One-shot atomic detection

Yifan Sun*, Michael Friedlander*
*Dept. of Comp. Science, University of British Columbia, Vancouver, BC
email: {ysun13, mpf}@cs.ubc.ca

***Abstract*—Feature selection in data science involves identifying the most prominent and uncorrelated features in the data, which can be useful for compression and interpretability. If these feature can be easily extracted, then a model can be trained over a reduced set of weights, which leads to more efficient training and possibly more robust classifiers. There are many approaches to feature selection; in this work, we propose screening the "atoms" of a gradient of a loss function taken at a random point. We illustrate this approach on sparse and low-rank optimization problems. Despite the simplicity of the approach, we are often able to select the dominant features easily, and greatly improve the runtime and robustness in training overparametrized models.**

I. Introduction

As data becomes more readily available, an important aspect of algorithm design is feature selection, or determining the core subset of features most prominent in accounting for a model's success. Feature selection gives algorithmic advantages that reduce computational and memory requirements, and often improves interpretability and generalizability of results, which are essential prerequisites for deployment.

Our approach to feature selection identifies a solution variable's *atomic structure* (e.g., the nonzero indices or low-rank subspaces). We propose a one-shot selection method by screening the gradient of the smooth objective function at a random point. This is motivated by the the observation that gradient properties can be used to obtain manifold identification rates [NLS17, SJNS19], construct more generalized dynamic safe screening rules [BERG15], and develop better initialization techniques in nonconvex optimization [CLS15].

We evaluate the effectiveness of this random screening procedure on 1) feature selection for interpretability; 2) problem size reduction; and 3) better initialization for optimizing over a nonconvex problem. We do this over three applications:

- logistic regression for binary classification;
- matrix completion for recommender systems [HMLZ15, BKV07]; and
- semidefinite optimization for the MAX-CUT problem [GW95].

For simple two-cluster models for classification, we show that this method mathematically captures the most influential features in the low-noise regime. This is then corroborated on both simulated and real-world data, often with as good or improved test error rates. Moreover, in all cases, the downstream optimization tasks themselves run much faster and more reliably, with less overfitting.

A. Related work

Two notable approaches for safe screening methods for feature selection include [EGVR10, TBF+12], which remove features during optimization and guarantee a sparse and optimal solution. Similar methods are applied to nuclear norm minimization [ZZ15] and dictionary learning [XXR11]. Relatedly, grafting [PLT03] picks nonzeros sequentially based on violations of the optimality condition. The main difference in our work is the one-shot aspect, in which we do not guarantee optimality, but often observe good performance in practice, and can give probabilistic guarantees in simplified models.

II. Gradient screening rules

Gradient screening rules arise as a consequence of the optimality conditions for the problem

$$\underset{x}{\text{minimize}} \quad f(x) + \rho\,\kappa_{\mathcal{A}}(x), \tag{1}$$

where f is a differentiable convex loss function and $\kappa_{\mathcal{A}}$ is a convex function that promotes sparsity with respect to an atomic set $\mathcal{A}$ [CRPW12]. Here, x^* is optimal if and only if $z^* = -\nabla f(x^*) \in \partial\kappa_{\mathcal{A}}(x^*)$ the subdifferential of the nonsmooth regularizer $\kappa_{\mathcal{A}}(x^*)$ [Roc70]. For example,

- $\kappa_{\mathcal{A}}(x) = \|x\|_1 = \sum_i |x_i|$ promotes nonzero element-wise sparsity of x, and at optimality,
$$x_i^* \neq 0 \Rightarrow |z_i^*| = \|z^*\|_\infty;$$
- $\kappa_{\mathcal{A}}(X) = \|X\|_*$ (sum of the singular values) promotes low-rank solutions, and at optimality, X^* and Z^* have singular value decompositions
$$X^* = U\Sigma V^T, \qquad Z^* = U\Lambda V^T,$$
and $\Sigma_{ii} > 0 \Rightarrow \Lambda_{ii} = \max_j \Lambda_{jj}$;
- $\kappa_{\mathcal{A}}(X) = \mathbf{tr}(X) + \delta_{\succeq 0}(X)$[1] promotes low-

[1]The indicator function $\delta_{\succeq 0}(X) = 0$ if $X \succeq 0$ and $= +\infty$ otherwise.

rank, but also constrains X to be positive semidefinite (PSD), and at optimality, X^* and Z^* have eigenvalue decompositions

$$X^* = U\Sigma U^T, \quad Z^* = U\Lambda U^T,$$

and $\Sigma_{ii} > 0 \Rightarrow \Lambda_{ii} = \max_j \Lambda_{jj}$.

At optimality, in order to find active values of x_i^* or subspaces of X^*, it suffices to find the maximal values of z^* or subspaces of Z^*. In general we cannot estimate x^* using a completely random point x. The surprising phenomenon we illustrate here is that the *gradient* at a random point $z = -\nabla f(x)$ *can* be used to approximate key features of $z^* = -\nabla f(x^*)$–namely, it can pick out which features should be nonzero. Since computing a gradient at any point requires one pass through all the data, we call this *one-shot atomic screening*.

Gradient screening rule. Let the set $\mathcal{S}$ characterizes the selected atomic features.

- If $\kappa_{\mathcal{A}} = \|\cdot\|_1$, $\mathcal{S}$ = the indices corresponding to the k largest values of $|z_i|$.
- If $\kappa_{\mathcal{A}} = \|\cdot\|_*$, $\mathcal{S} = \{uv^T : u^T Z v \geq \lambda_k\}$ where λ_k = kth largest singular value of Z.
- If $\kappa_{\mathcal{A}} = \mathbf{tr}(\cdot) + \delta_{\succeq 0}(\cdot)$, $\mathcal{S} = \{uv^T : u^T Z u \geq \lambda_k\}$ where λ_k = kth largest eigenvalue of Z.

We now evaluate the performance of these rules in several common applications.

III. Sparse binary classification

Given a distribution $\mathcal{D}$ over data vectors $a \in \mathbb{R}^n$ and binary labels $b \in \{-1, 1\}$, the goal is to identify a set of weights $x \in \mathbb{R}^n$ such that $\mathbf{sign}(a^T x) = b$ whenever a and b are drawn from $\mathcal{D}$. To do so, we train x from m draws $(a_i, b_i) \in \mathbb{R}^{n+1}$ for $i = 1, \ldots, m$ samples, and solve (1) under logistic loss with a one-norm regularizer:

$$f(x) = \frac{1}{m}\sum_{i=1}^{m} -\log \sigma(b_i a_i^T x), \quad \kappa_{\mathcal{A}}(x) = \|x\|_1$$

where $\sigma(\theta) = \frac{1}{1+e^{-\theta}}$. The ith margin value $b_i a_i^T x > 0$ if and only if $\mathbf{sign}(a_i^T x) = b_i$, and the larger this margin, the better classified is datapoint a_i under this choice of x. In fact, the gradient

$$-\nabla f(x) = \frac{1}{m}\sum_{i=1}^{m} \underbrace{(1 - \sigma(b_i a_i^T x))}_{>0} b_i a_i$$

can be interpreted as a weighted sum of margins. If the ith margin is large, then $\sigma(b_i a_i^T x) \to 1$, and $\nabla f(x)$ is primarily constructed from *weakly classified data samples*; that is, the classifier is working hard on the few points left that are still classified incorrectly, or classified with low confidence. If such datapoints are few and n is large, then at this point the classifier may be overfitting.

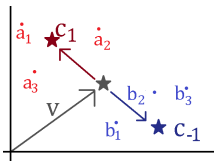

Fig. 1. Illustration of the two-cluster model for binary classification.

However, when x is random, then $1-\sigma(b_i a_i^T x)$ may not be particularly large or small, and a large gradient component corresponds to a data sample with large margin, which may be a better representation of the data distribution. For this reason, screening at a *random gradient* is not so affected by hard-to-classify points, and may resist overfitting more effectively.

A. Two-cluster model

To gain some intuition, we analyze the characteristics of our one-shot gradient over a generalized two-cluster model. We assume balanced labels, e.g., $b_i = \pm 1$ with equal probability, and choose two cluster centers

$$c_1 = v + c, \qquad c_{-1} = v - c,$$

for a bias variable v and cluster center c. The data features are then drawn as

$$a_i = c_{b_i} + u_i, \qquad u_i \sim \mathcal{N}(0, \mathbf{diag}(\bar{u})),$$

where u_i is the noise. (See Fig. 1.) The constant $c_j/(\bar{u}_j + v_j)$ is the signal-to-noise ratio of the jth feature.

The two-cluster model characterizes what we intuitively believe to be the data model in an easy-to-classify problem. Because everything comes from a Gaussian distribution, we can compute the expectation and variance of the margins; e.g., for fixed c and v,

$$\mathbb{E}[ba] = c, \qquad \mathrm{Var}(ba_j) = v_j^2 + \bar{u}_j^2.$$

Specifically, for logistic regression, we can further compute the mean and variance of the gradient terms, and applying the strong law of large numbers, we see that almost surely as $m \to \infty$, [2]

$$\lim_{m\to\infty} \mathbb{E}_x[\nabla f(x)] = \tfrac{1}{2}c,$$

$$\tfrac{1}{4}(v_j^2 + \bar{u}_j^2) \leq \lim_{m\to\infty} \mathrm{Var}([\nabla f(x)_j^2]) \leq \tfrac{1}{2}(v_j^2 + \bar{u}_j^2).$$

In other words, in the two-cluster model, in expectation the gradient exactly represents the feature signal strength, and the variance the noise.

B. Experiments

We evaluate the effectiveness of one-shot selection on 1) our two-cluster model, 2) 0-1 disambiguation of the MNIST dataset [LC10], and 3) the binary classification task over the Dorothea drug discovery dataset [GGBHD05].

[2] This follows from the observation that, for any symmetric distribution, $\mathbb{E}[\sigma(\theta)] = 0.5$ and $0.25 \leq \mathbb{E}[\sigma(\theta)^2] \leq 0.5$.

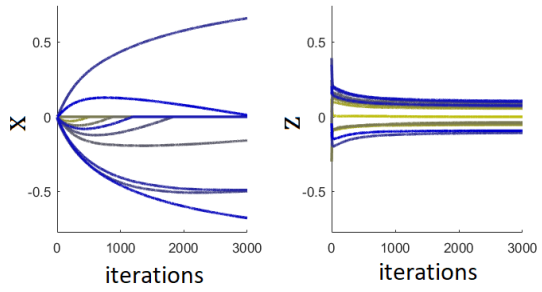

Fig. 2. Trajectory of variable $x_i^{(k)}$ (left) and gradient $z_i^{(k)}$ (right) entries in a sparse logistic regression problem over the weighted two-cluster model.

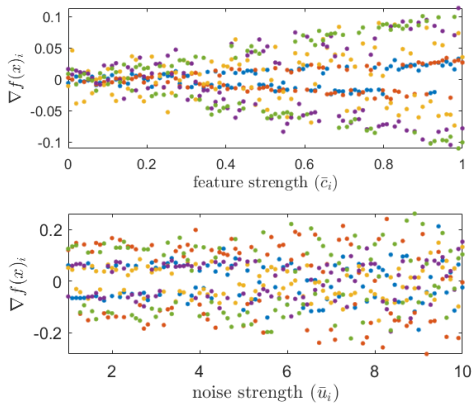

Fig. 3. Gradient strength in two-cluster model, over 5 random draws (different colored dots). Here, $m = n = 100$. In the top plot, we vary $\bar{c}$ and hold $\bar{u} = \mathbf{1}$; in the bottom, $\bar{u}$ is varied and $\bar{c}$ is held at $\mathbf{1}$.

a) Trajectory: Figure 2 shows the trajectory of $x^{(k)}$ and $z^{(k)} = -\nabla f(x^{(k)})$, where $x^{(k)}$ are the iterates of the proximal-gradient method [BJ75, LM79, BT09] over the two-cluster model. Here, $v \sim \mathcal{N}(0, I)$ and $c \sim \mathcal{N}(0, \mathbf{diag}(\bar{c}))$, with strength $\bar{c}_j$ for feature j. In the primal space, the variables $x_i^{(k)}$ interweave and, after a time, snap to 0; however, in the gradient space, although the magnitudes of the gradient values change, their size *relative to each other* seems somewhat constant.

b) Feature selection: Figure 3 illustrates the strength of $\nabla f(x) = -z$ for several random draws of x and training data (a_i, b_i) from our proposed two-cluster model. Clearly, a large $\bar{c}_j$ signal power causes domination of $|z_j|$; however, a large $\bar{u}_j$ noise power just causes more randomness. But, this is consistent with what we may expect; at any value of $\bar{u}_j > 5$, the clusters along the jth dimension are not separable. Figure 4 shows a surface plot of the gradient values for logistic regression over the MNIST handwritten dataset for 0's and 1's. From only a random point x, much of the data has already been captured–the parts that are more 0-like forming ridges and 1-like forming valleys. The

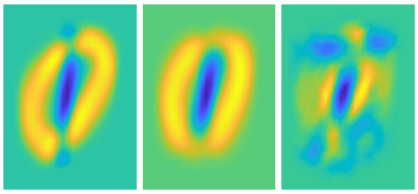

Fig. 4. Surface plots of gradient values at (left) $x =$ a random point, (middle) $x = 0$, and (right) $x = x^*$.

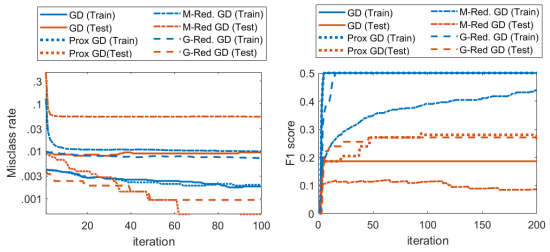

Fig. 5. Left: misclassification rate for MNIST 0-1 detection, and Right: F1 score for Dorothea binary classification task. M-red and G-red refer to solving the problem over the reduced space from screening the margin (M) or noisy gradient (G).

pattern actually disappears at $x = x^*$, where the more important features correspond to discriminating points in hard-to-classify datasets.

c) Optimization over reduced features.: Figure 5 gives a performance comparison for MNIST and Dorothea. We compare simple gradient descent (GD), proximal gradient descent (Prox GD), and gradient descent over the reduced support from the one-shot gradient screening method (M-red and G-red), all using an Armijo-Wolfe line search. With reduced support, the training performance suffers, but the test performance is as good, if not better, than using the full support. And, the runtime of the reduced support method is much faster than the gradient or prox-gradient method: for MNIST, selecting $p = 100 < n = 784$ features reduced the average time-per-iteration from 0.016 seconds to 0.003 seconds, and for Dorothea, selecting $p = 10,000 < n = 100,000$ reduced from 0.19 seconds to 0.020 seconds.

IV. Matrix Completion

We now geneneralize the notion of atomic sparsity to matrix rank, where

$$\kappa_{\mathcal{A}}(X) = \|X\|_* \quad \text{(sum of singular values)}$$

is the well-known nuclear norm. We apply this to recommender systems with binary feedback; e.g., $R_{ij} \in \{-1, 1\}$ for a subset of $i = 1, ..., n_1$ users and $j = 1, ..., n_2$ movies. The problem then reduces to binary classification on each unobserved pair (i, j), which is the solution to the convex problem (1) with loss function

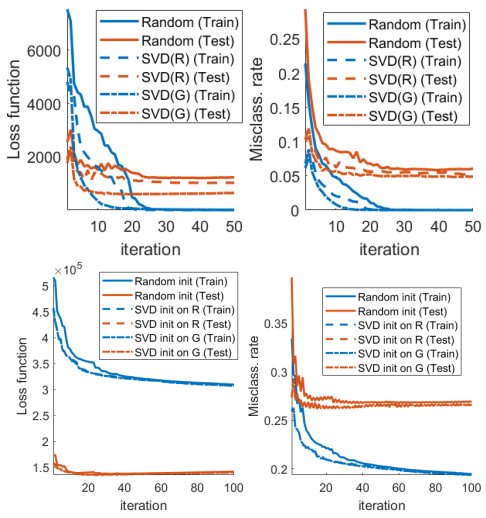

Fig. 6. Loss function and misclassification rate on (top) one-bit observations of a random low-rank matrix and (bottom) the Movielens 1M dataset (quantized so that ratings are $+1$ if > 3, and -1 otherwise). R is the rating matrix, G the noisy gradient.

$$f(X) := \frac{1}{|\mathcal{E}|} \sum_{i,j \in \mathcal{E}} -\log \sigma(R_{ij} X_{ij}), \tag{2}$$

where $\mathcal{E}$ contains the observed index pairs, and $R_{ij}X_{ij}$ is the margin for observation i, j. Our final prediction will be $R_{ij} = \mathbf{sign}(X_{ij})$.

In practice, dealing with the nuclear norm is computationally burdensome, as it requires repeated spectral calculations. A common alternative is to use alternating gradient descent on the nonconvex reformulation

$$\min_{U,V} \quad \frac{1}{|\mathcal{E}|} \sum_{i,j \in \mathcal{E}} -\log \sigma(R_{ij} u_i^T v_j),$$

where $U = [u_1, ..., u_{n_1}]^T \in \mathbb{R}^{n_1 \times r}$ and $V = [v_1, ..., v_{n_2}]^T \in \mathbb{R}^{n_2 \times r}$ implicitly force the solution to be of lower rank r. However, the solution quality to such a problem can depend heavily on initialization. A common technique [BG08] is to set the initial iterates $U^{(0)}, V^{(0)}$ as the dominant singular vectors of the sparse matrix R (where $R_{ij} = 0$ whenever $i, j \notin \mathcal{E}$). We compare this initialization scheme (SVD R) to a random initialization, and initialization via an SVD of the random gradient (SVD G) in Figure 6. In practice, we find the performance of using an SVD of R comparable to that of Z (with variations across trials), though both often outperform random initialization. This suggests that, though $Z \neq R$, this perturbation does not affect its range and corange significantly.

V. Semidefinite relaxation of MAX-CUT

Finally, we take a detour from machine learning to consider the semidefinite relaxations of the MAX-CUT problem, where the constraint

$$x_i \in \{-1, 1\} \quad \Longleftrightarrow \quad X = xx^T, \ \mathbf{diag}(X) = \mathbf{1}.$$

The relaxed MAX-CUT convex relaxation is then

$$\min_X \{-\mathbf{tr}(CX) : \mathbf{diag}(X) = \mathbf{1}, X \succeq 0\}, \tag{3}$$

where C is the adjacency matrix for an undirected graph with n nodes, and the rank-1 requirement is dropped. The solution is then "rounded"; e.g., $\bar{x} = \mathbf{sign}(U^T y)$ and $y \sim \mathcal{N}(0, 1)$, where UU^T is the best rank-r approximation of X, and the cut value is $\bar{x}^T C \bar{x}$.

By shifting the constraint to a smooth penalty, we can relax (3) to a problem of form (1), with

$$f(X) = -\mathbf{tr}((C + \rho I)X) + \frac{\beta}{2} \sum_{i=1}^{n} (X_{ii} - 1)^2 \tag{4}$$

and the eigenspace of $-\nabla f(X)$ resembles that of C (agnostic to ρ). Figure 7 shows histograms of cut values brought about by the rank-r factorizations of X^* the solution to (4), $-\nabla f(X)$ at a random point, and C. We also compare against rounded solutions using the r bottom eigenvectors of L the graph Laplacian [VL07].

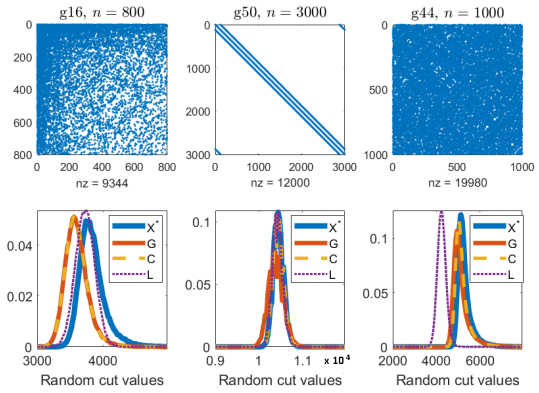

Fig. 7. Top: three adjacency matrices from the DIMACS MAX-CUT challenge set. Bottom: histogram of cut values formed from a rank $r = n/20$ approximate factorization 1) of the entire SDP solution (X^*), 2) a noisy gradient (G), 3) the adjacency matrix (C), and 4) the Laplacian matrix (L). The larger the number, the better the cut value. As a baseline, rounding of random matrices results in similar histogram shapes, but centered at 0.

VI. Conclusion

In sparse optimization, a variable value is nonzero only when its corresponding gradient value is "maximal". Moreover, because the gradient "contains the data structure", the relative size of values remain consistent at random points. We exploit this to promote fast screening techniques for feature selection, dimensionality reduction, and better initialization, observing similar (if not better) results in downstream tasks.

References

[BERG15] Antoine Bonnefoy, Valentin Emiya, Liva Ralaivola, and Remi Gribonval. Dynamic screening: Accelerating first-order algorithms for the lasso and group-lasso. *IEEE Transactions on Signal Processing*, 63(19):5121–5132, 2015.

[BG08] Christos Boutsidis and Efstratios Gallopoulos. Svd based initialization: A head start for nonnegative matrix factorization. *Pattern Recognition*, 41(4):1350–1362, 2008.

[BJ75] Ronald E Bruck Jr. An iterative solution of a variational inequality for certain monotone operators in hilbert space. *Bulletin of the American Mathematical Society*, 81(5):890–892, 1975.

[BKV07] Robert M Bell, Yehuda Koren, and Chris Volinsky. The bellkor solution to the netflix prize. *KorBell Team's Report to Netflix*, 2007.

[BT09] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[CLS15] Emmanuel J Candes, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.

[CRPW12] Venkat Chandrasekaran, Benjamin Recht, Pablo A Parrilo, and Alan S Willsky. The convex geometry of linear inverse problems. *Foundations of Computational mathematics*, 12(6):805–849, 2012.

[EGVR10] L El Ghaoui, V Viallon, and T Rabbani. Safe feature elimination in sparse supervised learning technical report no. Technical report, UCB/EECS-2010-126, EECS Department, University of California, Berkeley, 2010.

[GGBHD05] Isabelle Guyon, Steve Gunn, Asa Ben-Hur, and Gideon Dror. Result analysis of the nips 2003 feature selection challenge. In *Advances in neural information processing systems*, pages 545–552, 2005.

[GW95] Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

[HMLZ15] Trevor Hastie, Rahul Mazumder, Jason D Lee, and Reza Zadeh. Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, 16(1):3367–3402, 2015.

[LC10] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[LM79] Pierre-Louis Lions and Bertrand Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

[NLS17] Julie Nutini, Issam Laradji, and Mark Schmidt. Let's make block coordinate descent go fast: Faster greedy rules, message-passing, active-set complexity, and superlinear convergence. *arXiv preprint arXiv:1712.08859*, 2017.

[PLT03] Simon Perkins, Kevin Lacker, and James Theiler. Grafting: Fast, incremental feature selection by gradient descent in function space. *Journal of machine learning research*, 3(Mar):1333–1356, 2003.

[Roc70] R Tyrrell Rockafellar. *Convex analysis*, volume 28. Princeton university press, 1970.

[SJNS19] Yifan Sun, Halyun Jeong, Julie Nutini, and Mark Schmidt. Are we there yet? manifold identification of gradient-related proximal methods. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1110–1119, 2019.

[TBF+12] Robert Tibshirani, Jacob Bien, Jerome Friedman, Trevor Hastie, Noah Simon, Jonathan Taylor, and Ryan J Tibshirani. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):245–266, 2012.

[VL07] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[XXR11] Zhen J Xiang, Hao Xu, and Peter J Ramadge. Learning sparse representations of high dimensional data on large scale dictionaries. In *Advances in neural information processing systems*, pages 900–908, 2011.

[ZZ15] Qiang Zhou and Qi Zhao. Safe subspace screening for nuclear norm regularized least squares problems. In *International Conference on Machine Learning*, pages 1103–1112, 2015.