ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9415082

ON THE OPTIMALITY OF BACKWARD REGRESSION: SPARSE RECOVERY AND SUBSET SELECTION

Sebastian Ament and *Carla Gomes*

Cornell University
Department of Computer Science
Ithaca, NY 14850

ABSTRACT

Sparse recovery and subset selection are fundamental problems in varied communities, including signal processing, statistics and machine learning. Herein, we focus on an important greedy algorithm for these problems: Backward Stepwise Regression. We present novel guarantees for the algorithm, propose an efficient, numerically stable implementation, and put forth Stepwise Regression with Replacement (SRR), a new family of two-stage algorithms that employs both forward and backward steps for compressed sensing problems. Prior work on the backward algorithm has proven its optimality for the subset selection problem, provided the residual associated with the optimal solution is small enough. However, the existing bounds on the residual magnitude are NP-hard to compute. In contrast, our main theoretical result includes a bound that can be computed in polynomial time, depends chiefly on the smallest singular value of the matrix, and also extends to the method of magnitude pruning. In addition, we report numerical experiments highlighting crucial differences between forward and backward greedy algorithms and compare SRR against popular two-stage algorithms for compressed sensing. Remarkably, SRR algorithms generally maintain good sparse recovery performance on coherent dictionaries. Further, a particular SRR algorithm has an edge over Subspace Pursuit.

Index Terms— Sparse Recovery, Subset Selection, Compressed Sensing, Matching Pursuit, Stepwise Regression

1. INTRODUCTION

Sparse signal recovery and subset selection are problems with important applications in medicine [1, 2], materials science [3], and engineering [4]. Given a set of *atoms* $\mathcal{D} \stackrel{\text{def}}{=} \{\boldsymbol{\varphi}_i\}$, also referred to as a dictionary, both problems are about identifying a sparse subset of $\mathcal{D}$ that most accurately model an observation $\mathbf{y}$, but the assumptions vary. Formally, the central problem is the solution of the following expression with a desired sparsity:

$$\min_{|\mathcal{A}|=k} \min_{\mathbf{x}} \|\mathbf{y} - \sum_{i\in\mathcal{A}} x_i \boldsymbol{\varphi}_i\|_2 \tag{1.1}$$

If $\mathbf{y} = \boldsymbol{\Phi}\mathbf{x} + \boldsymbol{\epsilon}$ where $\boldsymbol{\Phi} \stackrel{\text{def}}{=} [\boldsymbol{\varphi}_1, ..., \boldsymbol{\varphi}_m]$, $\mathbf{x}$ has at most k non-zero entries, and $\boldsymbol{\epsilon}$ is a small perturbation, the problem is referred to as sparse recovery. On the other hand, if $\mathbf{y}$ is an arbitrary vector, not necessarily associated with a sparse $\mathbf{x}$, the problem is called sparse approximation or subset selection. Notably, the recent work of [5] generalizes the framework of sparse recovery for the sparsification of neural networks with promising results.

Because of the importance and ubiquity of the problems, many approaches have been proposed in the literature [6–14]. Two popular greedy algorithms are Forward and Backward Regression, which have been proposed repeatedly in different fields, leading to a perplexing number of names that all refer to the same two algorithms.

In particular, the forward algorithm is also known as Forward Regression, Forward Selection [15], Optimized Orthogonal Matching Pursuit (OOMP) [16], Order-recursive Matching Pursuit (ORMP) [17], and Orthogonal Least-Squares (OLS) [18–20]. The backward algorithm is also known as Backward Regression, Backward Elimination [15], Backward Optimized Orthogonal Matching Pursuit (BOOMP) [21], and the backward greedy algorithm [22]. The two algorithms add atoms to, respectively delete atoms from, an *active set* $\mathcal{A}$, based on the greedy heuristics

$$\arg\min_{i\notin\mathcal{A}} \|\mathbf{r}_{\mathcal{A}\cup i}\|_2, \qquad \text{and} \qquad \arg\min_{i\in\mathcal{A}} \|\mathbf{r}_{\mathcal{A}\setminus i}\|_2, \tag{1.2}$$

where $\mathbf{r}_{\mathcal{A}} = (\mathbf{I} - \boldsymbol{\Phi}_{\mathcal{A}}\boldsymbol{\Phi}_{\mathcal{A}}^{+})\mathbf{y}$ is the least-squares residual associated with the subset of atoms indexed by $\mathcal{A}$. Among all additions and deletions of a single atom, these heuristics leads to the minimum residual in the following iteration. By comparison, the well-known Orthogonal Matching Pursuit (OMP) algorithm adds atoms based on the rule

$$\arg\max_{i\notin\mathcal{A}} |\langle \boldsymbol{\varphi}_i, \mathbf{r}_{\mathcal{A}}\rangle|, \tag{1.3}$$

This research was supported by NSF awards CCF-1522054 (Expeditions in computing) and AFOSR Multidisciplinary University Research Initiatives (MURI) Program FA9550-18-1-0136, ARO award W911NF-17-1-0187 for our compute cluster, US DOE Award No. DE-SC0020383, and an award from the Toyota Research Institute.

which is equivalent to the largest component in the gradient of the least-squares objective at the current residual, $\arg\max_{i\notin\mathcal{A}} |\partial_{x_i}\|\mathbf{r}_{\mathcal{A}}\||$. In this sense, the stepwise algorithms "look ahead" further than OMP.

Herein, our primary focus is the backward algorithm, for which we propose an efficient and numerically stable implementation and derive novel theoretical insights which also extend to magnitude pruning. An important limitation of the algorithm is that it only works with matrices $\mathbf{\Phi}$ that have full column rank. Perhaps because of this limitation and the high dimensionality of modern datasets, recent focus has been primarily on forward algorithms. To overcome this limitation and take advantage of the strong theoretical guarantees of the backward algorithm, we further propose Stepwise Regression with Replacement, a new family of two-stage algorithms that uses both forward and backward steps as building blocks to solve challenging compressed sensing problems efficiently.

2. PRIOR WORK

We begin by reviewing the most relevant results on the optimality of the backward algorithm and make connections to other work whenever it becomes relevant. A central concept to the existing theory is the bisector of two linear spaces.

Definition 2.1 (Bisector)**.** *Let A and B be two subspaces of a normed linear space L with metric d. Their bisector is*

$$\mathcal{H}(A,B) \stackrel{\text{def}}{=} \{\mathbf{x}\in L | d(\mathbf{x},A) = d(\mathbf{x},B)\},$$

where $d(\mathbf{x},S) \stackrel{\text{def}}{=} \min_{\mathbf{s}\in S} d(\mathbf{x},\mathbf{s})$ for any $S\subset L$.

The proof of the following result, due to Covreur and Bresler [22], views certain bisectors as decision boundaries of the backward algorithm, and shows that, as long as the boundaries cannot be crossed due to the perturbation, the algorithm succeeds in recovering a sparse signal.

Theorem 2.2 (Covreur and Bresler [22])**.** *Backward Regression with input $\mathbf{\Phi}\mathbf{x}+\boldsymbol{\epsilon}$ recovers the support $\mathcal{S}$ of a k-sparse m-dimensional vector $\mathbf{x}$ in $m-k$ iterations provided*

$$\min_{k<r\le n}\min_{|\mathcal{A}|=r}\min_{i\in\mathcal{A},j\notin\mathcal{A}} d(\mathbf{y},\mathcal{H}(\mathrm{col}(\mathbf{\Phi}_{\mathcal{A}\setminus i}),\mathrm{col}(\mathbf{\Phi}_{\mathcal{A}\setminus j}))) > \|\boldsymbol{\epsilon}\|_2,$$

where $\mathrm{col}(\mathbf{\Phi})$ denotes the column space of $\mathbf{\Phi}$.

The authors postulate that the bound in Theorem 2.2 is NP-hard to compute and therefore of limited practical use. Further, Covreur and Bresler proved that the backward algorithm is not only capable of recovering the support of an *exactly* sparse vector, but that it solves the subset selection problem optimally, provided the residual of the optimal solution is small enough. In particular, for an arbitrary $\mathbf{y}$, not generally associated with a sparse $\mathbf{x}$, the following result holds.

Corollary 2.3 (Covreur and Bresler [22])**.** *Let $\mathbf{x}_k$ be the solution to the subset selection problem* (1.1) *with sparsity k. If the residual $\mathbf{r}_k \stackrel{\text{def}}{=} \mathbf{y}-\mathbf{\Phi}\mathbf{x}_k$ satisfies the bound in Theorem 2.2 in place of $\boldsymbol{\epsilon}$, the backward algorithm recovers $\mathbf{x}_k$.*

Algorithm 1: Efficient Backward Regression

Data: Matrix $\mathbf{\Phi}\in\mathbb{C}^{n\times m}$, target $\mathbf{y}$, desired sparsity k
Result: Support set $\mathcal{A}$

1 $\mathcal{A}\leftarrow[\![1,m]\!]$
2 $\mathbf{Q}_{\mathcal{A}},\mathbf{R}_{\mathcal{A}}\leftarrow \mathrm{qr}(\mathbf{\Phi}_{\mathcal{A}})$
3 **while** $|\mathcal{A}|>k$ **do**
4 $\quad\mathbf{x}_{\mathcal{A}}\leftarrow \mathbf{R}_{\mathcal{A}}^{-1}\mathbf{Q}_{\mathcal{A}}^{*}\mathbf{y}$
5 $\quad\boldsymbol{\gamma}_{\mathcal{A}}\leftarrow \mathrm{diag}[(\mathbf{R}_{\mathcal{A}}^{*}\mathbf{R}_{\mathcal{A}})^{-1}]$
6 $\quad i^{*}\leftarrow \arg\min_{i\in\mathcal{A}}|x_i|^2/\gamma_i$
7 $\quad\mathbf{Q}_{\mathcal{A}\setminus i^*},\mathbf{R}_{\mathcal{A}\setminus i^*}\leftarrow$ remove_column$(\mathbf{Q}_{\mathcal{A}},\mathbf{R}_{\mathcal{A}},i^*)$
8 $\quad\mathcal{A}\leftarrow\mathcal{A}\setminus i^*$
9 **end**

3. MAIN RESULTS

We first propose an efficient and numerically-stable procedure for the evaluation of the column deletion criterion. Subsequently, we present our main theoretical results for the backward algorithm. In the following, $\mathbf{\Phi}$ is a $n\times m$ matrix.

3.1. An Efficient and Stable Implementation

Previously, [23] proposed an efficient implementation for the backward algorithm based on rank-one updates to the normal equations, which are notorious for exacerbating ill-conditioning. Since the algorithm uses recursive updates, small errors are prone to blow up catastrophically. To make the backward algorithm usable for larger, potentially ill-conditioned systems, we require an implementation that is both efficient and numerically stable. To this end, let

$$\boldsymbol{\gamma}_{\mathcal{A}} \stackrel{\text{def}}{=} \mathrm{diag}\left([\mathbf{\Phi}_{\mathcal{A}}^{*}\mathbf{\Phi}_{\mathcal{A}}]^{-1}\right).$$

Using the analysis of [23], it can be proven that for $i\in\mathcal{A}$,

$$\|\mathbf{r}_{\mathcal{A}\setminus i}\|_2^2 - \|\mathbf{r}_{\mathcal{A}}\|_2^2 = |x_i|^2/\gamma_i. \tag{3.1}$$

Therefore, we can efficiently evaluate the deletion criterion in (1.2) by computing $\boldsymbol{\gamma}$ and the coefficients $\mathbf{x}$ corresponding to the least-squares solution given $\mathbf{\Phi}_{\mathcal{A}}$. But rather than computing $\boldsymbol{\gamma}$ and $\mathbf{x}$ via the normal equations, we keep and update a QR factorization of $\mathbf{\Phi}_{\mathcal{A}}$, and set $\boldsymbol{\gamma}\leftarrow\mathrm{diag}[(\mathbf{R}^{*}\mathbf{R})^{-1}]$. While the computation of $\boldsymbol{\gamma}$ is potentially still unstable, the instability cannot propagate through subsequent iterations. See Algorithm 1.

Further, equation (3.1) exposes a similarity of the backward criterion to *magnitude pruning*, a principle used by Subspace Pursuit [8] for its deletion stage, and popularly applied for the sparsification of deep neural networks [24–26]. Inspired by this observation, we also study a modification of Algorithm 1 where line 6 is replaced with

$$\arg\min_{i\in\mathcal{A}}|x_i|.$$

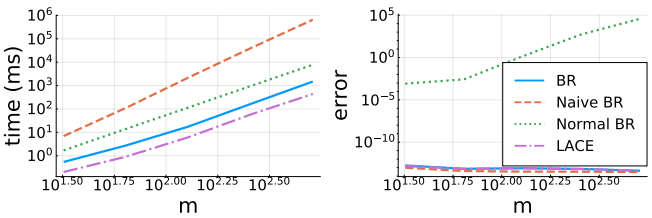

Fig. 3.1. Runtime (left) and error (right) as a function of m for ill-conditioned matrices with $k = 16$, no noise, and $n = m$.

We refer to the resulting algorithm as Least-Absolute Coefficient Elimination (LACE). We now highlight the advantages of the proposed implementation.

The left of Figure 3.1 shows representative runtimes of four implemenations: Algorithm 1 (BR), a naive implementation which calculates $\mathbf{r}_{\mathcal{A}\setminus i}$ for all $i \in \mathcal{A}$ by removing a column of the QR factorization and solving the resulting system (Naive BR), the efficient implementation of Reeves [23] (Normal BR), and LACE. We fixed the sparsity at $k = 16$ and increased m while keeping the matrices square. Notably, BR achieves the same scaling as LACE and is only a small constant factor slower, while the naive implementation is orders of magnitude slower than the other ones. The right of Figure 3.1 shows the error of the solution for the same algorithms on randomly generated matrices with a high condition number. As we did not add any noise, the error should mathematically be zero. But as predicted, the result of Normal BR blows up as the number of iterations of the algorithm grows with m. Consequently, Algorithm 1 is preferable for its efficiency and stability.

3.2. Theoretical Guarantees

The following is our main theoretical result, which provides a sparse recovery guarantee for both backward algorithms.

Theorem 3.1. *Given $\mathbf{\Phi}$ with full column rank, Backward Regression and LACE recover the support $\mathcal{S}$ of a k-sparse m-dimensional vector $\mathbf{x}$ in $m - k$ steps given $\boldsymbol{\epsilon}$ satisfies*

$$\frac{\sigma_{\min}(\mathbf{\Phi})}{2} \min_{i \in \mathcal{S}} |x_i| > \|\boldsymbol{\epsilon}\|_2,$$

where $\sigma_{\min}(\mathbf{\Phi})$ is the smallest singular value of $\mathbf{\Phi}$.

Theorem 3.1 reveals the direct dependence of the tolerable noise on the smallest singular value of the matrix, something which Theorem 2.2 obscures and was only stated heuristically in [22]. The following corollary is analogous to Corollary 2.3 but is based on Theorem 3.1 and extends to LACE.

Corollary 3.2. *Let $\mathbf{x}_k$ be the solution to the subset selection problem* (1.1) *with sparsity k. If the associated residual $\mathbf{r}_k \stackrel{def}{=} \mathbf{y} - \mathbf{\Phi}\mathbf{x}_k$ satisfies the bound in Theorem 3.1 in place of $\boldsymbol{\epsilon}$, Backward Regression and LACE recover $\mathbf{x}_k$.*

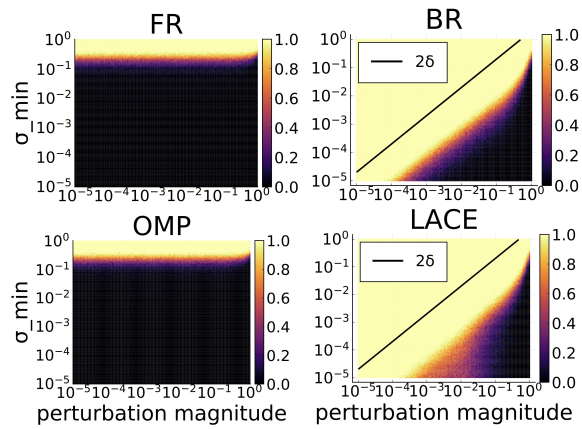

Fig. 3.2. Empirical frequency of support recovery as a function of the smallest singular value $\sigma_{\min}$ and perturbation magnitude for 64 by 64 matrices with signal sparsity 32.

While Corollary 3.2 cannot guarantee the success of the algorithm a-priori, it can be used as an efficient post-hoc check of the algorithms' return values. Alternatively, the algorithms could be stopped adaptively, once the residual surpasses the bound to guarantee that the returned value is optimal among all signals with the same number of non-zero elements.

Notably, a necessary condition for OMP and Forward Regression (FR) to recover the support of a sparse signal is $\mu_1(k) < 1/2$, where μ_1 is the Babel function of the matrix $\mathbf{\Phi}$, whose columns are assumed to be l_2-normalized [7, 27]. This implies $\sigma_{\min}(\mathbf{\Phi}_{\mathcal{A}}) > 1/\sqrt{2} \approx 0.71$ since for any singular value σ of $\mathbf{\Phi}_{\mathcal{A}}$, $|1 - \sigma^2| \leq \mu_1(k)$ for any set $\mathcal{A}$ with cardinality k [7]. Therefore, the forward algorithms are not guaranteed to recover a sparse support for even moderately ill-conditioned systems, regardless of how small the noise is. This is in stark contrast to Theorem 3.1 for the backward algorithms, which are always guaranteed to return the correct support of a sparse vector if the noise is small enough.

3.3. Empirical Evaluation

Figure 3.2 shows the empirical frequency of support recovery for Forward Regression (FR), BR, OMP, and LACE for 64 by 64 matrices as a function of perturbation magnitude $\delta = \|\boldsymbol{\epsilon}\|$ and the smallest singular value $\sigma_{\min}$ of $\mathbf{\Phi}$. For a given $\sigma_{\min}$, we define $\mathbf{S}_\sigma$ as the matrix with uniformly spaced values between $\sigma_{\min}$ and 1 on the diagonal, and let $\mathbf{\Phi}_\sigma = \mathbf{U}\mathbf{S}_\sigma\mathbf{V}^*$, where $\mathbf{U}, \mathbf{V}$ are two orthonormal matrices. The perturbation vectors are uniformly distributed on the δ-hypersphere, and x_i for $i \in \mathcal{S}$ has the Rademacher distribution. Each cell of the heat-maps is an average of 128 independent experiments.

Remarkably, the recovery performance of both BR and LACE exhibits a scaling that is approximately linear in δ, as predicted by Theorem 3.1, and surpasses the forward algorithms on matrices with small singular values. While the

Algorithm 2: Stepwise Regression with Replacement

Data: dictionary $\mathbf{\Phi}$, target $\mathbf{y}$, sparsity k, steps s
Result: Support set $\mathcal{A}$

$\mathcal{A} \leftarrow k$ column indices with largest correlation with $\mathbf{y}$
while *not converged* **do**
  for s *iterations* **do**
    $\mathcal{A} \leftarrow \mathcal{A} \cup \arg\min_{i \notin \mathcal{A}} \|\mathbf{r}_{\mathcal{A} \cup i}\|_2$ ▷ acquisition
  end
  for s *iterations* **do**
    $\mathcal{A} \leftarrow \mathcal{A} \setminus \arg\min_{i \notin \mathcal{A}} \|\mathbf{r}_{\mathcal{A} \setminus i}\|_2$ ▷ deletion
  end
end

forward algorithms only succeed on well-conditioned systems, as suggested by the theory [7, 27], notably, [28] proved that both FR and OMP are approximation algorithms to the subset selection problem, using a notion of approximate submodularity of the coefficient of determination, R^2. Indeed, additional experiments not reported herein show that the forward algorithms' performance in terms of R^2 degrades more gracefully with the smallest singular value, even as the support recovery performance deteriorates sharply.

4. STEPWISE REGRESSION WITH REPLACEMENT

4.1. Motivation and Algorithm Design

While the results presented above are powerful and do not require the same restrictive assumptions as existing forward and two-stage algorithms [7–9], the backward algorithms require the matrices to have full column rank. Thus, they are not immediately applicable for compressed sensing. To remedy this limitation, we propose Stepwise Regression with Replacement (SRR), a new family of two-stage algorithms which combines forward and backward steps to replace atoms in an active set. See Algorithm 2 for schematic code of SRR.

Two-stage algorithms generally keep track of an active set with a desired sparsity k, and improve the reconstruction of the observed signal by iteratively adding and deleting atoms from the active set. For example, both Subspace Pursuit (SP) [8] and Orthogonal Matching Pursuit with Replacement (OMPR) [29] initialize an active set of size k and subsequently alternate adding atoms to the active set based on the largest correlations with the residual, and deleting atoms with magnitude pruning. SP solves a least-squares problem for the expanded active set before deleting atoms, while OMPR only takes one gradient step. Further, SP adds and deletes k atoms, while OMPR adds and deletes s atoms, where s is passed as a parameter.

Stepwise Regression with Replacement has a similar basic structure, and also takes an additional "step" parameter s. However, it uses the heuristics in equation (1.2) to update the support, see Algorithm 2. SRR can be implemented efficiently with updates to a QR-factorization, similar to Algorithm 1.

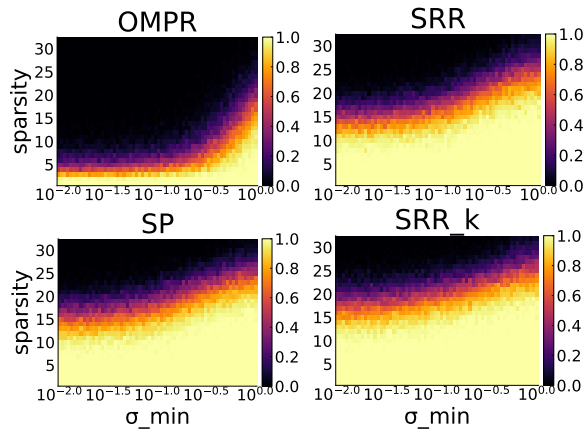

Fig. 4.1. Empirical frequency of support recovery as a function of $\sigma_{\min}$ and sparsity k for 64 by 128 matrices.

4.2. Empirical Evaluation

Figure 4.2 shows the empirical frequency of support recovery for the two-stage algorithms as a function of sparsity level and the smallest singular value $\sigma_{\min}$ for 64 by 128 matrices. We generated the problems in a similar fashion as for Figure 3.2. OMPR and SRR refer to $s = 1$, while SRR_k has $s = k$. Among the single step methods, SRR maintains much better performance on matrices with smaller $\sigma_{\min}$ than OMPR. Further, SRR is marginally less performant than SP, since SP adds and deletes k atoms, but is much more efficient than SP, as it requires far fewer updates - 2 versus $2k$ - to the QR factorization, which is the most expensive operation. The most performant method is SRR_k with a small but statistically significant edge over SP for matrices with smaller $\sigma_{\min}$. Since SP and SRR require a similar effort to implement, the SRR family provides a performant and elegant alternative to SP with a potential for higher computational efficiency by controlling the size of the support updates.

5. CONCLUSION

We derived novel guarantees for Backward Regression, which are efficiently computable, insightful, and experimentally verifiable. We put forth a new implementation of the backward algorithm that is both efficient and numerically stable. Our analysis of LACE connects the results to existing algorithms, like Subspace Pursuit, and is also related to magnitude pruning, a principle used to sparsify neural networks. Inspired by our theoretical results, we proposed a new family of two-stage algorithms for compressed sensing problems: Stepwise Regression with Replacement (SRR). Remarkably, SRR maintains high performance on coherent dictionaries, is efficient, and simple to implement. Lastly, the results herein can help guide users of Backward Regression in professional statistical software, like SAS, and inspire new research and applications.

6. REFERENCES

[1] J. Wan, Z. Zhang, J. Yan, T. Li, B. D. Rao, S. Fang, S. Kim, S. L. Risacher, A. J. Saykin, and L. Shen, "Sparse bayesian multi-task learning for predicting cognitive outcomes from neuroimaging measures in alzheimer's disease," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 940–947.

[2] Z. Zhang, T. Jung, S. Makeig, and B. D. Rao, "Compressed sensing for energy-efficient wireless telemonitoring of noninvasive fetal ecg via block sparse bayesian learning," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 2, pp. 300–309, 2013.

[3] Luca M Ghiringhelli, Jan Vybiral, Emre Ahmetcik, Runhai Ouyang, Sergey V Levchenko, Claudia Draxl, and Matthias Scheffler, "Learning physical descriptors for materials science by compressed sensing," *New Journal of Physics*, vol. 19, no. 2, pp. 023017, 2017.

[4] Wai Lam Chan, Kriti Charan, Dharmpal Takhar, Kevin F Kelly, Richard G Baraniuk, and Daniel M Mittleman, "A single-pixel terahertz imaging system based on compressed sensing," *Applied Physics Letters*, vol. 93, no. 12, pp. 121105, 2008.

[5] C. Lee, I. Fedorov, B. D. Rao, and H. Garudadri, "Ssgd: Sparsity-promoting stochastic gradient descent algorithm for unbiased dnn pruning," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 5410–5414.

[6] Emmanuel J Candès, Justin Romberg, and Terence Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on information theory*, vol. 52, no. 2, pp. 489–509, 2006.

[7] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[8] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2230–2249, 2009.

[9] Deanna Needell and Joel Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Communications of the ACM*, vol. 53, 12 2010.

[10] David P. Wipf and Srikantan S. Nagarajan, "A new view of automatic relevance determination," in *Advances in Neural Information Processing Systems 20*, J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, Eds., pp. 1625–1632. Curran Associates, Inc., 2008.

[11] T. Blumensath and M. E. Davies, "Gradient pursuits," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2370–2382, 2008.

[12] Thomas Blumensath and Mike E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265 – 274, 2009.

[13] Hao He, Bo Xin, Satoshi Ikehata, and David Wipf, "From bayesian sparsity to gated recurrent nets," in *Advances in Neural Information Processing Systems*, 2017, pp. 5554–5564.

[14] Michael E Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of machine learning research*, vol. 1, no. Jun, pp. 211–244, 2001.

[15] Alan Miller, *Subset selection in regression*, CRC Press, 2002.

[16] Laura Rebollo-Neira and David Lowe, "Optimized orthogonal matching pursuit approach," *IEEE signal processing Letters*, vol. 9, no. 4, pp. 137–140, 2002.

[17] S. F. Cotter, R. Adler, R. D. Rao, and K. Kreutz-Delgado, "Forward sequential algorithms for best basis selection," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 146, no. 5, pp. 235–244, 1999.

[18] Thomas Blumensath and Mike E Davies, "On the difference between orthogonal matching pursuit and orthogonal least squares," Available at `https://eprints.soton.ac.uk/142469/1/BDOMPvsOLS07.pdf`, 2007 (Accessed 09/2020).

[19] Sheng Chen, Stephen A Billings, and Wan Luo, "Orthogonal least squares methods and their application to non-linear system identification," *International Journal of control*, vol. 50, no. 5, pp. 1873–1896, 1989.

[20] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Transactions on Neural Networks*, vol. 2, no. 2, pp. 302–309, 1991.

[21] M. Andrle, L. Rebollo-Neira, and E. Sagianos, "Backward-optimized orthogonal matching pursuit approach," *IEEE Signal Processing Letters*, vol. 11, no. 9, pp. 705–708, 2004.

[22] Christophe Couvreur and Yoram Bresler, "On the optimality of the backward greedy algorithm for the subset selection problem," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 3, pp. 797–808, 2000.

[23] S. J. Reeves, "An efficient implementation of the backward greedy algorithm for sparse signal reconstruction," *IEEE Signal Processing Letters*, vol. 6, no. 10, pp. 266–268, 1999.

[24] Song Han, Jeff Pool, John Tran, and William Dally, "Learning both weights and connections for efficient neural network," in *Advances in neural information processing systems*, 2015, pp. 1135–1143.

[25] M. A. Carreira-Perpinan and Y. Idelbayev, ""learning-compression" algorithms for neural net pruning," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8532–8541.

[26] Xiaoliang Dai, Hongxu Yin, and Niraj K Jha, "Nest: A neural network synthesis tool based on a grow-and-prune paradigm," *IEEE Transactions on Computers*, vol. 68, no. 10, pp. 1487–1497, 2019.

[27] Charles Soussen, Rémi Gribonval, Jérôme Idier, and Cédric Herzet, "Joint k-step analysis of orthogonal matching pursuit and orthogonal least squares," *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 3158–3174, 2013.

[28] Abhimanyu Das and David Kempe, "Approximate submodularity and its applications: subset selection, sparse approximation and dictionary selection," *The Journal of Machine Learning Research*, vol. 19, no. 1, pp. 74–107, 2018.

[29] Prateek Jain, Ambuj Tewari, and Inderjit S Dhillon, "Orthogonal matching pursuit with replacement," in *Advances in neural information processing systems*, 2011, pp. 1215–1223.