Statistical Science
2020, Vol. 35, No. 4, 602–608
https://doi.org/10.1214/20-STS807
Main article: https://doi.org/10.1214/19-STS701, https://doi.org/10.1214/19-STS733

Discussion of "Best Subset, Forward Stepwise or Lasso? Analysis and Recommendations Based on Extensive Comparisons"

Rahul Mazumder

1. INTRODUCTION

I warmly congratulate the authors Hastie, Tibshirani and Tibshirani (HTT); and Bertsimas, Pauphilet and Van Parys (BPV) for their excellent contributions and important perspectives on sparse regression. Due to space constraints, and my greater familiarity with the content and context of HTT (I have had numerous fruitful discussions with the authors regarding their work), I will focus my discussion on the HTT paper.

HTT nicely articulate the relative merits of three canonical estimators in sparse regression: L0, L1 and (forward)-stepwise selection. I am humbled that a premise of their work is an article I wrote with Bertsimas and King [4] (BKM). BKM showed that current Mixed Integer Optimization (MIO) algorithms allow us to compute best subsets solutions for problem instances ($p \approx 1000$ features) much larger than a previous benchmark (software for best subsets in the R package `leaps`) that could only handle instances with $p \approx 30$. HTT by extending and refining the experiments performed by BKM, have helped clarify and deepen our understanding of L0, L1 and stepwise regression. They raise several intriguing questions that perhaps deserve further attention from the wider statistics and optimization communities.

In this commentary, I will focus on some of the key points discussed in HTT, with a bias toward some of the recent work I have been involved in. There is a large and rich body of work in high-dimensional statistics and related optimization techniques that I will not be able to discuss within the limited scope of my commentary.

2. SPARSE REGRESSION: CONVEX AND DISCRETE OPTIMIZATION LENSES

Over the past several years, algorithmic research in sparse statistical learning [12] have been positively influenced by tools and techniques from continuous optimization especially, convex optimization. First- and second-order methods, path-based algorithms (e.g, LARS) have led to useful algorithms for the Lasso with various algorithmic operating characteristics. It is worth emphasizing that the highly efficient and popular toolkit for Lasso: `glmnet`, is *not* an out-of-the-box implementation of cyclical coordinate descent—it has been positively influenced by several years' of research toward understanding the (statistical) structure of Lasso solutions. Interestingly, similar *continuous* optimization methods with suitable modifications lead to good *feasible* (i.e., locally optimal) solutions for nonconvex penalized optimization problems (e.g., with SCAD, MCP penalties) [11, 12, 19].

Discrete optimization techniques such as forward/backward stepwise selection are also used to obtain approximate solutions for best subsets, though their usage is a bit limited compared to continuous optimization based methods. There is a stark difference in computational performances and available software: `glmnet` can usually handle problems with $p \approx 10^5$ and $n \approx 100$ within a second, but the well-known R function `step` only works for $n > p$ regimes and is far less efficient.

MIO, a field within mathematical optimization, provides us with a rich set of algorithmic tools that allow us solve to *optimality*, a family of structured discrete optimization problems: best subsets being one such example. The broader message of BKM is that there is enormous potential and value in using MIO methods to design principled computational tools for sparse learning problems, often perceived to be computationally infeasible.[1] HTT by using MIO-tools concretely demonstrate certain undesirable and less-understood properties of best subsets. This helps us understand, critique existing estimators and propose improvements. While significant strides have been made at the intersection of convex optimization

Rahul Mazumder is the Robert G. James Career Development Professor and Associate Professor of Operations Research and Statistics at the MIT Sloan School of Management. He is also afilliated with the Operations Research Center, MIT Center for Statistics and MIT IBM Watson AI Lab. Address: 100 Main Street, Cambridge-02142, USA (e-mail: rahulmaz@mit.edu).

[1] To be clear, I do not intend to imply that best subsets is a straightforward computational problem that is simple to solve. MIO provides principled tools that allow us to solve to near-optimality, many interesting problem-instances that are often perceived to be practically impossible to compute.

and sparse statistical learning; research at the crossroads of convex optimization, MIO and statistics is rather immature in comparison. HTT raise questions that squarely belong to this intersection.

3. IS BEST SUBSETS THE HOLY GRAIL?

HTT raise an important point regarding whether best subsets should be considered the gold standard for noisy data, even if one desires a sparse model with few nonzero coefficients. Like most statistical estimators, the usefulness of best subsets, Lasso or Stepwise depends upon the context, statistical metric(s) of interest (e.g., prediction, estimation, variable selection, etc.) and associated bias-variance trade-offs (even if one were to ignore computational costs). As HTT point out, no method uniformly dominates the other, and each method is useful in its own way. From a methodological standpoint, it is important to study and create computational tools for all these estimators.

In low signal regimes, HTT observe that best subsets can lead to poor prediction performance (even after validation tuning)—the corresponding solutions generally have fewer nonzeros than the underlying truth (assuming data is generated from a sparse linear model). See also the work by Mazumder, Radchenko and Dedieu [16] for related observations. When the signal is low, the Lasso, which is a relaxation of the best subsets estimator, may lead to better prediction error due to shrinkage.[2] The same shrinkage can hurt Lasso in high signal regimes, when best subsets is a good estimator and the Lasso does not yield a good approximation.

Stepwise regression is a well-known greedy heuristic for best subsets. Depending upon the problem, stepwise regression may or may not lead to a good approximation to best subsets. BKM consider Stepwise only for the $n > p$ setting (they use the R function `step` which does not appear to work for $p > n$); and use AIC-based tuning resulting it to under-perform compared to best subsets. HTT's extended simulations and complementary experiments performed by Hazimeh and Mazumder [13] suggest that in some high signal-to-noise ratio (SNR) cases (e.g., with small n, high feature correlations) best subsets works better than Stepwise in terms of prediction and variable selection. In some other cases, their performances appear to be similar. When the noise is high, Stepwise does not appear to work as well as the Lasso in terms of prediction, perhaps due to a lack of shrinkage.

HTT's observation that Stepwise "searches less" compared to best subsets reminds us of Boosting (e.g., incremental forward stagewise regression, LSBoost) [9, 11], which is known to impart algorithmic regularization. Interestingly, Boosting algorithms with suitable tuning parameter choices, may lead to good predictions (comparable to L1) for low signal regimes. Variants of Boosting can lead to the Lasso solution path, or more aggressive variants such as matching pursuit or forward stepwise regression. While the notion of algorithmic regularization is very pertinent in machine learning applications, it is often perceived to be a bit opaque by some researchers—the estimators cannot be generally expressed as the solution to an optimization criterion like best subsets or the Lasso.

To echo the point raised by HTT: When various researchers view Lasso and Stepwise as heuristics for best subsets, their implicit goal is to focus on regimes where best subsets is a statistically useful estimator. This is an important setting from a methodological viewpoint; and has been studied in high-dimensional statistics and compressed sensing [19]. In particular, this can provide us guidance on when the Lasso, Stepwise and best subsets will be similar and when they might be different.

4. WHAT IS A REALISTIC SIGNAL-TO-NOISE RATIO?

Building on the discussion in HTT (Section 2.2), I would like to add that while SNR is important, it should be interpreted along with other problem parameters (sample size, number of features, feature correlations, etc.). Taken together, they determine what is the *achieved* Proportion of Variance Explained (PVE) by an estimator—in other words, how difficult is the underlying statistical problem? Consider the example in Figure 3 in HTT ($n = 200$, $p = 100$), here a SNR $= 6$ translates to a Lasso-PVE ≈ 0.84 and is quite optimistic as the authors note. However, if we were to consider $n = 20$, $p = 1000$ instead, then SNR $= 6$ would correspond to a Lasso-PVE≈ 0.02, which has very little predictive power.

To settle ideas, suppose data is generated from a sparse linear model $Y = X\beta_0 + \epsilon$ (same setup as Section 3 in HTT). Given an estimator, one may be interested in the following questions:

(i) Can we do full support recovery?
(ii) Can we get a model with estimation error better than the null model?
(iii) Can we get a model with prediction error better than the null model?

Along with SNR, we also need to know n, p, s and Σ to be able to answer the above points. There may be regimes where all of (i)–(iii) are possible; a subset of these are possible, or none are possible (Note that these answers will also depend upon the estimator under consideration). In low signal regimes, when only (iii) is possible, it may still be quite useful to have a sparse model with good prediction performance.

[2]Note that in low-signal regimes, ridge regression may lead to better prediction than L0, L1 and Stepwise [13, 16] even if the underlying model is sparse.

The experiments in HTT and also those in Hazimeh and Mazumder [13] shed light into the above. Figure 1, which is adapted from [13], shows a couple of concrete examples.[3] Since it may be difficult to predict a priori how (i)–(iii) might change as a function of the parameters $(\text{SNR}, n, p, s, \Sigma)$, it is important to have computational tools that facilitate our understanding of these questions.

4.1 High SNR is not that uncommon in applications

HTT make an interesting note regarding what SNR values they have seen in practice:

> *"In our experience, a PVE of 0.5 is rare for noisy observational data, and 0.2 may be more typical. A PVE of 0.86, corresponding to an SNR of 6, is unheard of! With financial returns data, explaining even 2% of the variance (PVE of 0.02) would be considered huge..."*

Some of my colleagues at MIT said that these numbers appear to be somewhat pessimistic in the context of several business analytics applications. To be concrete, in marketing and retail applications, for example, it is not uncommon to see instances where the PVE (or some equivalent) is quite high (70 to 90%+). See, for example, [3, 7, 10, 18]—all involving real-world applications.

It appears that high SNR problems are also seen in compressed sensing, image processing and spectroscopy applications.

In modern image classification and natural language processing tasks—thanks to significant advances in Neural Networks—we often come across test-accuracies of $\sim$ 99%. For example, in a recent work with collaborators from Google Research [15], we observe high predictive performance on 26 classification benchmarks—in all these examples, the AUC ranges from 0.73–1.0.

I agree with HTT that in financial applications, if one uses standard data sources, PVE is usually quite low. However, there appear to be ways to obtain somewhat improved predictions using additional or external data sources. Due to our significantly improved data-collection capabilities, there is an increasing trend to leverage data from multiple modalities, alternative sources (e.g., social media, news, weather, knowledge graphs)—they may result in improved predictions for certain financial indicators—see, for example, [1, 6, 21]. This is an insight I gathered from my ongoing collaboration with researchers at IBM (Financial Services, MIT-IBM Watson AI Lab).

[3] I did a quick survey of papers from different authors appearing in the statistics methodology literature: I have seen examples where SNR is in the high range 25 to 2000+; and also examples with low $\text{SNR} = 0.01 - 0.05$. In almost all these examples with various choices of (n, p, s, Σ), at least (ii) and (iii) seem to be possible using the Lasso.

While I do appreciate HTT's perspective, it seems that a wide range of SNR values occur in practice. Moreover, in many situations, practitioners may not know a priori what the SNR is for the application at hand. It appears to be useful to have a suite of tools that are applicable for both high and low SNR regimes.

5. ESTIMATORS FOR LOW AND HIGH SIGNAL REGIMES

To obtain an estimator that works well for both high and low signal regimes, HTT recommend the following Relaxed Lasso estimator:

$$\text{(1)}\quad \text{Relaxed Lasso:}\quad \hat{\beta}^{\text{relax}} = \gamma \hat{\beta}^{\text{L1}} + (1-\gamma)\hat{\beta}^{\text{L1,LS}},$$

where $\hat{\beta}^{\text{L1}}$ is the Lasso estimate (for a certain tuning parameter λ), the nonzero components of $\hat{\beta}^{\text{L1,LS}}$ are given by the least squares solution[4] on the support of $\hat{\beta}^{\text{L1}}$, and $\gamma \in [0, 1]$ is a second tuning parameter. This is a modification of the original Relaxed Lasso estimator by Meinshausen [17].

5.1 Regularized Subset Selection

Mazumder, Radchenko and Dedieu [16] propose[5] an alternative regularized subset selection procedure or *Regularized-Subsets*, which is given by the following optimization problem:

$$\text{Regularized-Subsets:}\quad \min_{\beta}\ \frac{1}{2}\|Y - X\beta\|_2^2 + \underbrace{\lambda\|\beta\|_q}_{\text{Shrinkage}} \quad \text{s.t.}\ \underbrace{\|\beta\|_0 \le k}_{\text{Selection}}, \tag{2}$$

where, $q \in \{1, 2\}$. Estimator (2) performs a modification to the best subsets criterion, by including an additional continuous regularizer (e.g., Lasso or Ridge[6]). Regularized-Subsets is designed for both low and high signal regimes: When the signal is low, the shrinkage effect of "$\lambda\|\beta\|_q$" attempts to mitigate the overfitting behavior of best subsets. When the signal is high, the continuous regularization term has little to no effect on the estimator; and estimator (2) behaves like best subsets. Criterion (2) allows us to explicitly control the support-size of β. Interestingly, the Relaxed Lasso estimator of Meinshausen is a feasible solution for Problem (2) ($q = 1$) for suitably chosen tuning parameters.

[4] Assuming this exists, as Lasso solutions from `glmnet` can have support sizes larger than n (for $n < p$ settings).

[5] This paper appeared on ArXiv a couple of weeks after the HTT paper was posted.

[6] Ridge involves the squared L2-norm instead of the L2-norm—we use the term interchangeably for simplicity.

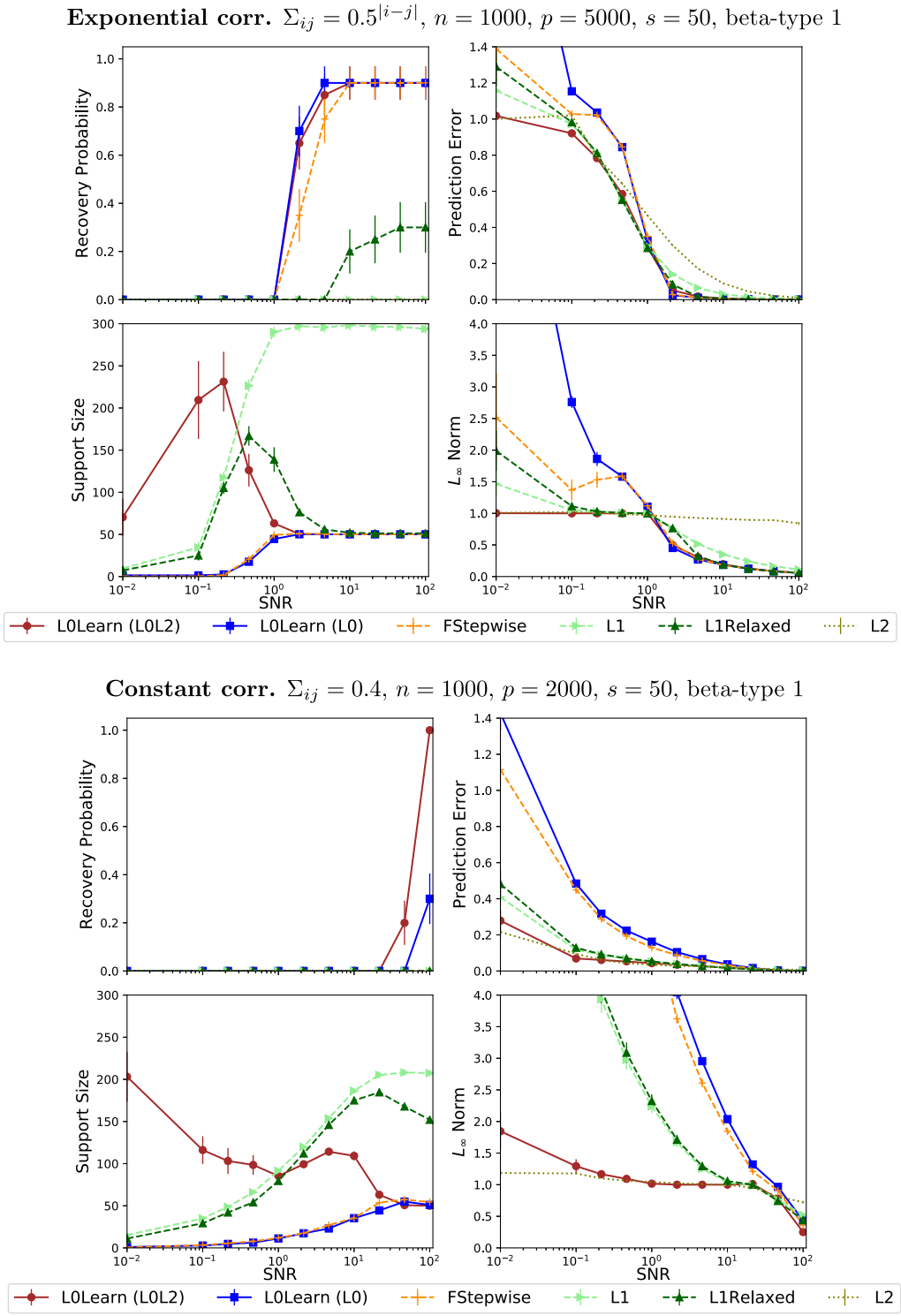

FIG. 1. *Figures adapted from* [13]. *We show recovery probability* (*proportion of times the full support is recovered across all replications*), *support size*, *prediction error*, *estimation error* (*this is the* L_∞ *norm of* $\hat{\beta} - \beta_0$). *We consider two correlation settings* (*top and bottom*) *and vary SNR. True regression coefficients* β_0 *are of type "beta-type* 1*"* (*as in HTT*) *with* s *nonzero coefficients and* Σ *is the covariance matrix of* X. *Here,* `L0Learn` (*L0L2*) *is Regularized-Subsets* ($q = 2$), `L0Learn` (*L0*) *is L0 solution—both computed by* `L0Learn`. *The tuning parameters for all these methods are based on minimizing prediction error on a separate validation set.*

Estimator (2) (for $q = 2$) is also considered in [5] (see also BPV) where they present a robust optimization interpretation for this problem. As discussed in [16], the Regularized-Subsets estimator bears similarities with other estimators that have appeared in the statistics literature.

5.1.1 *Experiments*. Figure 1 shows performances of Regularized-Subsets with $q = 2$ (denoted by L0L2), L0,

L2, L1, Relaxed Lasso and Stepwise—this is a subset of the results appearing in [13]. We vary SNR to see how the estimators fare in terms of prediction, estimation, variable selection (full support recovery and sparsity). We take two correlation models for X: (a) $\Sigma_{ij} = 0.5^{|i-j|}$ (exponential correlation) and (b) $\Sigma_{ij} = 0.4$ (constant correlation). We present a summary of our key findings from these experiments:

- The constant correlation setting is much harder (statistically) compared to the exponential correlation setting. While full support recovery for the exponential correlation setting occurs around SNR≈ 6, this does not happen until SNR$\approx$100 for the constant correlation setting.
- L0 (with no shrinkage) leads to poor predictions even for moderate values of SNR, so does Stepwise. Regularized-Subsets (L0L2) appears to mitigate the poor prediction performance of L0 via L2 shrinkage.
- For both settings, L0L2 seems to deliver better predictive models compared to Relaxed Lasso.
- When SNR is very low: L2 (Ridge) appears to deliver the best prediction performance. L0L2 is very close, and leads to much sparser models. Relaxed Lasso may be sparser than L0L2 but its prediction performance is slightly worse.
- L0 and L0L2 have an edge over Stepwise in terms of full support recovery (high SNR). Lasso, Relaxed Lasso fail to recover the full support in both cases.
- When SNR is very high, all methods have similar prediction error. The main difference is in variable selection performance.

The empirical findings summarized above help clarify the points made in Section 4, on how observed PVE relates to SNR along with other problem parameters (n, p, s, Σ).

5.1.2 *Theory.* In [16], we establish statistical properties of estimator (2), which we denote by $\hat{\beta}_q$. When the signal is high, we show that $\hat{\beta}_q$ has a prediction error bound similar to that of best subsets. When the signal is low, the prediction error of $\hat{\beta}_q$ is shown to be smaller than best subsets. We also show that best subsets can have a smaller prediction error if k is taken to be smaller than the true underlying sparsity. However, this error is shown to be generally worse than that of $\hat{\beta}_2$.

The Relaxed Lasso fit appearing in display (1) can be computed as a simple by-product of the Lasso solution path. Computing solutions to (2) is more challenging—this is addressed in [13, 14, 16]; see also related discussions in BPV. I outline some recent computational approaches below as they relate to the points raised by HTT.

6. COMPUTATIONAL ADVANCES FOR REGULARIZED SUBSET SELECTION

HTT rightly note that the computational cost of best subsets (solved via Gurobi) is substantially larger compared to highly efficient specialized solvers for Lasso and Stepwise:

> *"We note that this corresponds to a computational cost for "regular" practical usage of 30 minutes per value of k: if we wanted to use 10-fold cross-validation to choose between the subset sizes $k = 0, \ldots, 50$, then we are facing 250 hours (> 10 days) of computation time."*

This raises the question: what L0-method should one use for "regular" practical usage?

6.1 An algorithm for "regular" practical usage: `L0Learn`

Recently, Hazimeh and Mazumder [13] propose `L0Learn` to bridge the stark gap in computation times between Lasso, best subsets and common heuristics for best subsets.[7] We consider a family of Regularized-Subsets problems (in the Lagrangian form):

$$\min_{\beta} \ \frac{1}{2}\|Y - X\beta\|_2^2 + \lambda_0\|\beta\|_0 + \lambda_q\|\beta\|_q^q, \tag{3}$$

where, $q \in \{1, 2\}$ and (λ_0, λ_q) are tuning parameters. `L0Learn` draws inspiration from `glmnet`; and is available as an R-package on CRAN. `L0Learn` is based on two complementary algorithms:

(i) cyclic coordinate descent for quickly finding (local) solutions to Problem (3), and
(ii) algorithms based on local combinatorial search, which help improve solutions from (i).

In particular, solutions obtained by (ii) cannot be improved by making small changes to their support. Usually, we observe that (ii) leads to better solutions than (i) when n is small and/or the features are highly correlated. In particular, (ii) can offer key improvements over (i) in terms of variable selection accuracy, at the cost of higher computation times. Our `L0Learn` toolkit is open-source and provides useful insights into what types of algorithms lead to good solutions for regularized subset-selection type problems. This is an important contrast when compared to black-box implementations of sophisticated MIO commercial solvers like Gurobi.

[7] While the Stepwise implementation used in HTT works well for small values of p, it becomes expensive when $p \approx 10^5$ or so. The heuristics presented in BKM will be faster than Stepwise. Both algorithms however will be considerably slower than solving the Lasso via `glmnet`. See [13] for additional details.

TABLE 1

Table showing runtimes (in secs) to compute a path of 100 solutions for different methods on three different datasets. `L0Learn` *(L0Lq),* $q = 1, 2$ *is a solution to (3) as obtained by algorithm* `L0Learn` *(we do not use local search here, see [13] for details). We show test MSE and corresponding number of nonzeros (nnz) for the best model obtained via validation-set tuning. Stepwise regression (HTT) will not run on these problems due to excessive memory requirements. Relaxed Lasso may lead to sparser solutions (at the expense of higher runtimes)*

	Synthetic			Amazon Reviews			US Census		
	$(n = 200, p = 10^6)$			$(n = 2500, p = 175K)$			$(n = 5000, p = 56K)$		
Toolkit	Time(s)	MSE	nnz	Time(s)	$\text{MSE}_{\times 10^2}$	nnz	Time(s)	MSE	nnz
`glmnet` (L1)	22.5	4.55	185	49.4	5.11	256	28.7	61.3	222
`L0Learn` (L0L2)	16.5	4.64	11	31.7	5.18	37	19.6	60.7	15
`L0Learn` (L0L1)	16.7	5.12	15	29.5	5.20	36	16.7	60.8	11
Stepwise	–	–	–	–	–	–	–	–	–

To give an example of the efficiency of `L0Learn`, we present Table 1 (adapted from [13]). Table 1 helps reenforce some of the operating characteristics of Regularized-Subsets: shrinkage is necessary to reduce the overfitting of pure L0. The predictive performances of Regularized-Subsets and Lasso are similar, though the former may lead to higher sparsity. Interestingly, we do observe runtime improvements over Lasso (`glmnet`)—probably because (3) leads to sparser solutions. Note that `L0Learn` enables us to obtain the results in Figure 1 in practical runtimes (the most expensive method in the figure is Stepwise).

6.2 Global optimal solutions

It may be useful to clarify HTT's remarks regarding the long-runtimes of Gurobi's MIO solver for best subsets. Being a general purpose solver, Gurobi may be slow compared to specialized algorithms designed for specific problems (In particular, Gurobi will be generally slower than `glmnet` for solving the Lasso problem). More importantly, MIO-solvers attempt to obtain globally optimal solutions via branch-and-bound—making them operationally *very* different than Stepwise, which is content in delivering a good feasible solution. Devising better algorithms with *global* optimality certificates for Regularized-Subsets is a formidable challenge, requiring algorithmic innovations. This point has also been made in BPV.

Fortunately, since the work of BKM, improved algorithms have been proposed for solving a ridge regularized version of best subsets to *optimality*. A couple of different approaches appear in:

(a) Bertsimas and Van Parys [5] (a cutting plane method using Gurobi) and

(b) Hazimeh, Mazumder and Saab [14] (a stand-alone tailored nonlinear branch-and-bound algorithm).

A cutting plane method: Bertsimas and Van Parys [5] present an impressive cutting plane approach that uses Gurobi to solve a sequence of mixed integer linear problems; and can solve instances with $n \approx p \sim 10^5$ (these instances usually have low correlations in X and a generous ridge parameter). See BPV for additional discussions.

A tailored nonlinear branch-and-bound method: In a different line of work, Hazimeh, Mazumder and Saab [14] develop a stand-alone solver: `L0BnB`, written from scratch, that does not rely on off-the-shelf solvers like Gurobi. This seems to be the first work where first order methods (we use coordinate descent) are used within a branch-and-bound framework to solve the sparse regression problem to *optimality*. In some instances, our algorithm [14] exhibits speed-ups $> 3600\times$ compared to state-of-the-art MIO methods, in obtaining optimal solutions. The framework of [14] can solve, to optimality, a real data instance with $p \sim 10^7$ with small values of n and k within a few minutes. This provides encouraging preliminary evidence toward creating scalable discrete optimization solvers, leveraging our current understanding of first-order convex optimization methods.

An interesting line of work in optimization (see, e.g., [2, 8, 14, 20]) advocates the use of stronger formulations[8] for best subsets type problems—these ideas can potentially lead to improved algorithms and interesting new perspectives for sparse regression in the future.

ACKNOWLEDGMENTS

I would like to thank the Associate Editor and Cun-Hui Zhang for providing me the opportunity to discuss the interesting paper by HTT. A special thanks to Hussein Hazimeh, Peter Radchenko and Subhabrata Sen for helpful feedback on the manuscript. Thanks also to Dimitris Bertsimas, Robert Freund, David Gamarnik, Edward George,

[8]Loosely speaking, these formulations lead to relaxations that lead to integral solutions; and hence the overall branch-and-bound method can converge faster. In some cases, these can be better than the basic BigM formulation appearing in BKM (the approach used by HTT).

Chris Hansen, Xihong Lin, Arian Maleki, Georgia Perakis, Rama Ramakrishnan, Richard Samworth, Matthew Stephens and Cun-Hui Zhang for helpful discussions.

The author acknowledges research support from the Office of Naval Research ONR-N000141812298 (Young Investigator Award), the National Science Foundation (NSF-IIS-1718258), MIT IBM AI Watson Lab and Liberty Mutual Insurance.

REFERENCES

[1] ASUR, S. and HUBERMAN, B. A. (2010). Predicting the future with social media. In 2010 *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology* **1** 492–499. IEEE, New York.

[2] ATAMTURK, A. and GOMEZ, A. (2019). Rank-one convexification for sparse regression. Preprint. Available at arXiv:1901.10334.

[3] BAARDMAN, L., COHEN, M. C., PANCHAMGAM, K., PERAKIS, G. and SEGEV, D. (2019). Scheduling promotion vehicles to boost profits. *Manage. Sci.* **65** 50–70.

[4] BERTSIMAS, D., KING, A. and MAZUMDER, R. (2016). Best subset selection via a modern optimization lens. *Ann. Statist.* **44** 813–852. MR3476618 https://doi.org/10.1214/15-AOS1388

[5] BERTSIMAS, D. and VAN PARYS, B. (2020). Sparse high-dimensional regression: Exact scalable algorithms and phase transitions. *Ann. Statist.* **48** 300–323. MR4065163 https://doi.org/10.1214/18-AOS1804

[6] BOLLEN, J., MAO, H. and ZENG, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science* **2** 1–8.

[7] COHEN, M. C., LEUNG, N.-H. Z., PANCHAMGAM, K., PERAKIS, G. and SMITH, A. (2017). The impact of linear optimization on promotion planning. *Oper. Res.* **65** 446–468. MR3647851 https://doi.org/10.1287/opre.2016.1573

[8] DONG, H., CHEN, K. and LINDEROTH, J. (2015). Regularization vs. Relaxation: A conic optimization perspective of statistical variable selection. ArXiv e-prints.

[9] FREUND, R. M., GRIGAS, P. and MAZUMDER, R. (2017). A new perspective on boosting in linear regression via subgradient optimization and relatives. *Ann. Statist.* **45** 2328–2364. MR3737894 https://doi.org/10.1214/16-AOS1505

[10] GHOSE, A. and IPEIROTIS, P. G. (2010). Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE Trans. Knowl. Data Eng.* **23** 1498–1512.

[11] HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J. (2009). *The Elements of Statistical Learning*: *Data Mining*, *Inference*, *and Prediction*, 2nd ed. *Springer Series in Statistics*. Springer, New York. MR2722294 https://doi.org/10.1007/978-0-387-84858-7

[12] HASTIE, T., TIBSHIRANI, R. and WAINWRIGHT, M. (2015). *Statistical Learning with Sparsity*: *The Lasso and Generalizations*. *Monographs on Statistics and Applied Probability* **143**. CRC Press, Boca Raton, FL. MR3616141

[13] HAZIMEH, H. and MAZUMDER, R. (2020). Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms. *Oper. Res.* **68** 1285–1624.

[14] HAZIMEH, H., MAZUMDER, R. and SAAB, A. (2020). Sparse regression at scale: Branch-and-bound rooted in first-order optimization. Preprint. Available at arXiv:2004.06152.

[15] HAZIMEH, H., PONOMAREVA, N., MOL, P., TAN, Z. and MAZUMDER, R. (2020). The tree ensemble layer: Differentiability meets conditional computation. In *Proceedings of the 37th Annual International Conference on Machine Learning, ICML. Vol. 20.*

[16] MAZUMDER, R., RADCHENKO, P. and DEDIEU, A. (2017). Subset selection with shrinkage: Sparse linear modeling when the snr is low. Preprint. Available at arXiv:1708.03288.

[17] MEINSHAUSEN, N. (2007). Relaxed Lasso. *Comput. Statist. Data Anal.* **52** 374–393. MR2409990 https://doi.org/10.1016/j.csda.2006.12.019

[18] SMITH, S. A. and ACHABAL, D. D. (1998). Clearance pricing and inventory policies for retail chains. *Manage. Sci.* **44** 285–300.

[19] WAINWRIGHT, M. J. (2019). *High-Dimensional Statistics*: *A Non-asymptotic Viewpoint*. *Cambridge Series in Statistical and Probabilistic Mathematics* **48**. Cambridge Univ. Press, Cambridge. MR3967104 https://doi.org/10.1017/9781108627771

[20] XIE, W. and DENG, X. (2020). Scalable algorithms for the sparse ridge regression.

[21] ZHOU, D., ZHENG, L., ZHU, Y., LI, J. and HE, J. (2020). Domain adaptive multi-modality neural attention network for financial forecasting. In *Proceedings of the Web Conference* 2020 2230–2240.