Economics Letters 187 (2020) 108916

Contents lists available at ScienceDirect

Economics Letters

journal homepage: www.elsevier.com/locate/ecolet

On the sparsity of Mallows model averaging estimator[☆]

Yang Feng[a], Qingfeng Liu[b,*], Ryo Okui[c]

[a] Department of Biostatistics, College of Global Public Health, New York University, New York, NY, 10003, USA
[b] Department of Economics, Otaru University of Commerce, Otaru City, Hokkaido, Japan
[c] Department of Economics and the Institute of Economic Research, Seoul National University, Seoul, South Korea

ARTICLE INFO

Article history:
Received 24 July 2019
Received in revised form 18 October 2019
Accepted 19 December 2019
Available online 19 December 2019

JEL classification:
C51
C52

Keywords:
Sparsity
Model averaging
L_1 penalty
Coordinate-wise descent algorithm

ABSTRACT

We show that Mallows model averaging estimator proposed by Hansen (2007) can be written as a least squares estimation with a weighted L_1 penalty and additional constraints. By exploiting this representation, we demonstrate that the weight vector obtained by this model averaging procedure has a sparsity property in the sense that a subset of models receives exactly zero weights. Moreover, this representation allows us to adapt algorithms developed to efficiently solve minimization problems with many parameters and weighted L_1 penalty. In particular, we develop a new coordinate-wise descent algorithm for model averaging. Simulation studies show that the new algorithm computes the model averaging estimator much faster and requires less memory than conventional methods when there are many models.

1. Introduction

For a set of candidate models, model averaging aims to provide accurate predictions by combining the results from individual models. Subsequent to the seminal work of Mallows model averaging (MMA) (Hansen, 2007), model averaging has received a great deal of attention from econometricians and statisticians with an array of model averaging estimators proposed under various contexts; see Claeskens and Hjort (2008) and Moral-Benito (2015) for an overview. Among existing studies, researchers have mainly focused on the asymptotic properties of model averaging methods. However, there are open questions remaining for model averaging estimators. First, the finite sample behavior of model averaging is not yet considered. Second, the number of models to be averaged may be significant in practice, but the existing results and algorithms can only handle a relatively small number of models. For instance, a typical approach to solving the optimization problem of MMA is quadratic programming (QP) as described in Hansen (2007). However, while QP works well for low-dimensional problems, it becomes computationally prohibitive, and the memory requirements grow rapidly as the number of models under consideration increases. For example, suppose that we average linear regression models where the models differ in the set of regressors. If we are unwilling to place any structural assumptions on the candidate model set, there are then 2^p models where p is the number of regressors. This can be extremely large, even for a moderate p.

This letter demonstrates that the optimization problem involved in MMA to compute the weight vector can be viewed as a constrained least squares estimation with weighted L_1 penalty, i.e., constrained weighted Lasso. Lasso (Tibshirani, 1996) has received much attention from researchers in many different fields. The constrained Lasso representation of MMA may be of interest in its own right as it connects Lasso to model averaging, a matter not yet acknowledged in the literature. Moreover, this observation provides three crucial implications.

First, we show that in MMA, the estimated weight vector is sparse in finite samples for both underfitted and overfitted models. An important implication of the sparsity property is that MMA can handle a large number of models. Second, the constrained weighted Lasso representation of MMA suggests opportunities to adapt efficient algorithms developed for Lasso. Because Lasso has been widely used, several fast algorithms, such as Friedman et al. (2009), have been developed to compute the estimator. In particular, for MMA, we propose a new coordinate-wise descent (CD) algorithm, which is much faster and requires less memory than the commonly used QP algorithm. Third, with the new representation of MMA, we expect that the fields of MMA and Lasso

[☆] The authors gratefully acknowledge the support of a National Science Foundation, USA CAREER Grant DMS-1554804 (Feng), the Japan Society for the Promotion of Science through KAKENHI Grant Nos.JP16K03590 and JP19K01582 (Liu), a New Faculty Startup Grant from Seoul National University, South Korea, and the Housing and Commercial Bank Economic Research Fund from the Institute of Economic Research at Seoul National University, South Korea (Okui). The authors are solely responsible for any errors.

[*] Corresponding author.
E-mail address: qliu@res.otaru-uc.ac.jp (Q. Liu).

https://doi.org/10.1016/j.econlet.2019.108916
0165-1765/© 2019 Elsevier B.V. All rights reserved.

can borrow strength from each other and the new representation would provide interesting future research topics. For example, we may be able to develop theoretical properties of MMA by taking advantage of rich literature on the theory of Lasso. Also, MMA may provide some insights on the tuning parameter selection problem for Lasso as it corresponds to a particular value of the tuning parameter.

Zhang and Liu (2019) showed that the elements of the weight vector of MMA on underfitted models converge to zero asymptotically at a rate faster than that for the coefficients. Our results are nonasymptotic, and we show that the weight vector is sparse in finite samples. We also show that some overfitted models also receive zero weights in MMA. Note that Zhang and Liu (2019) also considered a modification of MMA so that the weights on the overfitted models converge to zero quickly, but this is still an asymptotic result and not a finite sample.

Another example of weighted Lasso is the recently proposed SLOPE (Bogdan et al., 2015; Bellec et al., 2018), which was motivated by controlling the false discovery rate for model selection. In particular, SLOPE uses the so-called sorted L_1 penalty, where the penalty level depends on the ordering of the estimated coefficients with the larger estimated coefficients receiving higher penalties. On the other hand, the penalty level corresponding to MMA depends on the size of each candidate model with the larger models receiving higher penalties. Therefore, MMA and SLOPE correspond to two very different weighting schemes in penalization.

The remainder of this letter is organized as follows. Section 2 shows that the MMA estimator can be viewed as a constrained weighted Lasso problem. We show the estimator to be sparse with the sparsity set identified in Section 3. To solve the optimization problem, Section 4 proposes a CD algorithm that turns out to be more efficient in terms of both memory and speed than the commonly used QP algorithm. The sparsity property of the CD and its computational advantages over QP are demonstrated via simulation studies in Section 5. We conclude the letter with a short discussion in Section 6.

2. MMA as a constrained weighted Lasso problem

Suppose we have a real-valued random sample $\{(x_i, y_i), i = 1, \dots, n\}$, where $x_i = (x_{i1}, x_{i2}, \dots)$ has countable elements. We consider the following linear model as described in Hansen (2007):

$$y_i = f_i + e_i, \tag{1}$$

where $f_i = \sum_{j=1}^{\infty} \theta_j x_{ij}$, in which some θ_j can equal zero, and e_i is a zero-mean unobservable random error with $E(e_i^2|x_i) = \sigma^2$. The goal is to use the random sample to obtain an estimate $\hat{\mu}$ for the true mean response $\mu = (f_1, \dots, f_n)^T$.

First, we provide a brief review of model averaging. Assume there are M candidate models and estimates of μ based on these models are available. For $m = 1, \dots, M$, let $\hat{\mu}_m$ and k_m be the estimate of μ and the number of parameters in the mth model, respectively. The premise of model averaging is to identify a weight vector $w = (w_1, \dots, w_M)^T$ such that $\sum_{m=1}^{M} w_m \hat{\mu}_m$ is a good estimate of μ.

MMA as proposed by Hansen (2007) uses linear regression models with different sets of predictors as candidate models. Suppose X_m is the design matrix corresponding to model m, then the ordinary least square estimate is $\hat{\beta}_m = (X_m^T X_m)^{-1} X_m^T y$ and the corresponding predicted value is $\hat{\mu}_m = X_m \hat{\beta}_m = X_m (X_m^T X_m)^{-1} X_m^T y$. MMA chooses w to minimize

$$C_n(w) = \frac{1}{2} \left\| y - \sum_{m=1}^{M} w_m \hat{\mu}_m \right\|_2^2 + \hat{\sigma}^2 \sum_{m=1}^{M} w_m k_m, \tag{2}$$

$$\text{s.t. } w \in \mathcal{H}_M = \left\{ w \in [0,1]^M \mid \sum_{m=1}^{M} w_m = 1 \right\},$$

where $\|\cdot\|_2$ is the Euclidean norm and $\hat{\sigma}^2$ is an estimate of σ^2. In simulation studies, we use the average of squared residuals from the largest model as $\hat{\sigma}^2$.

The key insight of this letter is to show that the MMA can be viewed as a constrained weighted Lasso problem. First, we review the standard weighted Lasso. For the classic linear model $y_{n\times 1} = Z_{n\times M} w_{M\times 1} + \epsilon_{n\times 1}$, the standard weighted Lasso seeks the vector w that minimizes

$$\tilde{C}_n(w) = \frac{1}{2} \|y - Zw\|_2^2 + \sum_{m=1}^{M} \lambda_m |w_m|, \tag{3}$$

where $(\lambda_1, \dots, \lambda_M)$ is the penalty parameter vector for different elements of w. Next, we see the correspondence between Lasso and MMA. Let $Z = (\hat{\mu}_1, \dots, \hat{\mu}_M)_{n\times M}$ be the matrix whose columns are the fitted response vector from the M candidate models. Then, the MMA optimization problem has the following equivalent representation.

$$C_n(w) = \frac{1}{2} \|y - Zw\|_2^2 + \hat{\sigma}^2 \sum_{m=1}^{M} k_m |w_m|, \tag{4}$$

$$\text{s.t. } w \in \mathcal{H}_M = \left\{ w \in [0,1]^M \mid \sum_{m=1}^{M} w_m = 1 \right\}.$$

This representation illustrates the correspondence between the weight estimation problem of MMA and a constrained weighted Lasso where the regression coefficients are the weights (w). Regarding the weighted penalty, we have $\lambda_m = \hat{\sigma}^2 k_m$ as the penalty level for w_m, which is proportional to the number of parameters for the mth model and hence promotes smaller models. Comparing (4) to the vanilla weighted Lasso problem (3), it has additional constraints for w, namely $w \in \mathcal{H}_M$, which leads to further regularization on w.

3. Sparsity of MMA

In this section, we analyze the sparsity of the weights for MMA. We first derive the Karush–Kuhn–Tucker (KKT) condition for MMA. We then examine the condition to investigate the source of the sparsity.

The Lagrangian of the minimization problem (4) is

$$\frac{1}{2} \|y - Zw\|_2^2 + \hat{\sigma}^2 \sum_{m=1}^{M} k_m |w_m| - v'w + l(w'1_M - 1),$$

where $v = (v_1, \dots, v_M)^T$ is the vector of Lagrange multipliers for the constraints $w \geq 0$ and l is the Lagrange multiplier for $\sum_{m=1}^{M} w_m = 1$. The corresponding KKT condition is as follows:

$$-Z'(y - Zw) + \hat{\sigma}^2 \Omega s - v + l 1_M = 0, \tag{5}$$

$$w' 1_M = 1, -w_m \leq 0, v_m \geq 0, v_m w_m = 0, \tag{6}$$

where Ω is a diagonal matrix whose mth diagonal element is k_m, 1_M is an $M \times 1$ vector with all elements being 1 and $s = (s_1, \dots, s_M)^T$ with s_m being the subgradient of $|w_m|$: $s_m = \operatorname{sign}(w_m)$ if $w_m \neq 0$ and $s_m \in [-1, 1]$ if $w_n = 0$. Note that, from (5), for $w_m > 0$, we have $\hat{\mu}_m^T (y - Zw) + v_m - l = \hat{\sigma}^2 k_m$.

Proposition 1. *Define $\mathcal{A} = \left\{ m \in \{1, \dots, M\} : \left| \hat{\mu}_m^T (y - Zw) + v_m - l \right| < \hat{\sigma}^2 k_m \right\}$. We have $w_m = 0$ for $m \in \mathcal{A}$.*

Proposition 1 provides a sufficient condition for the weight w_m being zero. We analyze the set $\mathcal{A}$ for the following two scenarios.

(a) If $v_m > 0$, by (6), we have $w_m = 0$.

(b) If $v_m = 0$, we have $\left|\hat{\mu}_m^T(y - Zw) - l\right| < \hat{\sigma}^2 k_m$. A larger model tends to obtain a zero weight as the corresponding k_m is larger. Note that this finding differs from the asymptotic result in Zhang and Liu (2019), which characterized the asymptotic distribution of the weight of an overfitted model but did not discuss the sparsity. In addition, we have the following observation. Note that

$$\hat{\mu}_m^T(y - Zw) = w_m \hat{\mu}_m^T(y - \hat{\mu}_m) + (1 - w_m)\hat{\mu}_m^T \left[y - \sum_{j \neq m} \tilde{w}_j \hat{\mu}_j \right],$$

$$= (1 - w_m)\hat{\mu}_m^T \left[y - \sum_{j \neq m} \tilde{w}_j \hat{\mu}_j \right],$$

where $\tilde{w}_j = w_j / \sum_{j \neq m} w_j$. If the mth model is redundant given the other models that are considered, $\hat{\mu}_m$ should be uncorrelated with the residual $y - \sum_{j \neq m} \tilde{w}_j \hat{\mu}_j$. As a result, we expect $\hat{\mu}_m^T(y - \sum_{j \neq m} \tilde{w}_j \hat{\mu}_j) = O_p(\sqrt{n})$. Conversely, where the mth model provides important information given the other models that are considered, $\hat{\mu}_m^T(y - \sum_{j \neq m} \tilde{w}_j \hat{\mu}_j)$ would be the sum of random variables with nonzero mean, and is thus of order $O_p(n)$. As a result, a redundant model tends to be inside set $\mathcal{A}$ compared with the informative models.

We now demonstrate the sparsity phenomenon of MMA in Section 5 via extensive numerical studies.

4. A coordinate-wise descent algorithm

In this section, we introduce a new CD algorithm for calculating the MMA estimate. The idea of CD has been successfully applied to solve high-dimensional problems where the solution could be sparse, e.g., the `glmnet` algorithm (Friedman et al., 2009) for Lasso. However, our problem (4) could not be directly solved using CD optimization owing to the equality constraint $\sum_{m=1}^M w_m = 1$, which makes the optimization nonseparable for w's. Here, we recast the original problem (4) by converting the equality constraint $\sum_{m=1}^M w_m = 1$ to a quadratic penalty term while retaining the positive constraints for w's as follows.

$$\min_w \left\{ f(w) = \frac{1}{2}\|y - Zw\|_2^2 + \hat{\sigma}^2 \sum_{m=1}^M k_m |w_m| + \frac{\gamma}{2}(1_M^T w - 1)^2 \right\}, \tag{7}$$

$$\text{s.t. } w_m \geq 0, m = 1, \ldots, M,$$

for $\gamma > 0$. As long as γ is sufficiently large, the optimization problem (7) is equivalent to the original problem (4) (Ruszczyński, 2006). The subgradient with respect to w_m becomes

$$\frac{\partial f(w)}{\partial w_m} = -\hat{\mu}_m^T(y - Zw) + \hat{\sigma}^2 k_m s_m + \gamma(1_M^T w - 1). \tag{8}$$

Then, for an initial estimate w^0, we update only the mth coordinate as

$$\hat{w}_m = \frac{\left[\hat{\mu}_m^T(y - Z_{-m}w_{-m}) - \gamma(1_M^T w_{-m} - 1) - \hat{\sigma}^2 k_m\right]^+}{\gamma + \hat{\mu}_m^T \hat{\mu}_m}, \tag{9}$$

where Z_{-m} corresponds to Z without the mth column, w_{-m} is w without the mth element, and $a^+ = a$ if $a > 0$ and 0 otherwise. Algorithm 1 summarizes the process.

The CD algorithm requires much less memory than the commonly used QP algorithm to solve MMA when the number of models is large. Consider cases in which there are p regressors, and we average all subset models (i.e., $M = 2^p$). In general, the dimension of the largest matrix considered in an algorithm determines its memory cost. In Algorithm 1, the dimension of the largest matrix Z is $n \times 2^p$, so we can see that the memory cost of CD is $\Theta(n2^p)$.[1] Alternatively, for QP we use quadprog in MATLAB, in which the Hessian matrix is the largest matrix whose dimension is $2^p \times 2^p$, hence the memory cost of QP here is $\Theta(2^{2p})$. Thus, the ratio of the memory cost of CD against that of QP is $\Theta(n/2^p)$, which decreases to zero as long as n does not increase exponentially with p.

Algorithm 1 Coordinate-wise Descent Algorithm for MMA

```
Input: ${(x_i, y_i), i = 1, \cdots, n}$, penalty parameter $\gamma$, convergence threshold $\epsilon$, maximum iteration number $K$.
Output: $\hat{w}$, $\hat{\mu}$.
Initialize $\hat{w}^0 = 1/M \cdot 1_M$.
For iteration number $k = 1, \cdots, K$.
    For $m = 1, \cdots, M$, update $w_m$ according to (9). Denote the estimate as $\hat{w}^k$.
    If $\|\hat{w}^k - \hat{w}^{k-1}\| < \epsilon$, stop the iteration for $k$.
```

5. Simulation

In this section, we conduct simulation studies to compare the newly proposed CD algorithm with the QP algorithm for calculating MMA estimates. In particular, we examine their computation speeds and memory requirements.[2] We also illustrate the sparsity property of the weight vector chosen by MMA. In the CD algorithm, we set $K = 10^{10}$, $\epsilon = 10^{-10}$, and $\gamma = 10^3$. Each experiment is repeated 1000 times.[3]

Consider the data generation process in (1) with sample size $n = 500$, $x_{i1} \equiv 1$ corresponds to the intercept, $x_{ij} \overset{i.i.d.}{\sim} N(0, 1)$ for $j \geq 2$, and the true coefficient vector is $\theta = c \cdot (1, 1, 1, 1/2, 1/2, 1/2, 1/3, 1/3, 1/3, 0, \ldots)^T$, where c is chosen to obtain a specific R^2 value and $\theta_j = 0$ for $j > 9$. To make a comprehensive comparison, we vary R^2 from 0.1 to 0.9 with increment 0.1 and the number of observed regressors p from 6 to 15 with increment 1. For each p, the candidate models consist of all subsets of the first p regressors with the first regressor being always included, i.e., we consider all submodels that include the intercept. As a result, the number of models considered here ranges from 2^5 (for $p = 6$) to 2^{14} (for $p = 15$).

In Fig. 1, we depict the median computation time comparison between CD and QP. From the figure, for all R^2 considered, we see the computation times increase at a log-linear rate in p, although the log-linear slope for CD is much smaller than that for QP, which represents a significant advantage of CD compared with QP, especially for a large p. For example, when $p = 15$, the median time for QP is over 1000 s while CD only takes about 100 s. This gap will become even larger as we further increase p. In Fig. 2, we plot the memory cost of QP and CD. As shown, CD requires much less memory than QP, e.g., when $p = 15$, the memory cost of QP is around 30 times that of CD. Note that when $p > 15$, the QP failed to run as its memory requirement exceeds the available memory (16 GB).

In Fig. 3, we present the proportion of zero weights over 1000 repetitions. Let M_m be the set of regressors in model m. Note

[1] $a_p = \Theta(b_p)$ if there exist constants $0 < c < C < \infty$, $cb_p \leq a_p \leq Cb_p$.

[2] Both algorithms produce nearly identical mean squared errors throughout all settings considered.

[3] All numerical experiments were performed on HP Enterprise XL170r with CPU E5-2650v4 (2.20 GHz) and 16 GB memory. The operating system is Unix and the software is MATLAB.

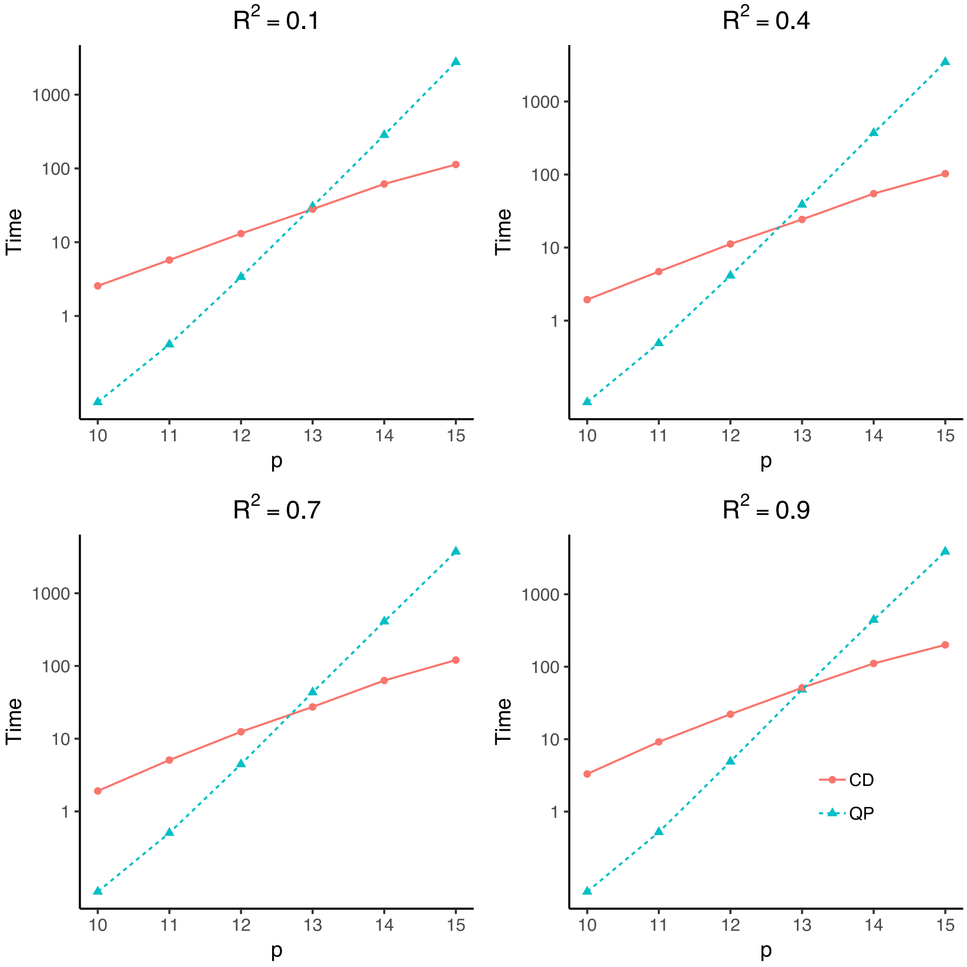

Fig. 1. Comparison of the computation cost (in log seconds) of the CD algorithm vs. QP over 1000 repetitions.[4]

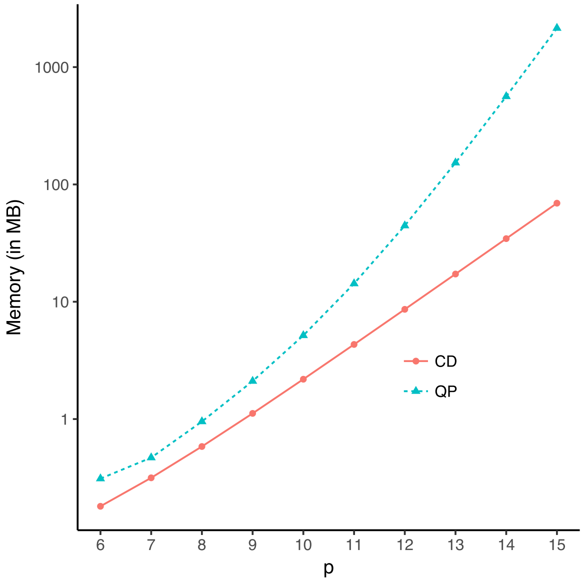

Fig. 2. Comparison of the memory usage (in MB) of the CD algorithm vs. QP for $R^2 = 0.5$.

that the set of regressors in the correct model, denoted by M_p^*, is the set of the first min$(p, 9)$ predictors. We divide all candidate models into the following four types:

- Correct fit (CF): $M_m = M_p^*$.
- Overfit (OF): $M_m \supset M_p^*$. The model contains all informative and some redundant regressors.
- Underfit With Noise (UWN): $M_m \setminus M_p^* \neq \emptyset$ and $M_p^* \setminus M_m \neq \emptyset$. The model does not contain some of the informative regressors but does contain some redundant regressors.
- Underfit Without Noise (UWoN): $M_m \setminus M_p^* = \emptyset$ and $M_p^* \setminus M_m \neq \emptyset$. The model does not contain any redundant regressors but omits some informative regressors.

Note that for $p \leq 9$, we only have CF and UWoN types. It is interesting to observe that as R^2 increases, the proportion of zero weights for the CF model gradually decreases. For $p = 9$, the proportion of zero weights for UWoN models is close to one. For $p = 12$ and 15, the proportions of zero weights for both UWN and UWoN are very close to one, which means the models with positive weights are largely of the CF or OF type. It is worth noting that as p increases, the proportions of zero weights for CF become larger, which is intuitive as we have more models with similar explanative power as the true model. Moreover, the proportions of zero weights for OF are always larger than 0.75, showing that

[4] The time is wall clock time and calculated using the tic/toc functions in Matlab.

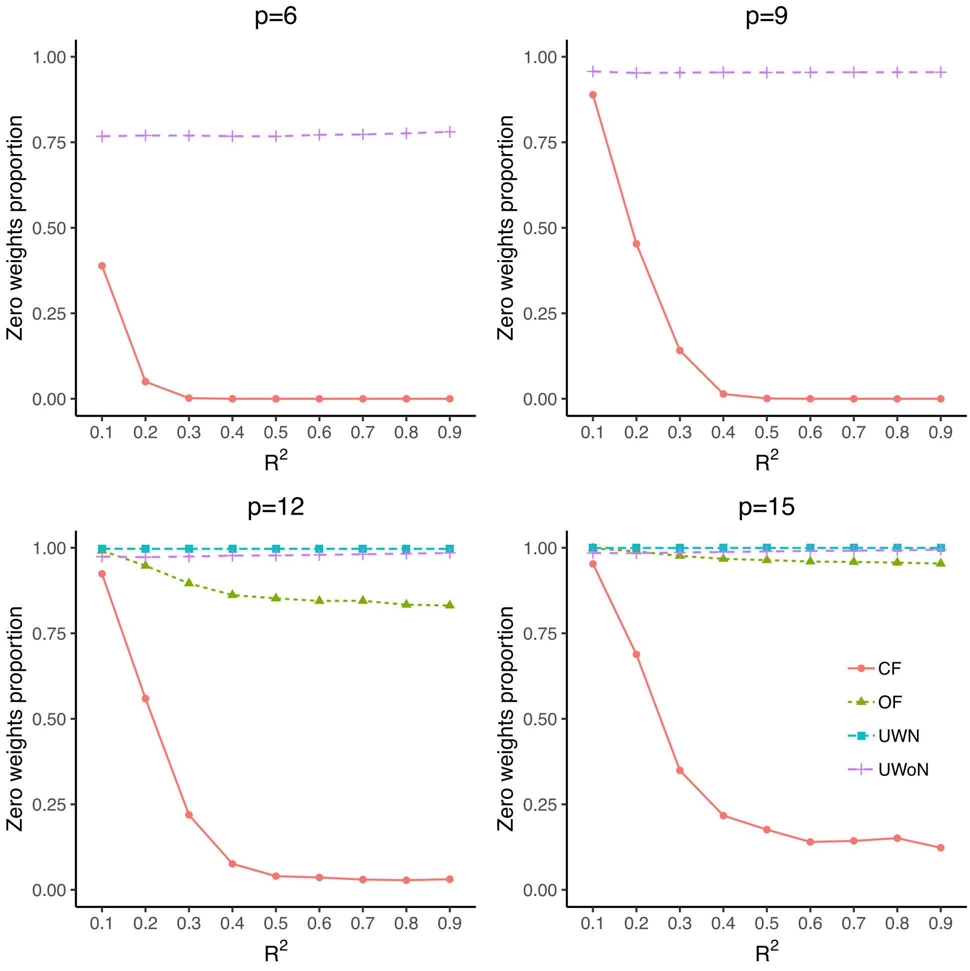

Fig. 3. Average proportion of models that receive zero weights over 1000 repetitions for the CD algorithm.

the weight vector is sparse for not only underfitted models but also for overfitted models.

6. Discussion

Some extensions are worth investigating. First, it would be interesting to pin down the relationship between the sparsity of the weights with the values of the coefficients and the error variance. Second, while this letter focuses on the original form of MMA for simplicity, similar arguments should be applicable to other model averaging procedures such as heteroscedasticity robust C_p model averaging (Liu and Okui, 2013).

References

Bellec, P.C., Lecué, G., Tsybakov, A.B., et al., 2018. Slope meets lasso: improved oracle bounds and optimality. Ann. Statist. 46 (6B), 3603–3642.

Bogdan, M., Van Den Berg, E., Sabatti, C., Su, W., Candès, E.J., 2015. SLOPE - adaptive variable selection via convex optimization. Ann. Appl. Stat. 9 (3), 1103–1140.

Claeskens, G., Hjort, N.L., 2008. Model Selection and Model Averaging. Tech. rep., Cambridge University Press.

Friedman, J., Hastie, T., Tibshirani, R., 2009. Glmnet: lasso and elastic-net regularized generalized linear models. R package version 1 (4).

Hansen, B.E., 2007. Least squares model averaging. Econometrica 75 (4), 1175–1189.

Liu, Q., Okui, R., 2013. Heteroscedasticity-robust C_p model averaging. Econom. J. 16 (3), 463–472.

Moral-Benito, E., 2015. Model averaging in economics: An overview. J. Econ. Surv. 29 (1), 46–75.

Ruszczyński, A.P., 2006. Nonlinear Optimization, Vol. 13. Princeton university press.

Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B Stat. Methodol. 58 (1), 267–288.

Zhang, X., Liu, C.-A., 2019. Inference after model averaging in linear regression models. Econometric Theory 35 (4), 816–841.