Journal of the
Royal Statistical Society

SERIES B
Statistical
Methodology

J. R. Statist. Soc. B (2009)
71, *Part* 1, *pp.* 127–142

DASSO: connections between the Dantzig selector and lasso

Gareth M. James, Peter Radchenko and Jinchi Lv

University of Southern California, Los Angeles, USA

[Received September 2007. Revised April 2008]

Summary. We propose a new algorithm, DASSO, for fitting the entire coefficient path of the Dantzig selector with a similar computational cost to the least angle regression algorithm that is used to compute the lasso. DASSO efficiently constructs a piecewise linear path through a sequential simplex-like algorithm, which is remarkably similar to the least angle regression algorithm. Comparison of the two algorithms sheds new light on the question of how the lasso and Dantzig selector are related. In addition, we provide theoretical conditions on the design matrix X under which the lasso and Dantzig selector coefficient estimates will be identical for certain tuning parameters. As a consequence, in many instances, we can extend the powerful non-asymptotic bounds that have been developed for the Dantzig selector to the lasso. Finally, through empirical studies of simulated and real world data sets we show that in practice, when the bounds hold for the Dantzig selector, they almost always also hold for the lasso.

Keywords: Dantzig selector; DASSO; Lasso; Least angle regression

1. Introduction

Consider the standard linear regression model,

$$\mathbf{Y} = X\boldsymbol{\beta} + \varepsilon, \tag{1}$$

where $\mathbf{Y}$ is an n-vector of responses, $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p)$ is an $n \times p$ design matrix, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^{\mathrm{T}}$ is a p-vector of unknown regression coefficients and ε is an n-vector of random errors with standard deviation σ. An interesting problem in this setting is that of variable selection, i.e. determining which $\beta_j \neq 0$. Traditionally, it has generally been assumed that we are dealing with data where $p \ll n$ in which case several standard approaches, such as 'best subset selection', have been proposed. Recently, situations where p is of similar size to n, or even possibly significantly larger than n, have become increasingly common. Some important examples include functional regression where the predictor is itself a function, functional magnetic resonance imaging and tomography, gene expression studies where there are many genes and few observations, signal processing and curve smoothing. In such situations classical approaches, such as best subset selection, will generally fail because they are usually not computationally feasible for large p. As a result there has been much development of new variable selection methods that work with large values of p. A few examples include the non-negative garrotte (Breiman, 1995), the lasso (Tibshirani, 1996; Chen *et al.*, 1998; Donoho, 2006; Donoho *et al.*, 2006; Bickel *et al.*, 2008; Meinshausen *et al.*, 2007), the adaptive lasso (Zou, 2006), smoothly clipped absolute deviation (Fan and Li, 2001), the elastic net (Zou and Hastie, 2005), least angle regression (the LARS

Address for correspondence: Gareth M. James, Information and Operations Management Department, University of Southern California, 401R Bridge Hall, Los Angeles, CA 90089-0809, USA.
E-mail: gareth@marshall.usc.edu

1369–7412/09/71127

algorithm) (Efron *et al.*, 2004), adaptive model selection (Shen and Ye, 2002) and the Dantzig selector (Candes and Tao, 2007a, b; Bickel *et al.*, 2008; Meinshausen *et al.*, 2007).

The Dantzig selector has already received a considerable amount of attention. It was designed for linear regression models such as model (1) where p is large but the set of coefficients is sparse, i.e. most of the β_js are 0. The Dantzig selector estimate $\hat{\boldsymbol\beta}$ is defined as the solution to

$$\min(\|\tilde{\boldsymbol\beta}\|_1) \qquad \text{subject to } \|X^{\mathrm{T}}(\mathbf{Y}-X\tilde{\boldsymbol\beta})\|_\infty \leqslant \lambda_{\mathrm{D}}, \tag{2}$$

where $\|\cdot\|_1$ and $\|\cdot\|_\infty$ respectively represent the L_1- and L_∞-norms and λ_{D} is a tuning parameter. The L_1-minimization produces coefficient estimates that are exactly 0 in a similar fashion to the lasso and hence can be used as a variable selection tool. This approach has shown impressive empirical performance on real world problems involving large values of p. Candes and Tao (2007a) have provided strong theoretical justification for this performance by establishing sharp non-asymptotic bounds on the L_2-error in the estimated coefficients. They showed that the error is within a factor of $\log(p)$ of the error that would be achieved if the locations of the non-zero coefficients were known.

For any given λ_{D}, the Dantzig selector optimization criterion (2) can be solved fairly efficiently by using a standard linear programming procedure. However, producing the entire coefficient path currently requires solving problem (2) over a fine grid of values for λ_{D}. This grid approach can become computationally expensive when, for example, performing cross-validation on large data sets. By comparison, in their landmark paper Efron *et al.* (2004) introduced a highly efficient algorithm called LARS. They demonstrated that the coefficient path of the lasso is piecewise linear and, with a slight modification, LARS fits the entire lasso path. Note that when we refer to LARS in this paper we mean the modified version.

This paper makes two contributions. First, we prove that a LARS-type algorithm can be used to produce the entire coefficient path for the Dantzig selector. Our algorithm, which we call DASSO for 'Dantzig selector with sequential optimization', efficiently constructs a piecewise linear path through a sequential simplex-like algorithm that identifies the break points and solves the corresponding linear programs. DASSO can be thought of as a modification of LARS, because a small change in one of its steps produces the LARS algorithm. DASSO has a similar computational cost to that of LARS and produces considerable computational savings over a grid search. Second, we establish explicit conditions on the design matrix under which the Dantzig selector and the lasso will produce identical solutions. We demonstrate empirically that at the sparsest points of the coefficient path, where most coefficients are 0, these conditions usually hold.

As an example, consider the coefficient paths that are plotted in Fig. 1. These plots correspond to the diabetes data set that was used in Efron *et al.* (2004). The data contain 10 baseline predictors, age, sex, body mass, blood pressure and six blood serum measurements (S1,. . . ,S6), for $n = 442$ diabetes patients. The response is a measure of progression of disease 1 year after baseline. Fig. 1(a) contains the entire set of possible lasso coefficients computed by using the LARS algorithm. Fig. 1(b) illustrates the corresponding coefficients for the Dantzig selector computed by using DASSO. The left-hand sides of each plot represent the sparsest fits where most coefficients are 0, and the extreme right-hand sides reflect the ordinary least squares estimates. The left 40% of each plot are identical, with the first six variables entering the models in the same fashion. However, the fits differ somewhat in the middle portions where S1 enters next using the lasso whereas S2 enters using the Dantzig selector, before the two fits converge again near the ordinary least squares solution. This pattern of identical fits for the sparsest solutions occurs for every data set that we have examined. Other researchers have noted similar relationships (Efron *et al.*, 2007). In fact we provide general conditions on the design matrix

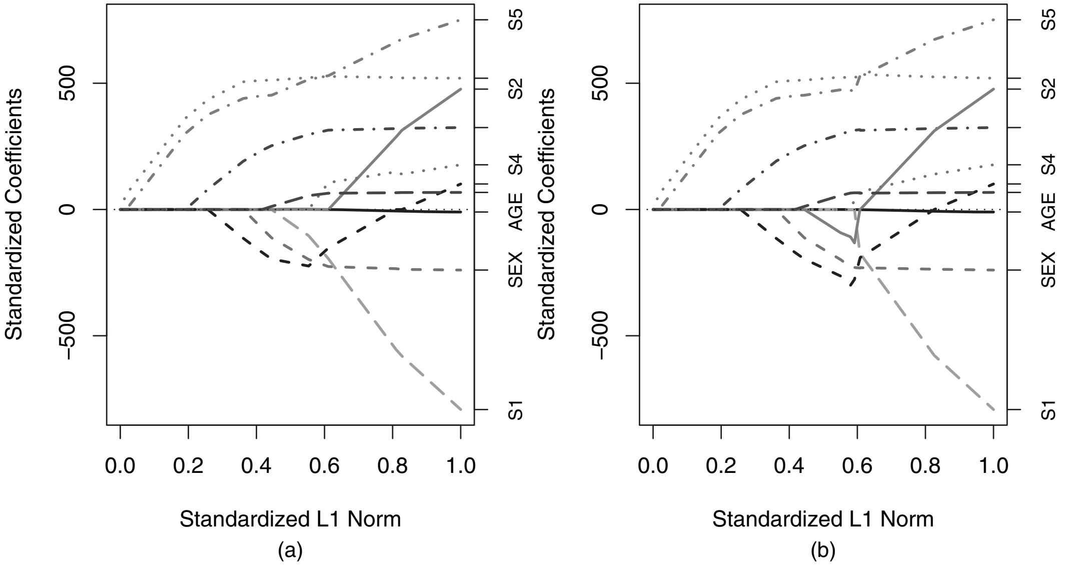

Fig. 1. (a) Lasso and (b) Dantzig selector coefficient paths on the diabetes data: the regions with standardized L_1-norms between 0 and 0.4 are identical

X, under which a given lasso solution will be identical to the corresponding Dantzig selector solution. This equivalence allows us to apply the impressive non-asymptotic bounds of Candes and Tao (2007a) to the lasso in addition to the Dantzig selector.

The paper is structured as follows. In Section 2 we present our DASSO algorithm and draw connections to LARS. Both algorithms are demonstrated on three real world data sets. Section 2 also provides the theoretical justification for DASSO. General conditions are derived in Section 3 under which the lasso and Dantzig selector solutions will be identical. In particular, we show that the strong similarities between the two approaches allow us, in many circumstances, to extend the Dantzig selector bounds to the lasso estimates. A detailed simulation study is provided in Section 4. Finally, we end with a discussion in Section 5. The proofs of all our results are provided in Appendix A.

2. DASSO algorithm

In Section 2.1 we present the key elements of our DASSO algorithm. We give the theoretical justification for this algorithm in Section 2.2. Using suitable location and scale transformations we can standardize the predictors and centre the response, so that

$$\sum_{i=1}^{n} Y_i = 0, \qquad \sum_{i=1}^{n} X_{ij} = 0, \qquad \sum_{i=1}^{n} X_{ij}^2 = 1, \qquad \text{for } j = 1, \ldots, p. \tag{3}$$

Throughout the paper we assume that conditions (3) hold. However, all numerical results are presented on the original scale of the data. For a given index set I we shall write X_I for the submatrix of X consisting of the columns that are indexed by I.

2.1. The algorithm

As with LARS, the DASSO algorithm progresses in piecewise linear steps. At the start of each step the variables that are most correlated with the residual vector are identified. These variables make up the current *active set* $\mathcal{A}$ and together with the current set of non-zero coefficients $\mathcal{B}$ determine the optimal direction that the coefficient path should move in. Along this direction, the maximal correlations are reduced towards 0 at the same rate. We then compute the exact distance that can be travelled before there is a change in either $\mathcal{A}$ or $\mathcal{B}$. At this point we adjust the two sets, and continue. Formally, the algorithm consists of the following steps (to simplify the notation, we omit superscript 'l' whenever possible).

Step 1: initialize β^1 as the zero vector in $\mathbb{R}^p$ and set $l=1$.
Step 2: let $\mathcal{B}$ be the set indexing the non-zero coefficients of β^l. Set $\mathbf{c}=X^{\mathrm{T}}(\mathbf{Y}-X\beta^l)$, let $\mathcal{A}$ be the active set indexing the covariances in $\mathbf{c}$ that are maximal in absolute value and let $\mathbf{s}_{\mathcal{A}}$ be the vector containing the signs of those covariances.
Step 3: identify either the index to be added to $\mathcal{B}$ or the index to be removed from $\mathcal{A}$. Use the new $\mathcal{A}$ and $\mathcal{B}$ to calculate the $|\mathcal{B}|$-dimensional direction vector $\mathbf{h}_{\mathcal{B}}=(X_{\mathcal{A}}^{\mathrm{T}}X_{\mathcal{B}})^{-1}\mathbf{s}_{\mathcal{A}}$. Let $\mathbf{h}$ be the p-dimensional vector with the components corresponding to $\mathcal{B}$ given by $\mathbf{h}_{\mathcal{B}}$ and the remainder set to 0.
Step 4: compute γ, the distance to travel in the direction $\mathbf{h}$ until a new covariance enters the active set or a coefficient path crosses zero. Set $\beta^{l+1}\leftarrow\beta^l+\gamma\mathbf{h}$ and $l\leftarrow l+1$.
Step 5: repeat steps 2–4 until $\|\mathbf{c}\|_\infty=0$.

The details for steps 3 ('direction') and 4 ('distance') are provided in Appendix A. As can be seen from the proof of theorem 1 in Section 2.2, the adjustments to $\mathcal{A}$ and $\mathcal{B}$ in step 3 exactly correspond to the changes in the active set and the set of non-zero coefficients of the estimate. Below we discuss the *one at a time* condition, under which $|\mathcal{A}|=|\mathcal{B}|+1$ after step 2 and, as a result, $|\mathcal{A}|=|\mathcal{B}|$ after step 3. The entire Dantzig selector coefficient path can be constructed by linearly interpolating $\beta^1,\beta^2,\ldots,\beta^L$, where L denotes the number of steps that the algorithm takes. Typically, L is of the same order of magnitude as $\min(n,p)$.

Note that both LARS and DASSO start with all the coefficients set to 0 and then add the variable that is most correlated with the residual vector. The only difference between the algorithms is in step 3. In LARS $\mathcal{A}$ and $\mathcal{B}$ are kept identical, but DASSO chooses $\mathcal{B}$ so that the direction $(X_{\mathcal{A}}^{\mathrm{T}}X_{\mathcal{B}})^{-1}\mathbf{s}_{\mathcal{A}}$ produces the greatest reduction in the maximal correlations per unit increase in $\|\hat{\beta}\|_1$. It turns out that $\mathcal{A}$ and $\mathcal{B}$ are often identical for DASSO as well, for at least the first few iterations. Hence, the sparsest sections of the lasso and Dantzig selector coefficient paths are indistinguishable.

To illustrate this point we compared the Dantzig selector and lasso coefficient paths on three real data sets. The first was the diabetes data that were discussed in Section 1. The second was a Boston housing data set with $p=13$ predictors and $n=506$ observations where the aim was to predict median house value. The final data set, on intelligence quotient IQ, considered the more difficult situation where $p>n$. The IQ data contained $n=74$ observations and $p=134$ predictors. The response was a measure of IQ for each individual and the predictors measured various characteristics of the brain as well as gender and race. The coefficient paths for each data set are plotted in Fig. 2 with full lines for the lasso and broken lines for the Dantzig selector. All three data sets have identical paths at the sparsest solutions, differ to varying degrees in the middle stages and then converge again at the end. These plots also show that the Dantzig selector paths are less smooth than those for the lasso.

2.2. Theoretical justification

We have performed numerous comparisons of the DASSO algorithm with the coefficient path that is produced by using the Dantzig selector applied to a fine grid of λ_{D}s. In all cases the

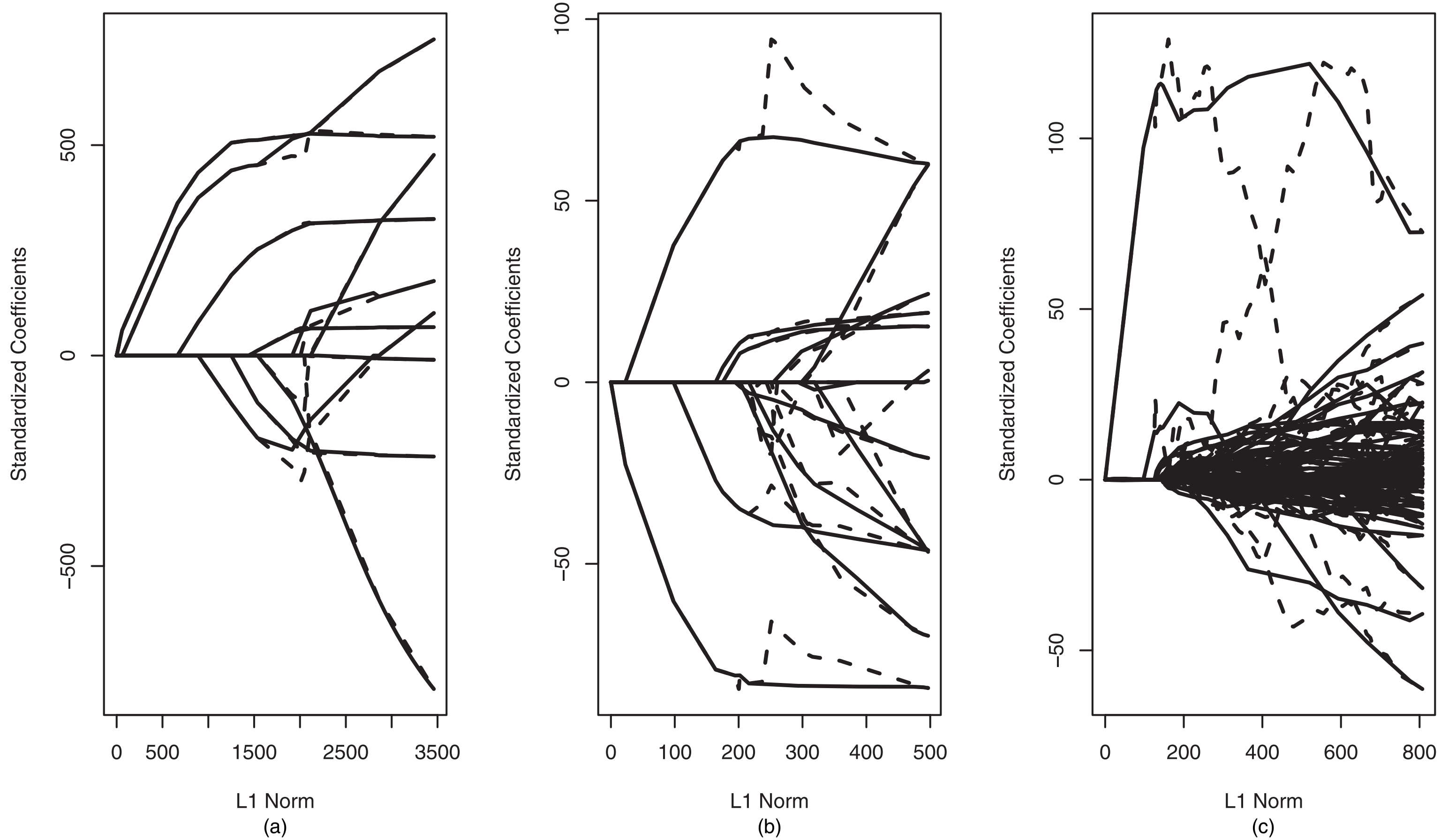

Fig. 2. Lasso (———) and Dantzig selector (– – –) coefficient paths on three different data sets: (a) diabetes data; (b) Boston housing data; (c) IQ data

results have been identical. Theorem 1 provides a rigorous theoretical justification for the accuracy of DASSO. As Efron *et al.* (2004) did for LARS, we assume the following one at a time condition: along the DASSO path the increases in the active set and the decreases in the set of non-zero coefficients happen one at a time. This situation is typical for quantitative data and can always be realized by adding a little jitter to the response values. Note that both LARS and DASSO can be generalized to cover other scenarios, e.g. the case of several coefficient paths crossing zero simultaneously. We choose to avoid this generalization, because it would provide little practical value and detract from the clarity of the presentation.

Theorem 1. If the one at a time condition holds, the DASSO algorithm produces a solution path for the Dantzig selector optimization problem (2), with β^1 corresponding to $\lambda_D = \|X^T Y\|_\infty$ and β^L corresponding to $\lambda_D = 0$.

Theorem 1 can be derived by an inductive argument. The base case is provided by the zero vector β^1, which solves optimization problem (2) for all λ_D at or above the level $\|X^T Y\|_\infty$. The induction step is established in Appendix A. It follows from the proof that $|\mathcal{A}| = |\mathcal{B}| = \text{rank}(X)$ at the end of the algorithm. Theorem 1 has an interesting connection with the regularized solution paths theory that was developed in Rosset and Zhu (2007). Their results imply that, if the Dantzig selector is viewed as the minimizer of the penalized criterion function $\|X^T(Y - X\tilde{\beta})\|_\infty + s\|\tilde{\beta}\|_1$, then the solution path $\hat{\beta}(s)$ is piecewise constant. We work with the definition of the Dantzig selector as the solution to the optimization problem (2), and we parameterize the coefficient paths by λ_D. Our theorem 1 shows existence of a continuous piecewise linear solution path $\hat{\beta}(\lambda_D)$ with $\lambda_D = \|X^T\{Y - X\,\hat{\beta}(\lambda_D)\}\|_\infty$.

3. The Dantzig selector *versus* the lasso

In this section we investigate the relationship between the Dantzig selector and the lasso. Section 3.1 gives general motivation for the similarities between the two approaches. In Section 3.2 formal conditions are provided that guarantee the equivalence of the Dantzig selector and the lasso. Section 3.3 then uses these conditions to extend the non-asymptotic Dantzig selector bounds to the lasso.

3.1. Connections

Assume that the data are generated from the linear regression model (1). The Dantzig selector estimate $\hat{\beta}_D = \hat{\beta}_D(\lambda_D)$ is defined as the solution to the Dantzig selector optimization problem (2). The lasso estimate $\hat{\beta}_L = \hat{\beta}_L(\lambda_L)$ is defined by

$$\hat{\beta}_L = \arg\min_{\tilde{\beta}}(\tfrac{1}{2}\|\mathbf{Y} - X\tilde{\beta}\|_2^2 + \lambda_L\|\tilde{\beta}\|_1), \tag{4}$$

where $\|\cdot\|_2$ denotes the L_2-norm and λ_L is a non-negative tuning parameter. We can define the lasso estimate equivalently as the solution to

$$\min(\|\tilde{\beta}\|_1) \qquad \text{subject to } \|\mathbf{Y} - X\tilde{\beta}\|_2^2 \leqslant s, \tag{5}$$

for some non-negative s.

The Dantzig selector and lasso share some similarities in view of problems (2) and (5). The only difference is that the Dantzig selector regularizes the sum of absolute coefficients $\|\tilde{\beta}\|_1$ with the L_∞-norm of the p-vector $X^T(\mathbf{Y} - X\tilde{\beta})$ whereas the lasso regularizes $\|\tilde{\beta}\|_1$ with the residual sum of squares. Furthermore, note that vector $-2X^T(\mathbf{Y} - X\tilde{\beta})$ is exactly the gradient of the residual sum of squares. As we show in Appendix A, when the tuning parameters λ_L and

λ_D are chosen to be the same, the lasso estimate is always a feasible solution to the Dantzig selector minimization problem, although it may not be an optimal solution. Therefore, when the corresponding solutions are not identical, the Dantzig selector solution is sparser, in terms of the L_1-norm, than the lasso solution.

These similarities can be made even more striking by noting that the solutions to problems (2) and (5) are respectively equal to those of

$$\min(\|\tilde{\beta}\|_1) \qquad \text{subject to } \|X^T X(\tilde{\beta}-\hat{\beta}_{ls})\|_\infty \leqslant t_D \tag{6}$$

and

$$\min(\|\tilde{\beta}\|_1) \qquad \text{subject to } \|X(\tilde{\beta}-\hat{\beta}_{ls})\|_2^2 \leqslant t_L \tag{7}$$

for some t_D and t_L, where $\hat{\beta}_{ls}$ is the least squares estimate. From equations (6) and (7) we see that the Dantzig selector minimizes the L_1-norm of $\tilde{\beta}$ subject to lying within a certain diamond that is centred at $\hat{\beta}_{ls}$, whereas the lasso solution lies within an ellipse with the same centre. Despite the different shapes, the solutions for the two methods will often be the same. Fig. 3 provides a graphical illustration in $p=2$ dimensions. We have generated six plots corresponding to correlations between the two columns of X ranging from -0.5 to 0.9. For each plot the broken ellipses represent the lasso constraint, whereas the full diamonds correspond to the Dantzig selector. The dotted line is the L_1-norm that is being minimized. In all six plots, despite the

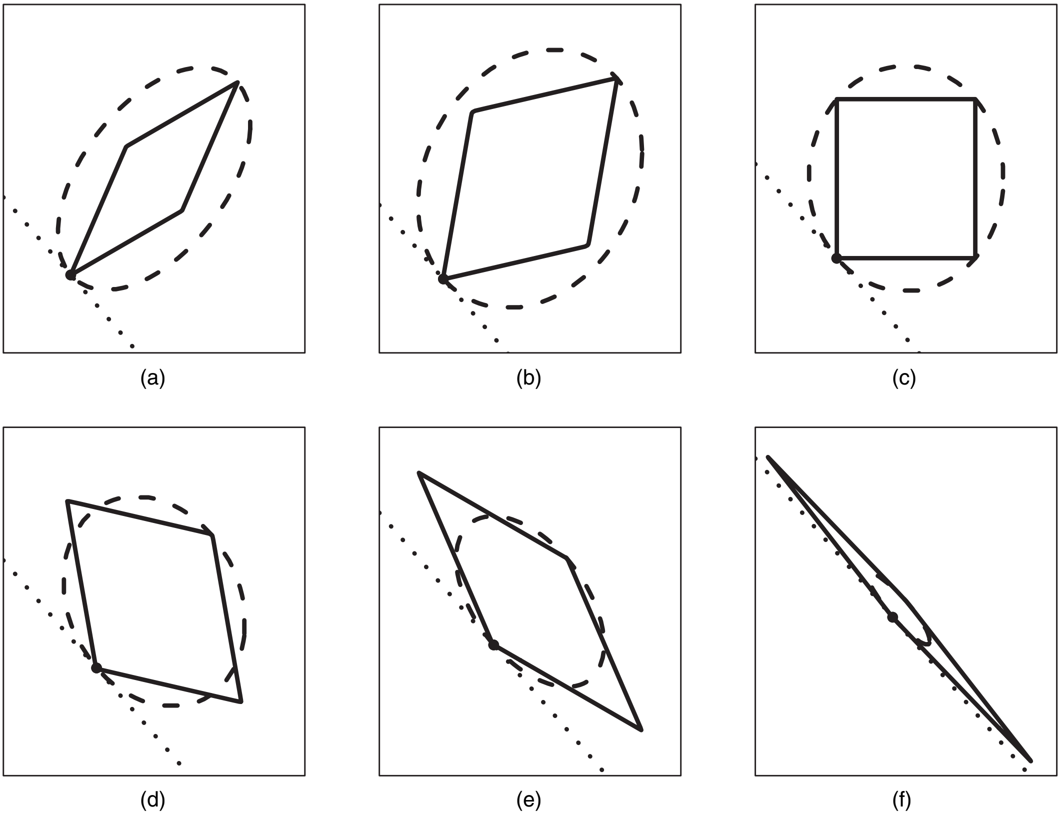

Fig. 3. Lasso (– – –) and Dantzig selector (◇) solutions in a $p=2$ dimensional space (·······, L_1-norm that is being minimized): (a) $\rho=-0.5$; (b) $\rho=-0.2$; (c) $\rho=0$; (d) $\rho=0.2$; (e) $\rho=0.5$; (f) $\rho=0.9$

differences in the shapes of the feasible regions, both methods give the same solution, which is represented by the larger dot on the L_1-line. We show in Section 3.2 that in $p=2$ dimensions the two methods will always be identical, whereas in higher dimensions this will not always be so. In fact, the situation where $p \geqslant 3$ can be different from the case where $p=2$, as demonstrated in, for example, Donoho and Tanner (2005), Plumbley (2007) and Meinshausen *et al.* (2007).

3.2. Conditions for equivalence

Define I_{L} as the set indexing the non-zero coefficients of $\hat{\beta}_{\mathrm{L}}$. Let X_{L} be the $n \times |I_{\mathrm{L}}|$ matrix that is constructed by taking $X_{I_{\mathrm{L}}}$ and multiplying its columns by the signs of the corresponding coefficients in $\hat{\beta}_{\mathrm{L}}$. The following result provides easily verifiable sufficient conditions for equality between the Dantzig selector and the lasso solutions.

Theorem 2. Assume that X_{L} has full rank $|I_{\mathrm{L}}|$ and $\lambda_{\mathrm{D}}=\lambda_{\mathrm{L}}$. Then $\hat{\beta}_{\mathrm{D}}=\hat{\beta}_{\mathrm{L}}$ if

$$\mathbf{u}=(X_{\mathrm{L}}^{\mathrm{T}}X_{\mathrm{L}})^{-1}\mathbf{1} \geqslant \mathbf{0} \quad \text{and} \quad \|X^{\mathrm{T}}X_{\mathrm{L}}\mathbf{u}\|_{\infty} \leqslant 1, \tag{8}$$

where $\mathbf{1}$ is an $|I_{\mathrm{L}}|$-vector of 1s and the vector inequality is understood componentwise.

In theorem 4 in Appendix A we provide a necessary and sufficient condition (14) for the equality of the two solutions, which is in general weaker than the sufficient condition (8). We placed this theorem in Appendix A because its condition is somewhat difficult to check in practice. The proof of theorem 2 is also given in Appendix A. It should be noted that theorem 2 provides conditions for the lasso and Dantzig solutions to be identical at a given point on the coefficient path. However, if condition (8) holds for all possible sets $|I_{\mathrm{L}}|$, or even just the sets that are generated by the LARS algorithm, then the entire lasso and Dantzig selector coefficient paths will be identical.

Next we list some special cases where condition (8) will hold at all points of the coefficient path.

Corollary 1. Assume that X is orthonormal, i.e. $X^{\mathrm{T}}X=I_p$. Then the entire lasso and Dantzig selector coefficient paths are identical.

Corollary 1 holds, because, if X is orthonormal, then $X_I^{\mathrm{T}}X_I=I_{|I|}$ for each index set I. Hence, condition (8) will hold at all points of the coefficient path. Corollary 1 can be extended to the case where the columns of X have non-negative correlation, provided that the correlation is common among all pairs of columns, which is shown to imply condition (8).

Theorem 3. Suppose that all pairwise correlations between the columns of X are equal to the same value ρ that lies in $[0,1)$. Then the entire lasso and Dantzig selector coefficient paths are identical. In addition, when $p=2$, the same holds for each ρ in $(-1,1)$.

When $p \geqslant 3$ the common correlation ρ cannot be negative because a matrix of this type is no longer positive definite and thus cannot be a covariance matrix. We stress that in $p \geqslant 3$ dimensions the equivalence of the Dantzig selector and lasso in general does not hold, and it holds if and only if condition (14) in Appendix A on the design matrix is satisfied.

We would like to point out that, after the paper was submitted, two recent references, Bickel *et al.* (2008) and Meinshausen *et al.* (2007), came to our attention. Bickel *et al.* (2008) established an approximate equivalence between the lasso and Dantzig selector and derived parallel oracle inequalities for the prediction risk in the general non-parametric regression model, as well as bounds on the L_q-estimation loss for $1 \leqslant q \leqslant 2$ in the linear model under assumptions that are different from ours. Meinshausen *et al.* (2007) gave a diagonal dominance condition of the

$p \times p$ matrix $(X^{\mathrm{T}}X)^{-1}$ that ensures the equivalence of the lasso and Dantzig selector. Clearly, this diagonal dominance condition implicitly assumes that $p \leqslant n$, whereas our conditions (8) and (14) have no such constraint.

3.3. Non-asymptotic bounds on the lasso error

A major contribution of Candes and Tao (2007a) is establishing oracle properties for the Dantzig selector under a so-called uniform uncertainty principle (Candes and Tao, 2005) on the design matrix X. They proved sharp non-asymptotic bounds on the L_2-error in the estimated coefficients and showed that the error is within a factor of $\log(p)$ of the error that would be achieved if the locations of the non-zero coefficients were known. These results assume that the tuning parameter is set to the *theoretical optimum* value of $\lambda_{\mathrm{D}} = \sigma\sqrt{\{2\log(p)\}}$.

Recently, there have also been several studies on the lasso with different foci. See, for example, Meinshausen and Bühlmann (2006), Zhang and Huang (2007) and Zhao and Yu (2006). Some of this work has developed bounds on the lasso error. However, most existing results in this area are asymptotic. In theorem 2 we identified conditions on the design matrix X under which the lasso estimate is the same as the Dantzig selector estimate. Under these conditions, or under a weaker general condition (14), the powerful theory that has already been developed for the Dantzig selector directly applies to the lasso.

4. Simulation study

We performed a series of four simulations to provide comparisons of performance between the Dantzig selector and lasso. For each set of simulations we generated 100 data sets and tested six different approaches. The first involved implementing the Dantzig selector by using the tuning parameter λ_{D} that was selected via Monte Carlo simulation as the maximum of $|X^{\mathrm{T}}\mathbf{Z}|$ over 20 realizations of $\mathbf{Z} \sim N(\mathbf{0}, \sigma^2 I)$ as suggested in Candes and Tao (2007a). The second approach used the lasso with the same tuning parameter as that suggested for the Dantzig selector. The next two methods, DS CV and LASSO CV, implemented the Dantzig selector and lasso in each case using the tuning parameter giving the lowest mean-squared error under tenfold cross-validation. The final two methods, double Dantzig and double lasso, use ideas that were proposed in James and Radchenko (2008) and Meinshausen (2007). First, the Dantzig selector and lasso are fitted by using the theoretical optimum. We then discard all parameters with zero coefficients and refit the Dantzig selector and lasso to the reduced number of predictors with the tuning parameters chosen by using cross-validation. This two-step approach reduces over-shrinkage of the coefficients. In all cases σ^2 was estimated by using the mean sum of squares from the DS CV and LASSO CV methods.

The results for the six approaches applied to the four simulations are summarized in Table 1. Five statistics are provided. The first,

$$\rho^2 = \frac{\sum_{j=1}^{p} (\hat{\boldsymbol{\beta}}_j - \boldsymbol{\beta}_j)^2}{\sum_{j=1}^{p} \min(\boldsymbol{\beta}_j^2, \sigma^2)},$$

represents the ratio of the squared error between the estimated and true $\boldsymbol{\beta}$-coefficients to the theoretical optimum squared error assuming that the identity of the non-zero coefficients was known. The second statistic, pred, provides the mean-squared error in predicting the expected

Table 1. Simulation results measuring accuracy of coefficient estimation ρ^2, prediction accuracy pred, number of predictors selected S, false positive rate F^+ and false negative rate F^- for six different methods†

Method	*Results for simulation 1*					*Results for simulation 2*				
	ρ^2	*pred*	S	F^+	F^-	ρ^2	*pred*	S	F^+	F^-
Dantzig selector	5.07	0.130	7.08	0.059	0.114	7.42	0.153	7.13	0.058	0.092
Lasso	5.07	0.130	7.08	0.059	0.114	7.36	0.152	7.11	0.057	0.090
DS CV	3.82	0.099	14.47	0.220	0.082	4.73	0.103	13.96	0.206	0.064
LASSO CV	3.81	0.098	13.76	0.204	0.082	4.65	0.101	13.78	0.202	0.064
Double Dantzig	2.24	0.060	4.65	0.010	0.160	2.85	0.066	5.08	0.016	0.128
Double lasso	2.24	0.060	4.65	0.010	0.160	2.83	0.066	5.08	0.016	0.128
	Results for simulation 3					*Results for simulation 4*				
Dantzig selector	6.77	0.141	6.88	0.057	0.138	23.01	2.345	8.86	0.026	0.260
Lasso	6.77	0.141	6.89	0.057	0.138	22.42	2.283	9.01	0.027	0.256
DS CV	4.57	0.100	15.16	0.236	0.094	9.42	0.969	28.04	0.122	0.142
LASSO CV	4.50	0.099	14.18	0.215	0.100	8.82	0.913	24.97	0.106	0.138
Double Dantzig	2.98	0.066	4.79	0.014	0.164	8.93	0.917	15.83	0.060	0.164
Double lasso	2.99	0.066	4.79	0.014	0.164	9.23	0.947	16.14	0.061	0.168

†Each simulation consisted of 100 different data sets.

response over a large test data set. The next value, S, provides the average number of variables selected in the final model. The final two statistics, F^+ and F^-, respectively represent the fraction of times that an unrelated predictor is included and the fraction of times that an important predictor is excluded from the model.

The first three sets of simulations used data that were generated with $n=200$ observations and $p=50$ predictors of which only five were related to the response with the true $\boldsymbol{\beta}$-coefficients generated from a standard normal distribution. The design matrices X in the first simulation were generated by using independent and identically distributed (IID) standard normal random variables and the error terms were similarly generated from a standard normal distribution. As a result of the IID entries, the correlations between columns of X were close to 0 (the standard deviations of the correlations was approximately 0.07). The results of Section 3 suggest that the lasso and Dantzig selector should give similar performance in this setting. This is exactly what we observe with the statistics for simulations with the Dantzig selector, DS CV and double Dantzig almost identical to their lasso counterparts. The Dantzig selector and lasso results are the worst in terms of coefficient estimation and prediction accuracy. However, both methods perform well at identifying the correct coefficients (F^+ and F^- are both low), suggesting overshrinkage of the true parameters. The cross-validation versions have lower error rates on the coefficient estimates and predictions but at the expense of including many irrelevant variables. The double Dantzig and double lasso methods provide the best performance, both in terms of the coefficients and prediction accuracy and in terms of a low F^+-rate with approximately the correct number of variables.

The second simulation used data sets with a similar structure to those in the first simulation. However, we introduced significant correlation structure into the design matrix by dividing the 50 predictors into five groups of 10 variables. Within each group we randomly generated predictors with common pairwise correlations between 0 and 0.5. Predictors from different groups were independent. As a result, the standard deviation of the pairwise correlations was twice that of the first simulation, which caused a deterioration in most of the statistics. However, all three versions of the Dantzig selector and lasso still gave extremely similar performance. Additionally, the double Dantzig and double lasso performed best followed by the cross-validated versions. The third simulation aimed to test the sensitivity of the approaches to the Gaussian error assumption by repeating the second simulation except that the errors were generated from a Student t-distribution with 3 degrees of freedom. The errors were normalized to have unit standard deviation. The Dantzig selector and lasso as well as the DS CV and LASSO CV methods actually improved slightly (though the differences were not statistically significant). The double Dantzig and double lasso both showed small deterioration but still outperformed the other approaches.

Finally, the fourth simulation tested out the large p, small n, scenario that the Dantzig selector was originally designed for. We produced design matrices with $p=200$ variables and $n=50$ observations, using IID standard normal entries. The error terms were also IID standard normal. As with the previous simulations, we used five non-zero coefficients, but their standard deviation was increased to 2. Not surprisingly, the performance of all methods deteriorated. This was especially noticeable for the Dantzig selector and lasso approaches. Athough these methods included roughly the right number of variables, they missed over 25% of the correct variables and hence had very high error rates. In terms of prediction accuracy, the CV and double Dantzig or double lasso methods performed similarly. However, as with the other simulations, the CV methods selected far more variables than the other approaches. In this setting the LASSO CV outperformed the DS CV but the double Dantzig outperformed the double lasso method. Both differences were statistically significant at the 5% level. None of the other simulations contained any statistically significant differences between the lasso and Dantzig selector.

Overall, the simulation results suggest very similar performance between the Dantzig selector and the lasso. To try to understand this result better we computed the differences between the lasso and Dantzig selector paths for each simulation run at each value of λ. For all four simulations the mean differences in the paths, averaged over all values of λ, were extremely low. The maximum differences were somewhat larger, especially for the fourth simulation. However, among the approaches that we investigated, these differences tended to be in suboptimal parts of the coefficient path. Amazingly, among the 300 data sets in the first three simulations, there was not a single one where the lasso and Dantzig selector differed when using the theoretical optimum value for λ. Even in the fourth simulation, where we might expect significant differences, only about a quarter of the data sets resulted in a difference at the theoretical optimum. This seems to provide strong evidence that the non-asymptotic bounds for the Dantzig selector also hold in most situations for the lasso.

5. Discussion

Our aim here has not been definitively to address which approach is superior: the lasso or Dantzig selector. Rather, we identify the similarities between the two methods and, through our simulation study, provide some suggestions about the situations where one approach may outperform the other.

Our DASSO algorithm means that the Dantzig selector can be implemented with similar computational cost to that of the lasso. Modulo one extra calculation in the direction step, the LARS and DASSO algorithms should perform equally fast. As a result of its optimization criterion, for the same value of $\|\hat{\beta}\|_1$ the Dantzig selector does a superior job of driving the correlation between the predictors and the residual vector to 0. In searching for these reduced correlations the Dantzig selector tends to produce less smooth coefficient paths: a potential concern if the coefficient estimates change dramatically for a small change in λ_{D}. Our simulation results suggest that in practice the rougher paths may cause a small deterioration in accuracy when using cross-validation to choose the tuning parameter. However, averaged over all four simulations, the performance of the double Dantzig method was the best of the six approaches that we examined, suggesting that this method may warrant further investigation. Finally, it is worth mentioning that, although our focus here has been on linear regression, the DASSO algorithm can also be used in fields such as compressive sensing (Baraniuk *et al.*, 2008; Candes *et al.*, 2006), where the Dantzig selector is frequently applied to recover sparse signals from random projections.

Acknowledgements

We thank the Associate Editor and referees for many helpful suggestions that improved the paper. The IQ data set was kindly supplied by Dr Adrian Raine. This work was partially supported by National Science Foundation grant DMS-0705312. Lv's research was partially supported by National Science Foundation grant DMS-0806030 and the 2008 Zumberge Individual Award from the James H. Zumberge Faculty Research and Innovation Fund at the University of Southern California.

Appendix A: Direction step

Write $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$ for the positive and negative parts of $\boldsymbol{\beta}^l$. Suppose that the indices in $\mathcal{A}$ are ordered according to the time that they were added and write $S_{\mathcal{A}}$ for the diagonal matrix whose diagonal is given by $\mathbf{s}_{\mathcal{A}}$. The set-up of the algorithm and the one at a time condition ensure that $|\mathcal{A}|=|\mathcal{B}|+1$. If the set $\mathcal{B}$ is empty, add to it the index of the variable that is most correlated with the current residual. In all other cases, the index to be either added to $\mathcal{B}$ or removed from $\mathcal{A}$ is determined by the calculations in the following three steps.

Step 1: compute the $|\mathcal{A}| \times 2p+|\mathcal{A}|$ matrix $A=(-S_{\mathcal{A}}X_{\mathcal{A}}^{\mathrm{T}}X \quad S_{\mathcal{A}}X_{\mathcal{A}}^{\mathrm{T}}X \quad I)$, where I is the identity matrix. The first p columns correspond to $\boldsymbol{\beta}^+$ and the next p columns to $\boldsymbol{\beta}^-$.

Step 2: let $\tilde{B}$ be the matrix that is produced by selecting all the columns of A that correspond to the non-zero components of $\boldsymbol{\beta}^+$ and $\boldsymbol{\beta}^-$, and let A_i be the ith column of A. Write $\tilde{B}$ and A_i in the following block form:

$$\tilde{B}=\begin{pmatrix} B_1 \\ B_2 \end{pmatrix},$$

$$A_i=\begin{pmatrix} A_{i_1} \\ A_{i_2} \end{pmatrix},$$

where B_1 is a square matrix of dimension $|\mathcal{A}|-1$, and A_{i_2} is a scalar. Define

$$i^* = \underset{i:q_i\neq 0,\alpha/q_i>0}{\arg\max}\, (\mathbf{1}^{\mathrm{T}}B_1^{-1}A_{i_1}-\mathbf{1}_{\{i\leqslant 2p\}})|q_i|^{-1} \tag{9}$$

where $q_i=A_{i_2}-B_2B_1^{-1}A_{i_1}$ and $\alpha=B_2B_1^{-1}\mathbf{1}-1$.

Step 3: if $i^* \leqslant 2p$, augment $\mathcal{B}$ by the index of the corresponding β-coefficient. If $i^*=2p+m$ for $m\geqslant 1$, leave $\mathcal{B}$ unchanged, but remove the mth element from the set $\mathcal{A}$.

Note that B_1^{-1} is typically the $(X_{\mathcal{A}}^{\mathrm{T}}X_{\mathcal{B}})^{-1}$ matrix from the previous iteration of the algorithm. In addition, the current $(X_{\mathcal{A}}^{\mathrm{T}}X_{\mathcal{B}})^{-1}$ matrix can be efficiently computed by using a simple update from B_1^{-1}, as can be seen from the proof of lemma 1 in Appendix C.1.

Appendix B: Distance step

The following formulae for computing the distance until the next break point on the coefficient path are exactly the same as those for the LARS algorithm. There are two scenarios that can cause the coefficient path to change direction. The first occurs when

$$\max_{j\in\mathcal{A}^c} |\mathbf{x}_j^{\mathrm{T}}\{\mathbf{Y}-X(\boldsymbol{\beta}^l+\gamma\mathbf{h})\}| = \max_{j\in\mathcal{A}} |\mathbf{x}_j^{\mathrm{T}}\{\mathbf{Y}-X(\boldsymbol{\beta}^l+\gamma\mathbf{h})\}|,$$

because this is the point where a new variable enters the active set $\mathcal{A}$. Standard algebra shows that the first point at which this will happen is given by

$$\gamma_1 = \overset{+}{\min_{j\in\mathcal{A}^c}} \left\{ \frac{c_k-c_j}{(\mathbf{x}_k-\mathbf{x}_j)^{\mathrm{T}}X\mathbf{h}}, \frac{c_k+c_j}{(\mathbf{x}_k+\mathbf{x}_j)^{\mathrm{T}}X\mathbf{h}} \right\},$$

where $\min^+$ is the minimum taken over the positive components only, c_k and c_j are the kth and jth components of $\mathbf{c}$ evaluated at $\boldsymbol{\beta}^l$ and $\mathbf{x}_k$ represents any variable that is a member of the current active set. The second possible break in the path occurs if a coefficient path crosses zero, i.e. $\beta_j^l+\gamma h_j=0$. It is easily verified that the corresponding distance is given by $\gamma_2 = \min_j^+\{-\beta_j^l/h_j\}$. Combining the two possibilities, we set $\gamma = \min\{\gamma_1,\gamma_2\}$.

Appendix C: Proof of theorem 1

As we have mentioned, $\boldsymbol{\beta}^1$ solves the Dantzig selector optimization problem (2) for $\lambda_{\mathrm{D}} = \|X^{\mathrm{T}}\mathbf{Y}\|_\infty$. We need to establish the induction step between $\boldsymbol{\beta}^l$ and $\boldsymbol{\beta}^{l+1}$ for $l<L$. Throughout this section we assume that the one at a time condition holds, $\mathcal{A}$, $\mathcal{B}$ and $\mathbf{c}$ are defined as in step 2 of the DASSO algorithm and $\mathbf{h}$ is defined as in step 3 of the algorithm. Suppose that we have established that $\boldsymbol{\beta}^l$ solves problem (2) for $\lambda_{\mathrm{D}} = \|\mathbf{c}\|_\infty$. It is left to show that $\boldsymbol{\beta}^l+t\mathbf{h}$ solves problem (2) for $\lambda_{\mathrm{D}} = \|\mathbf{c}\|_\infty - t$ and t in $[0,\gamma]$. This fact is a consequence of the following result, which we prove at the end of the section.

Lemma 1. For each t in $[0,\gamma]$, vector $\boldsymbol{\beta}^l(t)=\boldsymbol{\beta}^l+t\mathbf{h}$ solves the optimization problem

$$\min(\|\tilde{\boldsymbol{\beta}}\|_1) \qquad \text{subject to } S_{\mathcal{A}}X_{\mathcal{A}}^{\mathrm{T}}(\mathbf{Y}-X\tilde{\boldsymbol{\beta}}) \leqslant \|\mathbf{c}\|_\infty - t. \tag{10}$$

Optimization problems (2) and (10) have the same criterion, but the constraint set in problem (2) is a subset of the constraint set in problem (10). Hence, if $\boldsymbol{\beta}^l(t)$ is a feasible solution for problem (2), it is also an optimal solution. The Dantzig selector constraints that remain active throughout the current iteration of the algorithm are satisfied for $\boldsymbol{\beta}^l(t)$, because the corresponding covariances do not change their signs in between the two break points. The remaining constraints are automatically satisfied because of the choice of γ in the distance step of the algorithm. This completes the proof of theorem 1.

C.1. Proof of lemma 1

For concreteness assume that the lth break point occurred because of an increase in the active set $\mathcal{A}$. The proof for the zero-crossing case is almost identical. Denote the size of $\mathcal{A}$ by K and recall the definitions from appendices A and B. Note that the K columns of X given by $\mathcal{A}$ are linearly independent because, for a column to enter the active set, it must be linearly independent from the columns that are already in it. Consequently, $\mathrm{rank}(A)=K$. Let $\mathbf{d}$ stand for the $(2p+K)$-vector with 1s for the first $2p$ components and 0s for the remaining components, and write $\mathbf{b}$ for the K-vector $(\|\mathbf{c}\|_\infty - t)\mathbf{1} - S_{\mathcal{A}}X_{\mathcal{A}}^{\mathrm{T}}\mathbf{Y}$. Problem (10) can be reformulated as a linear program:

$$\min(\mathbf{d}^{\mathrm{T}}\mathbf{x}) \qquad \text{subject to } A\mathbf{x}=\mathbf{b}, \quad \mathbf{x}\geqslant 0. \tag{11}$$

The correspondence between vectors $\mathbf{x}$ and $\tilde{\boldsymbol{\beta}}$ is as follows: the first p components of $\mathbf{x}$ contain the positive parts of $\tilde{\boldsymbol{\beta}}$, the second p components of $\mathbf{x}$ contain the negative parts and the remaining K components are slack variables that correspond to the K constraints in optimization problem (10). Write $\mathbf{x}^l$ for the vector corresponding to $\boldsymbol{\beta}^l$ and let $\tilde{\mathbf{x}}^l$ stand for the subvector of non-zero elements of $\mathbf{x}^l$ with one extra 0 at the bottom. Note that $\mathbf{x}^l$ has $K-1$ non-zero components, but the number of active constraints has just increased to K; hence we must increase the number of non-zero components to satisfy the constraints.

We shall search for an optimal solution to the linear program (11) among those vectors that have exactly K non-zero components. Write B for a $K\times K$ submatrix of A whose first $K-1$ columns correspond to the $K-1$ non-zero components of $\mathbf{x}^l$. We shall consider solutions whose non-zero components correspond to the columns of B. Each feasible solution of this form is represented by the subvector $\mathbf{x}_B$ of length K that satisfies $\mathbf{x}_B = B^{-1}\mathbf{b}$. Such B is necessarily invertible, because otherwise its last column is just a linear combination of the rest; thus continuing the solution path through the lth break point along the original direction would not violate the constraints, which is impossible. The following fact is established by standard linear programming arguments; for example see page 298 of Bazaraa *et al.* (1990). The last column of B can be assigned in such a way that, if we define $\mathbf{x}_B(t) = \tilde{\mathbf{x}}^l - tB^{-1}\mathbf{1}$ and let δ be the first positive t at which a component of $\mathbf{x}_B(t)$ crosses zero, then vector $\mathbf{x}_B(t)$ represents an optimal solution to the linear program (11) for each t in $[0, \delta]$. We shall now check that the index of the correct last column to be assigned to B is the index that is specified by equation (9) in the direction step of the algorithm.

Suppose that we try A_i as the last column of matrix B. Simple algebra shows that B is invertible if and only if $q_i \neq 0$, and the inverse has the following four-block form:

$$B^{-1} = \begin{pmatrix} B_1^{-1} + B_1^{-1} A_{i_1} B_2 B_1^{-1} q_i^{-1} & -B_1^{-1} A_{i_1} q_i^{-1} \\ -B_2 B_1^{-1} q_i^{-1} & q_i^{-1} \end{pmatrix}. \tag{12}$$

Observe that when an A_i corresponding to a β_j^+ is replaced by the column corresponding to β_j^-, or vice versa, the sign of α/q_i is flipped; hence the set $\{i : q_i \neq 0, \alpha/q_i \geqslant 0\}$ is non-empty. Write $\mathbf{d}_B$ for the subvector of $\mathbf{d}$ corresponding to the columns of B. The cost value that is associated with $\mathbf{x}_B(t)$ is given by $\mathbf{d}_B^{\mathrm{T}} B^{-1} b = \mathbf{d}^{\mathrm{T}} \mathbf{x}^l - t\mathbf{d}_B^{\mathrm{T}} B^{-1}\mathbf{1}$. Formula (12) implies that

$$\mathbf{d}_B^{\mathrm{T}} B^{-1}\mathbf{1} = \mathbf{1}^{\mathrm{T}} B_1^{-1}\mathbf{1} + (\mathbf{1}^{\mathrm{T}} B_1^{-1} \mathbf{A}_{i_1} - \mathbf{1}_{\{i \leqslant 2p\}})\alpha/q_i. \tag{13}$$

For each t in $[0, \delta]$, the last component of $\mathbf{x}_B(t)$ has the same sign as the last component of $-B^{-1}\mathbf{1}$, which is given by α/q_i. Thus, $\mathbf{x}_B(t)$ is a feasible solution if and only if $\alpha/q_i \geqslant 0$. Note that $\alpha = 0$ would imply that the first $K-1$ components of $B^{-1}\mathbf{1}$ are given by $B_1^{-1}\mathbf{1}$, which again leads to a contradiction. Consequently, using A_{i^*} as the last column of B will produce an optimal solution if and only if i^* maximizes the second term on the right-hand side of equation (13) over the set $\{i : q_i \neq 0, \alpha/q_i > 0\}$, as is prescribed, indeed, by the direction step. Because $\gamma \leqslant \delta$ by the definition of γ in the distance step, $\boldsymbol{\beta}^l + t\mathbf{h}$ is an optimal solution to problem (10) for each t in $[0, \gamma]$.

Appendix D: Proof of theorem 2

Define two index sets, $I_1 = \{1 \leqslant j \leqslant p : \zeta_j = \lambda_{\mathrm{L}}\}$ and $I_2 = \{1 \leqslant j \leqslant p : \zeta_j = -\lambda_{\mathrm{L}}\}$, where $X^{\mathrm{T}}(\mathbf{Y} - X\hat{\boldsymbol{\beta}}_{\mathrm{L}}) = (\zeta_1, \ldots, \zeta_p)^{\mathrm{T}}$, and let $I = I_1 \cup I_2$. For notational convenience, we let $f(\tilde{\boldsymbol{\beta}}) = \|\tilde{\boldsymbol{\beta}}\|_1$. Let X_{I_1,I_2} be an $n \times |I|$ matrix that is generated from the columns of X that correspond to the indices in I after multiplying those columns in index set I_2 by -1. Define two index sets $I_1^0 = I_1 \cap I_{\mathrm{L}}$ and $I_2^0 = I_2 \cap I_{\mathrm{L}}$. We show in the proof of theorem 4 that $I_{\mathrm{L}} \subset I = I_1 \cup I_2$, and thus it follows that $I_1^0 \cup I_2^0 = I_{\mathrm{L}}$. Define matrix $X_{I_1^0,I_2^0}$ analogously to X_{I_1,I_2}. Note that $X_{I_1^0,I_2^0}$ and X_{L} are identical. Hence, it is easy to see that, provided that $I = I_{\mathrm{L}}$, condition (14) holds whenever condition (8) is true. By setting certain elements of $\mathbf{u}$ to 0 the same result also holds at the break points where I_{L} is a subset of I. Therefore, theorem 2 is a direct consequence of the following result.

Theorem 4. Assume that $X^{\mathrm{T}} X_{I_1,I_2}$ has full rank $|I|$ and $\lambda_{\mathrm{D}} = \lambda_{\mathrm{L}}$. Then $\hat{\boldsymbol{\beta}}_{\mathrm{D}} = \hat{\boldsymbol{\beta}}_{\mathrm{L}}$ if and only if there is an $|I|$-vector $\mathbf{u}$ with non-negative components such that

$$\mathbf{0} \in \partial_{\tilde{\boldsymbol{\beta}}} f(\hat{\boldsymbol{\beta}}_{\mathrm{L}}) - X^{\mathrm{T}} X_{I_1,I_2}\mathbf{u}, \tag{14}$$

where $\partial_{\tilde{\boldsymbol{\beta}}} f(\hat{\boldsymbol{\beta}}_{\mathrm{L}}) = \{(v_1, \ldots, v_p)^{\mathrm{T}} : v_j = \mathrm{sgn}(\hat{\boldsymbol{\beta}}_{\mathrm{L}}, j)$ if $j \in I_{\mathrm{L}}$ and $v_j \in [-1, 1]$ otherwise$\}$.

We spend the remainder of this section proving theorem 4. Denote by $l(\tilde{\boldsymbol{\beta}}, \lambda_{\mathrm{L}})$ the objective function in equation (4). It is easy to see that $l(\tilde{\boldsymbol{\beta}}, \lambda_{\mathrm{L}})$ is a convex function in $\tilde{\boldsymbol{\beta}}$, and thus the zero p-vector $\mathbf{0}$ belongs to the subdifferential $\partial_{\tilde{\boldsymbol{\beta}}} l(\hat{\boldsymbol{\beta}}_{\mathrm{L}}, \lambda_{\mathrm{L}})$ since $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ minimizes $l(\tilde{\boldsymbol{\beta}}, \lambda_{\mathrm{L}})$. The subdifferential of $l(\tilde{\boldsymbol{\beta}}, \lambda_{\mathrm{L}})$ in $\tilde{\boldsymbol{\beta}}$ is given by

$$\begin{aligned} \partial_{\tilde{\boldsymbol{\beta}}} l(\tilde{\boldsymbol{\beta}}, \lambda_{\mathrm{L}}) &= \tfrac{1}{2} \nabla_{\tilde{\boldsymbol{\beta}}} \|\mathbf{Y} - X\tilde{\boldsymbol{\beta}}\|_2^2 + \lambda_{\mathrm{L}} \partial_{\tilde{\boldsymbol{\beta}}} \|\tilde{\boldsymbol{\beta}}\|_1 \\ &= -X^{\mathrm{T}}(\mathbf{Y} - X\tilde{\boldsymbol{\beta}}) + \lambda_{\mathrm{L}} \partial_{\tilde{\boldsymbol{\beta}}} \|\tilde{\boldsymbol{\beta}}\|_1, \end{aligned}$$

where it is a known fact that, for $\tilde{\boldsymbol{\beta}} = (\beta_1, \ldots, \beta_p)^{\mathrm{T}}$,

$$\partial_{\tilde{\boldsymbol{\beta}}} \|\tilde{\boldsymbol{\beta}}\|_1 = \{\mathbf{v} = (v_1, \ldots, v_p)^{\mathrm{T}} : v_j = \mathrm{sgn}(\beta_j) \text{ if } \beta_j \neq 0 \text{ and } v_j \in [-1, 1] \text{ if } \beta_j = 0,\ i = 1, \ldots, p\}.$$

Thus, there is some $\mathbf{v} \in \partial_{\tilde{\boldsymbol{\beta}}} \|\tilde{\boldsymbol{\beta}}\|_1$ at $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ such that $X^{\mathrm{T}}(\mathbf{Y} - X\hat{\boldsymbol{\beta}}_{\mathrm{L}}) - \lambda_{\mathrm{L}}\mathbf{v} = \mathbf{0}$. In particular, we see that $\|X^{\mathrm{T}}(\mathbf{Y} - X\hat{\boldsymbol{\beta}}_{\mathrm{L}})\|_\infty \leqslant \lambda_{\mathrm{L}}$. Note that the set of indices $I_{\mathrm{L}} = \{1 \leqslant j \leqslant p : \hat{\beta}_{\mathrm{L},j} \neq 0\}$ that is selected by the lasso is a subset of the index set $I=\{1\leqslant j\leqslant p:|\zeta_j|=\lambda_{\mathrm{L}}\}$ and, for any $j\in I_{\mathrm{L}}$, $\boldsymbol{\zeta}_j=\operatorname{sgn}(\hat{\beta}_{\mathrm{L},j})\lambda_{\mathrm{L}}$, where $(\zeta_1,\ldots,\zeta_p)^{\mathrm{T}}=X^{\mathrm{T}}(\mathbf{Y}-X\hat{\boldsymbol{\beta}}_{\mathrm{L}})$.

Now take $\lambda_{\mathrm{D}}=\lambda_{\mathrm{L}}$. Then, we have shown that $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ is a feasible solution to the minimization problem (2), but it might not be an optimal solution. Let us investigate when $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ is indeed the Dantzig selector, i.e. a solution to problem (2). We make a simple observation that the constraint $\|X^{\mathrm{T}}(\mathbf{Y}-X\tilde{\boldsymbol{\beta}})\|_\infty\leqslant\lambda_{\mathrm{D}}$ can equivalently be written into a set of $2p$ linear constraints, $\mathbf{x}_j^{\mathrm{T}}(\mathbf{Y}-X\tilde{\boldsymbol{\beta}})\leqslant\lambda_{\mathrm{D}}$ and $-\mathbf{x}_j^{\mathrm{T}}(\mathbf{Y}-X\tilde{\boldsymbol{\beta}})\leqslant\lambda_{\mathrm{D}}$, $j=1,\ldots,p$.

Suppose that the $p\times|I|$ matrix $X^{\mathrm{T}}X_{I_1,I_2}$ has full rank $|I|$, which means that all its column p-vectors are linearly independent. Clearly, this requires that $|I|\leqslant\min(n,p)$. Since the minimization problem (2) is convex, it follows from classical optimization theory that $\hat{\boldsymbol{\beta}}_{\mathrm{L}}$ is a solution to problem (2) if and only if it is a Karush–Kuhn–Tucker point, i.e. there is an $|I|$-vector $\mathbf{u}$ with non-negative components such that $\mathbf{0}\in\partial_{\tilde{\boldsymbol{\beta}}}f(\hat{\boldsymbol{\beta}}_{\mathrm{L}})-X^{\mathrm{T}}X_{I_1,I_2}\mathbf{u}$, where the matrix $X^{\mathrm{T}}X_{I_1,I_2}$ is associated with the gradients of the linear constraints in the Dantzig selector.

Appendix E: Proof of theorem 3

When all the column n-vectors of the $n\times p$ design matrix X have unit norm and equal pairwise correlation ρ it will be the case that

$$X^{\mathrm{T}}X=\mathbf{A}=(1-\rho)I_p+\rho\mathbf{1}\mathbf{1}^{\mathrm{T}}. \tag{15}$$

A direct calculation using the Sherman–Morrison–Woodbury formula gives

$$\mathbf{A}^{-1}=\frac{1}{1-\rho}I_p-\frac{\rho}{(1-\rho)\{1+(p-1)\rho\}}\mathbf{1}\mathbf{1}^{\mathrm{T}}, \tag{16}$$

which shows that $\mathbf{A}$ is invertible for each $\rho\in[0,1)$. Thus, equation (15) in fact entails that $p\leqslant n$.

Now, we show that equation (15) implies condition (8). Note that all principal submatrices of $\mathbf{A}$ have the same form as $\mathbf{A}$. Thus, to show that the first part of condition (8) holds, without loss of generality we need only to show that it holds for $I_{\mathrm{L}}=\{1,\ldots,p\}$. Let $\mathbf{v}=(v_1,\ldots,v_p)^{\mathrm{T}}$ be a p-vector with components 1 or -1, where $\{1\leqslant j\leqslant p:v_j=1\}=I_1^0$ and $\{1\leqslant j\leqslant p:v_j=-1\}=I_2^0$. It follows from equation (15) that $\mathbf{u}=(X_{\mathrm{L}}X_{\mathrm{L}})^{-1}\mathbf{1}=\operatorname{diag}(\mathbf{v})\mathbf{A}^{-1}\mathbf{v}$, where $\operatorname{diag}(\mathbf{v})$ denotes a diagonal matrix with diagonal elements being the components of vector $\mathbf{v}$. Then, we need to prove that $\operatorname{diag}(\mathbf{v})\mathbf{A}^{-1}\mathbf{v}\geqslant\mathbf{0}$. Note that $\operatorname{diag}(\mathbf{v})\mathbf{v}=\mathbf{1}$, $\operatorname{diag}(\mathbf{v})\mathbf{1}=\mathbf{v}$ and $|\mathbf{1}^{\mathrm{T}}\mathbf{v}|\leqslant p$. Thus, by equation (16), we have, for any $\rho\in[0,1)$,

$$\operatorname{diag}(\mathbf{v})\mathbf{A}^{-1}\mathbf{v}=\frac{1}{1-\rho}\left(\mathbf{1}-\frac{\mathbf{1}^{\mathrm{T}}\mathbf{v}\rho}{1-\rho+p\rho}\mathbf{v}\right)\geqslant\mathbf{0},$$

which proves the first part of condition (8).

It remains to prove the second part of condition (8). We need only to show that, for each $j\in\{1,\ldots,p\}\setminus I_{\mathrm{L}}$,

$$|\mathbf{x}_j^{\mathrm{T}}X_{\mathrm{L}}\mathbf{u}|\leqslant 1. \tag{17}$$

Let $\mathbf{v}$ be an $|I_{\mathrm{L}}|$-vector with components 1 or -1, where $\{1\leqslant j\leqslant|I_{\mathrm{L}}|:v_j=1\}$ and $\{1\leqslant j\leqslant|I_{\mathrm{L}}|:v_j=-1\}$ correspond to I_1^0 and I_2^0 in the index set I_{L} respectively. We denote by $\mathbf{A}_p=\mathbf{A}$ to emphasize its order. Then, we have shown that

$$\mathbf{u}=(X_{\mathrm{L}}^{\mathrm{T}}X_{\mathrm{L}})^{-1}\mathbf{1}=\operatorname{diag}(\mathbf{v})\mathbf{A}_{|I_{\mathrm{L}}|}^{-1}\mathbf{v}=\frac{1}{1-\rho}\left(\mathbf{1}-\frac{\mathbf{1}^{\mathrm{T}}\mathbf{v}\rho}{1-\rho+|I_{\mathrm{L}}|\rho}\mathbf{v}\right)\geqslant\mathbf{0}. \tag{18}$$

Note that all the off-diagonal elements of $\mathbf{A}$ are ρ. Hence it follows from equation (18), equality $\mathbf{v}^{\mathrm{T}}\mathbf{v}=|I_{\mathrm{L}}|$ and inequality $|\mathbf{v}^{\mathrm{T}}\mathbf{1}|\leqslant|I_{\mathrm{L}}|$ that, for each $\rho\in[0,1)$,

$$\begin{aligned}|\mathbf{x}_j^{\mathrm{T}}X_{\mathrm{L}}\mathbf{u}|=|\rho\mathbf{v}^{\mathrm{T}}\mathbf{u}|&=\left|\frac{\rho}{1-\rho}\left(\mathbf{1}-\frac{\mathbf{1}^{\mathrm{T}}\mathbf{v}\rho}{1-\rho+|I_{\mathrm{L}}|\rho}\mathbf{v}\right)\right|\\&=\left|\frac{\rho}{1-\rho}\mathbf{v}^{\mathrm{T}}\mathbf{1}\left(1-\frac{|I_{\mathrm{L}}|\rho}{1-\rho+|I_{\mathrm{L}}|\rho}\right)\right|\leqslant\frac{|I_{\mathrm{L}}|\rho}{1-\rho}\frac{1-\rho}{1-\rho+|I_{\mathrm{L}}|\rho}\\&=\frac{|I_{\mathrm{L}}|\rho}{1-\rho+|I_{\mathrm{L}}|\rho}<1,\end{aligned}$$

which proves inequality (17).

References

Baraniuk, R., Davenport, M., DeVore, R. and Wakin, M. (2008) A simple proof of the restricted isometry property for random matrices. *Construct. Approximn*, to be published.

Bazaraa, M., Jarvis, J. and Sherali, H. (1990) *Linear Programming and Network Flows*, 2nd edn. New York: Wiley.

Bickel, P. J., Ritov, Y. and Tsybakov, A. (2008) Simultaneous analysis of Lasso and Dantzig selector. *Ann. Statist.*, to be published.

Breiman, L. (1995) Better subset regression using the non-negative garrote. *Technometrics*, **37**, 373–384.

Candes, E. and Romberg, J. (2006) Quantitative robust uncertainty principles and optimally sparse decompositions. *Foundns Comput. Math.*, **6**, 227–254.

Candes, E., Romberg, J. and Tao, T. (2006) Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inform. Theory*, **52**, 489–509.

Candes, E. and Tao, T. (2005) Decoding by linear programming. *IEEE Trans. Inform. Theory*, **51**, 4203–4215.

Candes, E. and Tao, T. (2007a) The Dantzig selector: statistical estimation when p is much larger than n. *Ann. Statist.*, **35**, 2313–2351.

Candes, E. and Tao, T. (2007b) Rejoinder—the Dantzig selector: Statistical estimation when p is much larger than n. *Ann. Statist.*, **35**, 2392–2404.

Chen, S., Donoho, D. and Saunders, M. (1998) Atomic decomposition by basis pursuit. *SIAM J. Scient. Comput.*, **20**, 33–61.

Donoho, D. (2006) For most large underdetermined systems of linear equations, the minimal l_1-norm near-solution approximates the sparsest near-solution. *Communs Pure Appl. Math.*, **59**, 907–934.

Donoho, D., Elad, M. and Temlyakov, V. (2006) Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Inform. Theory*, **52**, 6–18.

Donoho, D. and Tanner, J. (2005) Sparse nonnegative solution of underdetermined linear equations by linear programming. *Proc. Natn. Acad. Sci. USA*, **102**, 9446–9451.

Efron, B., Hastie, T., Johnstone, I. and Tibshirani, R. (2004) Least angle regression (with discussion). *Ann. Statist.*, **32**, 407–451.

Efron, B., Hastie, T. and Tibshirani, R. (2007) Discussion of the "dantzig selector". *Ann. Statist.*, **35**, 2358–2364.

Fan, J. and Li, R. (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Statist. Ass.*, **96**, 1348–1360.

James, G. M. and Radchenko, P. (2008) A generalized Dantzig selector with shrinkage tuning. *Technical Report*. University of Southern California, Los Angeles.

Meinshausen, N. (2007) Relaxed Lasso. *Computnl Statist. Data Anal.*, **52**, 374–393.

Meinshausen, N. and Bühlmann, P. (2006) High dimensional graphs and variable selection with the lasso. *Ann. Statist.*, **34**, 1436–1462.

Meinshausen, N., Rocha, G. and Yu, B. (2007) A tale of three cousins: Lasso, L2Boosting, and Dantzig (discussion on Candes and Tao's Dantzig Selector paper). *Ann. Statist.*, **35**, 2372–2384.

Plumbley, M. D. (2007) On polar polytopes and the recovery of sparse representations. *IEEE Trans. Inform. Theory*, **53**, 3188–3195.

Rosset, S. and Zhu, J. (2007) Piecewise linear regularized solution paths. *Ann. Statist.*, **35**, 1012–1030.

Shen, X. and Ye, J. (2002) Adaptive model selection. *J. Am. Statist. Ass.*, **97**, 210–221.

Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. *J. R. Statist. Soc.* B, **58**, 267–288.

Zhang, C.-H. and Huang, J. (2007) Model-selection consistency of the lasso in high-dimensional linear regression. *Manuscript*.

Zhao, P. and Yu, B. (2006) On model selection consistency of lasso. *J. Mach. Learn. Res.*, **7**, 2541–2563.

Zou, H. (2006) The adaptive lasso and its oracle properties. *J. Am. Statist. Ass.*, **101**, 1418–1429.

Zou, H. and Hastie, T. (2005) Regularization and variable selection via the elastic net. *J. R. Statist. Soc.* B, **67**, 301–320.