left\|\boldsymbol{Y} - \tilde{\mathbf{X}}\boldsymbol{D}_1^{-1}\boldsymbol{D}_1\boldsymbol{\theta}\right\|_2^2 + \lambda\left\|\boldsymbol{D}_1\boldsymbol{\theta}\right\|_1 + \lambda\left\|\boldsymbol{D}_2\boldsymbol{\theta}\right\|_1 \\ &= \frac{1}{2}\left\|\boldsymbol{Y} - (\tilde{\mathbf{X}}\boldsymbol{D}_1^{-1})\boldsymbol{D}_1\boldsymbol{\theta}\right\|_2^2 + \lambda\left\|\boldsymbol{D}_1\boldsymbol{\theta}\right\|_1 + \lambda\left\|\boldsymbol{D}_2\boldsymbol{D}_1^{-1}\boldsymbol{D}_1\boldsymbol{\theta}\right\|_1 \end{aligned}$$

Using the change of variables

$$\boldsymbol{\beta}_1 = \boldsymbol{D}_1\boldsymbol{\theta}, \quad \boldsymbol{\beta}_2 = \boldsymbol{D}_2\boldsymbol{D}_1^{-1}\boldsymbol{D}_1\boldsymbol{\theta} = \boldsymbol{D}_2\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_1, \quad \text{and} \quad \boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2 \end{bmatrix},$$

we can rewrite the generalized lasso problem as follows:

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{2}\left\|\boldsymbol{Y} - \tilde{\mathbf{X}}\boldsymbol{\theta}\right\|_2^2 + \lambda\left\|\boldsymbol{D}\boldsymbol{\theta}\right\|_1 &= \min_{\boldsymbol{\beta}\in\mathbb{R}^r}\left\{\frac{1}{2}\left\|\boldsymbol{Y} - \tilde{\mathbf{X}}\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_1\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}\right\|_1 \;\middle|\; \boldsymbol{D}_2\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2 = \mathbf{0}\right\}, \\ &= \min_{\boldsymbol{\beta}\in\mathbb{R}^r}\left\{\frac{1}{2}\left\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}\right\|_1 \;\middle|\; \mathbf{C}\boldsymbol{\beta} = \mathbf{0}\right\}, \end{aligned}$$

where $\mathbf{X} = \begin{bmatrix}\tilde{\mathbf{X}}\boldsymbol{D}_1^{-1} & \mathbf{0}\end{bmatrix}$ and $\mathbf{C} = \begin{bmatrix}\boldsymbol{D}_2\boldsymbol{D}_1^{-1} & -\boldsymbol{I}\end{bmatrix}$. Note that $\boldsymbol{\theta} = [\boldsymbol{D}^{-1} \;\; \mathbf{0}]\begin{bmatrix}\boldsymbol{\beta}_1 \\ \boldsymbol{\beta}_2\end{bmatrix}$, and thus, the generalized lasso is a special case of the constrained lasso, which is a standard formulation for PAC.

B. comScore Data Details

Table 4 provides an overview of correlation in viewership among the 16 website groups in the data for the NCL case study. The table includes both within-group correlations and among-group correlations. Within-group correlation in the table is calculated by taking the mean of all absolute correlations between websites in a particular group. These are displayed on the diagonal. For example, in January 2011 the Gaming category shows extremely low average correlation in viewership with other Gaming websites (0.01). Most groups show very

Category	Com	Email	Ent	File	Game	Gen	Info	News	Onl	Photo	Port	Ret	Serv	Soc	Sport	Travel
Community	0.02	0.06	0.27	0.15	0.23	0.57	0.43	0.14	0.16	0.24	0.49	0.35	0.07	0.21	0.20	0.12
Email	.	0.00	0.12	0.04	0.14	0.05	0.08	0.04	0.04	0.02	0.89	0.07	0.09	0.07	0.05	0.03
Entertainment	.	.	0.02	0.38	0.38	0.61	0.53	0.32	0.22	0.23	0.74	0.28	0.12	0.19	0.39	0.09
Fileshare	.	.	.	0.05	0.25	0.06	0.53	0.09	0.09	0.10	0.28	0.08	0.05	0.15	0.11	0.04
Gaming	.	.	.	.	0.01	0.13	0.96	0.11	0.16	0.11	0.25	0.15	0.89	0.32	0.10	0.92
General News	.	.	.	.	.	0.05	0.61	0.31	0.08	0.06	0.62	0.11	0.15	0.16	0.53	0.14
Information	.	.	.	.	.	.	0.02	0.24	0.52	0.12	0.65	0.33	0.12	0.39	0.34	0.33
Newspaper	.	.	.	.	.	.	.	0.06	0.08	0.04	0.46	0.13	0.06	0.27	0.87	0.31
Online Shop	.	.	.	.	.	.	.	.	0.04	0.06	0.15	0.22	0.07	0.59	0.06	0.12
Photos	.	.	.	.	.	.	.	.	.	0.03	0.11	0.13	0.03	0.16	0.08	0.03
Portal	.	.	.	.	.	.	.	.	.	.	0.03	0.26	0.27	0.24	0.45	0.10
Retail	.	.	.	.	.	.	.	.	.	.	.	0.03	0.11	0.20	0.19	0.14
Service	.	.	.	.	.	.	.	.	.	.	.	.	0.01	0.04	0.18	0.06
Social Network	.	.	.	.	.	.	.	.	.	.	.	.	.	0.02	0.09	0.36
Sports	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0.03	0.06
Travel	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0.14

Table 4: Overview of viewership correlation within and across the sixteen website categories in the January 2011 data set, used in the NCL case study. The diagonal elements represent the mean absolute correlation in that particular website category, while the off-diagonal elements represent the maximum absolute correlation between each pair of groups.

low average correlations; the highest comes from the Travel category (0.14), which we expect due to users doing travel website price comparisons.

The off-diagonal elements of Table 4 show the maximum absolute correlation between each pair of groups. This is calculated by taking the maximum correlation between two websites from the respective groups. For example, in January 2011 there is a high correlation of 0.89 between an E-mail and Portal site. Likely this means users are accessing a particular portal site (for example, Yahoo) and also receiving e-mail there. In contrast, there is a low correlation between E-mail and Filesharing sites, only 0.04. Particular e-mail users are not also using the same Filesharing website.

C. Extension to Inequality Constraints

We can also consider the more general optimization problem given by (5). One might imagine that a reasonable approach would be to initialize with $\boldsymbol{\beta}$ such that

$$\mathbf{C}\boldsymbol{\beta} \leq \mathbf{b} \tag{12}$$

and then apply a coordinate descent algorithm subject to ensuring that at each update (12) is not violated. Unfortunately, this approach typically gets stuck at a constraint boundary point where no improvement is possible by changing a single coordinate. In this setting the criterion can often be improved by moving along the constraint boundary, but such a move requires adjusting multiple coefficients simultaneously which is not possible using coordinate descent, because it only updates one coefficient at a time.

Instead we introduce a set of m slack variables $\boldsymbol{\delta}$ so that (12) can be equivalently expressed as

$$\mathbf{C}\boldsymbol{\beta} + \boldsymbol{\delta} = \mathbf{b}, \boldsymbol{\delta} \geq \mathbf{0} \quad \text{or} \quad \tilde{\mathbf{C}}\tilde{\boldsymbol{\beta}} = \mathbf{b}, \boldsymbol{\delta} \geq \mathbf{0}, \tag{13}$$

where $\tilde{\boldsymbol{\beta}} = (\boldsymbol{\beta}, \boldsymbol{\delta})$ is a $p + m$-dimensional vector, $\tilde{\mathbf{C}} = [\mathbf{C} \;\; \mathbf{I}]$, and $\mathbf{I}$ is an m-dimensional identity matrix. Let $\mathbf{e}_{\boldsymbol{\delta}}(\mathbf{a})$ be a function which selects out the elements of $\mathbf{a}$ that correspond to $\boldsymbol{\delta}$. For example, $\mathbf{e}_{\boldsymbol{\delta}}(\tilde{\boldsymbol{\beta}}) = \boldsymbol{\delta}$ while $\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\tilde{\boldsymbol{\beta}}) = \boldsymbol{\beta}$. Then, the inclusion of the slack variables in (13) allows us to reexpress the criterion (5) as

$$\arg\min_{\tilde{\boldsymbol{\beta}}} \frac{1}{2}\|\boldsymbol{Y} - \tilde{\mathbf{X}}\tilde{\boldsymbol{\beta}}\|_2^2 + \lambda\|\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\tilde{\boldsymbol{\beta}})\|_1 \quad \text{such that} \quad \tilde{\mathbf{C}}\tilde{\boldsymbol{\beta}} = \mathbf{b}, \; \mathbf{e}_{\boldsymbol{\delta}}(\tilde{\boldsymbol{\beta}}) \geq \mathbf{0}, \tag{14}$$

where $\tilde{\mathbf{X}} = [\mathbf{X} \;\; \mathbf{0}]$, and $\mathbf{0}$ is a n by m matrix of zero elements. The criterion in (14) is very similar to the equality PAC, (5). The only differences are that the components of $\tilde{\boldsymbol{\beta}}$ corresponding to $\boldsymbol{\delta}$ do not appear in the penalty term and are required to be non-negative.

Even with these minor differences, the same basic approach from Section 4 can still be adopted for fitting (14). In particular Lemma 4 provides a set of conditions under which (5) can be solved.

Lemma 4. *For a given index set $\mathcal{A}$ and vector $\mathbf{s}$ such that $\mathbf{e}_{\boldsymbol{\delta}}(\mathbf{s}) = 0$, define $\tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}},\mathbf{s}}$ by:*

$$\tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}},\mathbf{s}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{2}\|\tilde{\boldsymbol{Y}} - \mathbf{X}^*\boldsymbol{\theta}\|_2^2 + \lambda\|\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\boldsymbol{\theta})\|_1 \;\; \textit{such that} \;\; \mathbf{e}_{\boldsymbol{\delta}}(\boldsymbol{\theta}) \geq 0, \tag{15}$$

where $\tilde{\boldsymbol{Y}} = \boldsymbol{Y} - \tilde{\mathbf{X}}_{\mathcal{A}}\tilde{\mathcal{C}}_{\mathcal{A}}^{-1}\mathbf{b} + \lambda\mathbf{X}^{-}\left(\tilde{\mathbf{C}}_{\mathcal{A}}^{-1}\tilde{\mathbf{C}}_{\bar{\mathcal{A}}}\right)^T\mathbf{s}$, $\mathbf{X}^{-}$ is a matrix such that $\mathbf{X}^{*T}\mathbf{X}^{-} = \mathbf{I}$ and $\mathbf{X}^* = \tilde{\mathbf{X}}_{\bar{\mathcal{A}}} - \tilde{\mathbf{X}}_{\mathcal{A}}\tilde{\mathcal{C}}_{\mathcal{A}}^{-1}\tilde{\mathcal{C}}_{\mathcal{A}}$. Suppose

$$\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\mathbf{s}) \;\; = \;\; \textit{sign}\left(\mathbf{e}_{\bar{\boldsymbol{\delta}}}\left(\tilde{\boldsymbol{\beta}}_{\mathcal{A},\mathbf{s}}\right)\right), \tag{16}$$

$$\mathbf{e}_{\boldsymbol{\delta}}\left(\tilde{\boldsymbol{\beta}}_{\mathcal{A},\mathbf{s}}\right) \;\; \geq \;\; 0, \tag{17}$$

where $\tilde{\boldsymbol{\beta}}_{\mathcal{A},\mathbf{s}} = \tilde{\mathbf{C}}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \tilde{\mathbf{C}}_{\bar{\mathcal{A}}}\tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}},\mathbf{s}}\right)$. Then, the solution to the PAC criterion (5) is given by $\mathbf{e}_{\bar{\delta}}(\tilde{\boldsymbol{\beta}})$ where,

$$\tilde{\boldsymbol{\beta}} = \begin{bmatrix} \tilde{\boldsymbol{\beta}}_{\mathcal{A},\mathbf{s}} \\ \tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}},\mathbf{s}} \end{bmatrix}.$$

The proof of this result is similar to that for Lemma 3, so we omit it here. Lemma 4 shows that, provided an appropriate $\mathcal{A}$ and $\mathbf{s}$ are chosen, the solution to the PAC can still be computed by solving a lasso type criterion, (15). However, we must now ensure that both (16) and (17) hold. Condition 16 is equivalent to (8) in the equality constraint setting, while (17) along with the constraint in (15) ensure that $\boldsymbol{\delta} \geq \mathbf{0}$. We use the same strategy as in Section 4. First, obtain an initial coefficient estimate, $\tilde{\boldsymbol{\beta}}_0$. Next, select $\mathcal{A}$ corresponding to the largest m elements of $|\tilde{\boldsymbol{\beta}}_0|$, say $\tilde{\boldsymbol{\beta}}_{\mathcal{A}}$. Finally, the m-dimensional $\mathbf{s}$ vector is chosen by fixing $\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\mathbf{s}) = \text{sign}(\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\tilde{\boldsymbol{\beta}}_{\mathcal{A}}))$ and setting the remaining elements of $\mathbf{s}$ to zero, i.e. $\mathbf{e}_{\boldsymbol{\delta}}(\mathbf{s}) = 0$. $\tilde{\boldsymbol{\beta}}_{\mathcal{A}}$ is our initial guess for $\tilde{\boldsymbol{\beta}}_{\mathcal{A},\mathbf{s}}$ so as long as $\tilde{\boldsymbol{\beta}}_{\mathcal{A}}$ is sign consistent for $\tilde{\boldsymbol{\beta}}_{\mathcal{A},\mathbf{s}}$ then (16) will hold. Similarly, by only including the largest current values of $\boldsymbol{\delta}$ in $\mathcal{A}$, for a small enough step in λ, none of these elements will become negative, and (17) will hold.

Algorithm 2 PAC with Inequality Constraints

1. Initialize $\tilde{\boldsymbol{\beta}}_0$ by solving (14) using $\lambda_0 = \lambda_{\max}$.

2. At step k select $\mathcal{A}_k$ and $\mathbf{s}_k$ using the largest m elements of $|\tilde{\boldsymbol{\beta}}_{k-1}|$ and set

$$\lambda_k \leftarrow 10^{-\alpha}\lambda_{k-1}.$$

3. Compute $\tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}}_k,\mathbf{s}_k}$ by solving (15). Let $\tilde{\boldsymbol{\beta}}_{\mathcal{A}_k,\mathbf{s}_k} = \tilde{\mathbf{C}}_{\mathcal{A}_k}^{-1}\left(\mathbf{b} - \tilde{\mathbf{C}}_{\bar{\mathcal{A}}_k}\tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}}_k,\mathbf{s}_k}\right)$.

4. If (16) and (17) hold then set $\tilde{\boldsymbol{\beta}}_k = \begin{bmatrix}\tilde{\boldsymbol{\beta}}_{\mathcal{A}_k,\mathbf{s}_k} \\ \tilde{\boldsymbol{\beta}}_{\bar{\mathcal{A}}_k,\mathbf{s}_k}\end{bmatrix}$, $k \leftarrow k+1$ and return to 2.

5. If (16) or (17) do not hold then the step size must be too large. Hence, set

$$\lambda_k \leftarrow \lambda_{k-1} - \frac{1}{2}(\lambda_{k-1} - \lambda_k)$$

   and return to 3.

6. Iterate until $\lambda_k < \lambda_{\min}$.

Hence, Algorithm 2 can be used to fit the inequality PAC criterion. Notice that Algorithm 2 only involves slight changes to Algorithm 1. In particular solving (15) in Step 3 poses little additional complication over fitting the standard lasso criterion. The only differences are that the elements of $\boldsymbol{\theta}$ that correspond to $\boldsymbol{\delta}$, i.e. $\mathbf{e}_{\boldsymbol{\delta}}(\boldsymbol{\theta})$, have zero penalty and must be non-negative. However, these changes are simple to incorporate into a coordinate descent algorithm. For any θ_j that is an element of $\mathbf{e}_{\boldsymbol{\delta}}(\boldsymbol{\theta})$, we first compute the unshrunk least squares estimate, $\hat{\theta}_j$, and then set

$$\tilde{\theta}_j = \left[\hat{\theta}_j\right]_+. \tag{18}$$

It is not hard to show that (18) enforces the non-negative constraint on $\boldsymbol{\delta}$ while also ensuring that no penalty term is applied to the slack variables. The update step for the original $\boldsymbol{\beta}$ coefficients, $\mathbf{e}_{\bar{\boldsymbol{\delta}}}(\boldsymbol{\theta})$ (those that do not involve $\boldsymbol{\delta}$), is identical to that for the standard lasso. The initial solution, $\tilde{\boldsymbol{\beta}}_0$, can still be computed by solving a standard linear programming problem, subject to inequality constraints.

D. Algorithm Implementation Details

Implementing the PAC lasso algorithm requires making a choice for $\mathbf{X}^-$, which is generally not difficult. If $p \leq n + m$, then it is easy to see that $\mathbf{X}^- = \mathbf{U}\mathbf{D}^{-1}\mathbf{V}^T$ satisfies $\mathbf{X}^{*T}\mathbf{X}^- = \mathbf{I}$ where $\mathbf{X}^* = \mathbf{U}\mathbf{D}\mathbf{V}^T$ represents the singular value decomposition of $\mathbf{X}^*$. If $p > n+m$, then we use the fact that in general for Lemma 3 to hold, we only require $\mathbf{X}^-$ to be chosen such that

$$\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}} = \mathbf{X}^{-T}\mathbf{X}^*\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}, \tag{19}$$

where $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$ is the solution to (7). But standard properties of the lasso tell us that $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$ can have at most n non-zero components. Hence, (19) will hold if we choose $\mathbf{X}^-$ to be the inverse of the columns of $\mathbf{X}^*$ corresponding to the (at most) n non-zero elements of $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$. Of course we do not know a priori with complete certainty which elements of $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$ will be non-zero. However, based on the solution to the previous step in the algorithm, it is easy to compute the elements that are furthest from becoming non-zero, and these can generally be safely ignored in computing $\mathbf{X}^-$. On the rare occasions where an incorrect set of columns is selected, we simply reduce the step size in λ.

As mentioned in Section 4, one can generally initialize the algorithm using the solution to (9). However, this approach could potentially fail if one of the constraints in $\mathbf{C}$ is parallel with $\|\boldsymbol{\beta}\|_1$, for example $\sum \beta_j = 1$, in which case there may not be a unique solution to (9). In this setting we use quadratic programming to initialize the algorithm, which is slightly less efficient but does not unduly impact the computational burden, because the solution only needs to be found for a single value of λ.

E. Proofs of Lemmas 2 and 3

Consider any index set $\mathcal{A}$ such that $\mathbf{C}_{\mathcal{A}}$ is non-singular. The constraint $\mathbf{C}\boldsymbol{\beta} = \mathbf{b}$ can be written as

$$\mathbf{C}_{\mathcal{A}}\boldsymbol{\beta}_{\mathcal{A}} + \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}} = \mathbf{b} \quad \Leftrightarrow \quad \boldsymbol{\beta}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right) \ ,$$

and thus, we can determine $\boldsymbol{\beta}_{\mathcal{A}}$ from $\boldsymbol{\beta}_{\bar{\mathcal{A}}}$. Then, for any $\boldsymbol{\beta}$ such that $\mathbf{C}\boldsymbol{\beta} = \mathbf{b}$,

$$\begin{aligned}
&\frac{1}{2}\left\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}\right\|_1 \\
&= \frac{1}{2}\left\|\boldsymbol{Y} - \mathbf{X}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}} - \mathbf{X}_{\mathcal{A}}\boldsymbol{\beta}_{\mathcal{A}}\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right\|_1 + \lambda\left\|\boldsymbol{\beta}_{\mathcal{A}}\right\|_1 \\
&= \frac{1}{2}\left\|\boldsymbol{Y} - \mathbf{X}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}} - \mathbf{X}_{\mathcal{A}}\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right)\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right\|_1 + \lambda\left\|\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right)\right\|_1 \\
&= \frac{1}{2}\left\|\left[\boldsymbol{Y} - \mathbf{X}_{\mathcal{A}}\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{b}\right] - \left(\mathbf{X}_{\bar{\mathcal{A}}} - \mathbf{X}_{\mathcal{A}}\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right\|_1 + \lambda\left\|\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right)\right\|_1 .
\end{aligned}$$

By using the change of variable $\boldsymbol{\theta} = \boldsymbol{\beta}_{\bar{\mathcal{A}}}$, the PAC problem is equivalent to the following unconstrained optimization problem:

$$\min_{\boldsymbol{\theta}} \frac{1}{2}\left\|\boldsymbol{Y}^* - \mathbf{X}^*\boldsymbol{\theta}\right\|_2^2 + \lambda\left\|\boldsymbol{\theta}\right\|_1 + \lambda\left\|\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}\right)\right\|_1 ,$$

and let $\boldsymbol{\theta}_{\bar{\mathcal{A}}}$ denote a solution to the above optimization problem. Then, a solution to the original PAC problem is given

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_{\mathcal{A}} \\ \boldsymbol{\beta}_{\bar{\mathcal{A}}} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}_{\bar{\mathcal{A}}}\right) \\ \boldsymbol{\theta}_{\bar{\mathcal{A}}} \end{bmatrix} ,$$

and this completes Lemma 2.

To prove Lemma 3, consider an arbitrary $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$ and $\mathbf{s}$ such that $\mathbf{s} = \operatorname{sign}\left(\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}\right)\right)$. Let $F : \mathbb{R}^{p-m} \to \mathbb{R}_+$ denote the objective function of the optimization problem in Equation (6); that is, for each $\boldsymbol{\theta}$,

$$F(\boldsymbol{\theta}) = \frac{1}{2}\left\|\boldsymbol{Y}^* - \mathbf{X}^*\boldsymbol{\theta}\right\|_2^2 + \lambda\left\|\boldsymbol{\theta}\right\|_1 + \lambda\left\|\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}\right)\right\|_1 .$$

By definition of $\boldsymbol{\beta}_{\bar{\mathcal{A}}}$, we have $F(\boldsymbol{\beta}_{\bar{\mathcal{A}}}) \leq F(\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}})$. To complete the proof, it suffices to show that $F(\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}) \leq F(\boldsymbol{\beta}_{\bar{\mathcal{A}}})$. Suppose, on the contrary, that $F(\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}) > F(\boldsymbol{\beta}_{\bar{\mathcal{A}}})$. For each $\alpha \in [0,1]$, let $\boldsymbol{\theta}_\alpha \in \mathbb{R}^{p-m}$ and $g(\alpha) \in \mathbb{R}_+$ be defined by:

$$\boldsymbol{\theta}_\alpha \equiv (1-\alpha)\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}} + \alpha\boldsymbol{\beta}_{\bar{\mathcal{A}}} \qquad \text{and} \qquad g(\alpha) \equiv F(\boldsymbol{\theta}_\alpha) = F\left((1-\alpha)\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}} + \alpha\boldsymbol{\beta}_{\bar{\mathcal{A}}}\right) .$$

Note that g is a convex function on $[0,1]$ because $F(\cdot)$ is convex. Moreover, we have $g(0) = F(\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}) > F(\boldsymbol{\beta}_{\bar{\mathcal{A}}}) = g(1)$. Thus, for all $0 < \alpha \leq 1$,

$$g(\alpha) = g(\alpha \cdot 1 + (1-\alpha)\cdot 0) \leq \alpha g(1) + (1-\alpha)g(0) < g(0) .$$

By our hypothesis, $|s_i| = 1$ for all i, and thus, every coordinate of the vector $\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}\right)$ is bounded away from zero. So, we can choose α_0 sufficiently small so that

$$\text{sign}\left(\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}_{\alpha_0}\right)\right) = \text{sign}\left(\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}\right)\right) \quad .$$

Then, it follows that

$$\begin{aligned}
F(\boldsymbol{\theta}_{\alpha_0}) &= \frac{1}{2}\left\|\boldsymbol{Y}^* - \mathbf{X}^*\boldsymbol{\theta}_{\alpha_0}\right\|_2^2 + \lambda\left\|\boldsymbol{\theta}_{\alpha_0}\right\|_1 + \lambda\mathbf{s}^T\mathbf{C}_{\mathcal{A}}^{-1}\left(\mathbf{b} - \mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}_{\alpha_0}\right) \\
&= \frac{(\boldsymbol{Y}^*)^T\boldsymbol{Y}^*}{2} - (\boldsymbol{Y}^*)^T\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0} - \lambda\mathbf{s}^T\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\boldsymbol{\theta}_{\alpha_0} + \frac{(\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0})^T\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0}}{2} + \lambda\left\|\boldsymbol{\theta}_{\alpha_0}\right\|_1 + \lambda\mathbf{s}^T\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{b} \\
&= \frac{(\boldsymbol{Y}^*)^T\boldsymbol{Y}^*}{2} - (\boldsymbol{Y}^*)^T\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0} - \left(\lambda\mathbf{X}^-\left(\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)^T\mathbf{s}\right)^T\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0} \\
&\qquad + \frac{(\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0})^T\mathbf{X}^*\boldsymbol{\theta}_{\alpha_0}}{2} + \lambda\left\|\boldsymbol{\theta}_{\alpha_0}\right\|_1 + \lambda\mathbf{s}^T\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{b} \\
&= \frac{1}{2}\left\|\tilde{\boldsymbol{Y}} - \mathbf{X}^*\boldsymbol{\theta}_{\alpha_0}\right\|_2^2 + \lambda\left\|\boldsymbol{\theta}_{\alpha_0}\right\|_1 + \mathbf{d}
\end{aligned}$$

where the last equality follows from $\tilde{\boldsymbol{Y}} = \boldsymbol{Y}^* + \lambda\mathbf{X}^-\left(\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)^T\mathbf{s}$, and $\mathbf{d}$ is defined by

$$\mathbf{d} = -\lambda(\boldsymbol{Y}^*)^T\mathbf{X}^-\left(\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)^T\mathbf{s} \;-\; \frac{\left[\lambda\mathbf{X}^-\left(\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)^T\mathbf{s}\right]^T\left[\lambda\mathbf{X}^-\left(\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{C}_{\bar{\mathcal{A}}}\right)^T\mathbf{s}\right]}{2} \;+\; \lambda\mathbf{s}^T\mathbf{C}_{\mathcal{A}}^{-1}\mathbf{b} \;\; .$$

Also, the third equality follows from the fact that $(\mathbf{X}^-)^T\mathbf{X}^* = \mathbf{I}$.

It follows from the same argument that

$$F(\boldsymbol{\theta}_{\mathcal{A},\mathbf{s}}) = \frac{1}{2}\left\|\tilde{\boldsymbol{Y}} - \mathbf{X}^*\boldsymbol{\theta}_{\mathcal{A},\mathbf{s}}\right\|_2^2 + \lambda\left\|\boldsymbol{\theta}_{\mathcal{A},\mathbf{s}}\right\|_1 + \mathbf{d}$$

Since $g(\alpha) < g(0)$, we have that $F(\boldsymbol{\theta}_{\alpha_0}) < F(\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}})$, and this implies that

$$\frac{1}{2}\left\|\tilde{\boldsymbol{Y}} - \mathbf{X}^*\boldsymbol{\theta}_{\alpha_0}\right\|_2^2 + \lambda\left\|\boldsymbol{\theta}_{\alpha_0}\right\|_1 \quad < \quad \frac{1}{2}\left\|\tilde{\boldsymbol{Y}} - \mathbf{X}^*\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}\right\|_2^2 + \lambda\left\|\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}\right\|_1 \;\; ,$$

but this contradicts the optimality of $\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}$! Therefore, it must be the case that $F(\boldsymbol{\beta}_{\bar{\mathcal{A}},\mathbf{s}}) \leq F(\boldsymbol{\beta}_{\bar{\mathcal{A}}})$, which completes the proof.

F. Simulation Studies: PAC Comparison to Binomial Methods

In this section, we present further simulation results to compare PAC's performance relative to unconstrained binomial fits. Here we consider the setting corresponding to data generated from a binomial logistic regression model with $g(\boldsymbol{\beta})$ equal to the corresponding loglikelihood. The setup for the binomial simulation results is very similar to the procedure followed in

Section 5.1. We again consider six simulation settings, three different combinations of n observations and p predictors, and two different correlation structures. The training data sets and constraints are produced in an identical fashion to that in Section 5.1 except that the response is Bernoulli with a logistic link. However, instead of using RMSE for our error computations, the error metric is the percentage of incorrect binomial predictions. This process is repeated 100 times for each of the six settings.

The test error values for the six resulting settings are displayed in Table 5. GLM versions of the four comparison methods from Table 2 are included, along with the Bayes error rate for comparison. For the low-dimensional, traditional setting ($m = 5$, $n = 100, p = 50$), PAC shows a moderate improvement over the standard logistic regression fit. As in the lasso case presented in Section 5.1, one might expect this result given this is a relatively low-dimensional problem with only a small number of constraints. Both relaxed methods display lower error rates than their unrelaxed counterparts, and the higher correlations in the $\rho = 0.5^{|i-j|}$ design structure do not change the relative rankings of the four approaches. For the second, more complex situation with $n = 1000, p = 500$, and $m = 10$,[7] the low ratio of m relative to p results in PAC showing only small improvements over its unconstrained counterparts. Again, however, our main purpose in examining this setting was to prove that the PAC algorithm is still efficient enough to optimize the constrained criterion even for large data sets and very high-dimensional data.

The final setting examined data with $n = 50, p = 100$ and a larger number of constraints, $m = 30$. This setting is more statistically favorable for PAC, because it has the potential to produce significantly more accurate regression coefficients by correctly incorporating the larger number of constraints. However, this is also a computationally difficult setting for PAC, because a large value of m causes the coefficient paths to be highly variable. Nevertheless, the large improvements in accuracy for both PAC and relaxed PAC demonstrate that our algorithm is quite capable of dealing with this added complexity.

[7]We use a larger value for n in the binomial setting because these distributions provide less information for estimating the regression coefficients.

	ρ	Bayes	GLM	PAC	Relaxed GLM	PAC Relaxed
$n = 100, p = 50$	0	12.27(0.11)	19.36(0.23)	18.56(0.24)	19.33(0.45)	17.68(0.31)
$m = 5$	$0.5^{\|i-j\|}$	9.30(0.10)	14.60(0.18)	14.08(0.19)	14.51(0.20)	13.34(0.25)
$n = 1000, p = 500$	0	11.02(0.19)	12.33(0.22)	12.00(0.26)	12.14(0.26)	11.68(0.28)
$m = 10$	$0.5^{\|i-j\|}$	8.60(0.19)	10.15(0.28)	9.76(0.31)	10.01(0.33)	9.44(0.27)
$n = 50, p = 100$	0	8.20(0.06)	43.17(1.07)	36.06(0.85)	41.60(1.01)	31.23(0.63)
$m = 30$	$0.5^{\|i-j\|}$	7.26(0.10)	37.73(1.58)	28.03(0.78)	35.67(1.47)	24.54(0.70)

Table 5: Average misspecification error (in percentages) over 100 training data sets for four binomial methods tested in three different simulation settings and two different correlation structures. The Bayes error rate is given for comparison; it represents the minimum error rate. The numbers in parentheses are standard errors, also in percentages.

References

Ambroziak, A. (2015). Cruise lines increasingly enticing solo travelers aboard.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2010). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122.

Clements, M. (2013). Solo travelers find a berth with norwegian cruise lines.

CruiseCritic (2017). The Truth About Solo Cruise Cabins. URL: https://www.cruisecritic.com/articles.cfm?ID=1989 Accessed 22 January 2018.

Danaher, P. (2007). Modeling page views across multiple websites with an application to internet reach and frequency prediction. *Marketing Science*, 26(3):422–437.

Dave, K. and Varma, V. (2010). Learning the click-through rate for rare/new ads from similar ads. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 897–898. ACM.

de Leeuw, J. (1994). *Block-relaxation Algorithms in Statistics*. Springer Berlin / Heidelberg.

Efron, B., Hastie, T., Johnston, I., and Tibshirani, R. (2004). Least angle regression (with discussion). *The Annals of Statistics*, 32(2):407–451.

eMarketer (2012). Digital Ad Spending Tops 37 Billion. URL: http://www.emarketer.com/newsroom/index.php/digital-ad-spending-top-37-billion-2012-market-consolidates. Accessed 4 Jun 2015.

Fan, J., Zhang, J., and Yu, K. (2012). Vast portfolio selection with gross-exposure constraints. *Journal of the American Statistical Association*, 107:592–606.

Floudas, C. and Visweswaran, V. (1995). Quadratic optimization. *Nonconvex Optimization and Its Applications*, 2:217–269.

Frank, M. and Wolfe, P. (1956). An algorithm for quadratic programming. *Naval Research Logistics Quarterly 3*, pages 95–110.

Friedman, J., Hastie, T., Hoefling, H., and Tibshirani, R. (2007). Pathwise coordinate optimization. *Annals of Applied Statistics*, 1:302–332.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):302–332.

Gaines, B., Kim, J., and Zhou, H. (2018). Algorithms for fitting the constrained lasso. *Journal of Computational and Graphical Statistics*. (forthcoming).

HBR (2015). Is Programmatic Advertising the Future of Marketing? URL: https://hbr.org/2015/06/is-programmatic-advertising-the-future-of-marketing Accessed 2 November 2016.

He, T. (2011). Lasso and general l1-regularized regression under linear equality and inequality constraints.

Immorlica, N., Jain, K., Mahdian, M., and Talwar, K. (2005). Click fraud resistant methods for learning click-through rates. *Internet and Network Economics*, pages 34–45.

James, G. M., Radchenko, P., and Lv, J. (2009a). DASSO: Connections between the Dantzig selector and lasso. *Journal of the Royal Statistical Society, Series B*, 71:127–142.

James, G. M., Wang, J., and Zhu, J. (2009b). Functional linear regression that's interpretable. *Annals of Statistics*, 37:2083–2108.

Lange, K. (2012). *Optimization*. New York: Springer, 2 edition.

Lange, K., Chu, E., and Zhou, H. (2014). A brief survey of modern optimization for statisticians. *International Statistical Review*, 82(1):46–70.

Liaukonyte, J., Teixeira, T., and Wilbur, K. (2015). Television advertising and online shopping. *Marketing Science*, 34(3):311–330.

Lipsman, A. (2010). The New York Times Ranks as Top Online Newspaper According to May 2010 U.S. comScore Media Metrix data. Technical report, comScore, Inc.

Markowitz, H. M. (1952). Portfolio selection. *Journal of Finance*, 7(1):77–91.

Markowitz, H. M. (1959). *Portfolio Selection: Efficient Diversification of Investments*. New York: Wiley.

Montgomery, A. L., Li, S., Srinivasan, K., and Liechty, J. (2004). Modeling online browsing and path analysis using clickstream data. *Marketing Science*, 23(4):579–595.

Park, Y. and Fader, P. (2004). Modeling browsing behavior at multiple websites. *Marketing Science*, 23(3):280–303.

Paulson, C., Luo, L., and James, G. (2018). Efficient large-scale internet media selection optimization for online display advertising. *Journal of Marketing Research (to appear)*.

Post, T. W. (2017). How solo travelers can beat the high cost of going it alone. https://www.washingtonpost.com/lifestyle/travel/how-solo-travelers-can-beat-the-high-cost-of-going-it-alone Accessed 22 January 2018.

Satchell, A. (2011). Norwegian: Cruise fares to increase up to 10 percent April 1. *South Florida Sun-Sentinel*.

She, Y. (2010). Sparse regression with exact clustering. *Electronic Journal of Statistics*, 4:1055–1096.

Teo, C., Vishwanthan, S. V. N., Smola, A. J., and Le, Q. V. (2010). Bundle methods for regularized risk minimization. *Journal of Machine Learning Research*, 11:311–365.

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

Tibshirani, R., Saunders, M., Rosset, S., and Zhu, J. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society, Series B*, 67(1):91–108.

Tibshirani, R. and Taylor, J. (2011). The solution path of the generalized lasso. *Annals of Statistics*, 39(3):1335–1371.

Wu, T. and Lange, K. (2008). Coordinate descent algorithms for lasso penalized regression. *Annals of Applied Statistics*, 2(1):224–244.

Xu, Y. and Yin, W. (2013). A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal of Imaging Sciences*, 6(3):1758–1789.