Chapter 8

In this chapter we discuss the problem of sparse model selection, i.e., how to perform optimization when the desired/unknown model is constrained by sparsity. While the problem has natural convex translations, we study the iterative hard thresholding (IHT) algorithm and prove results about its performance.

Sparse model selection | Iterative hard thresholding

Over the run of this course, we have mostly been discussing problems of the form:

$$\min_{x\in\mathcal{C}\subseteq\mathbb{R}^p} f(x),$$

where $f(\cdot)$ represents a convex objective, and $x \in \mathcal{C}$ represents some constraint. In this lecture, we will be discussing a case in which $\mathcal{C} \subseteq \mathbb{R}^p$ is *not* a convex constraint, namely, when it is the requirement that x be k-sparse. In order to think about this problem, let us introduce its simplest non-trivial version of this problem: the *sparse linear regression* problem, defined as follows:

$$\begin{aligned} \underset{x\in\mathbb{R}^p}{\text{minimize}} \quad & \tfrac{1}{2}\|y - Ax\|_2^2 \\ \text{subject to} \quad & \|x\|_0 \leq k. \end{aligned}$$

Here, $A \in \mathbb{R}^{n\times p}$ and $y \in \mathbb{R}^n$. Linear regression problems can have a "teacher" generative model assumption where $y = Ax^\star + \text{noise}$, where $x^\star$ is the unknown k-sparse signal we look for. What makes this problem interesting is when we impose the restriction that $n \ll p$; i.e., the problem is ill-posed and classical linear algebra solvers on sets of linear equations do not necessarily recover $x^\star$.

Let us first discuss some procedures that deal with this problem.

- The above problem is non-convex: the inclusion of the ℓ_0-pseudonorm makes the problem non-convex. Classical approaches include convexification: the tightest convex relaxation of the ℓ_0-pseudonorm is that of the ℓ_1-norm (assuming bounded-energy on the initial non-convex set).This leads to the re-definition of the problem as:

  $$\begin{aligned} \underset{x\in\mathbb{R}^p}{\text{minimize}} \quad & \tfrac{1}{2}\|y - Ax\|_2^2 \\ \text{subject to} \quad & \|x\|_1 \leq \lambda. \end{aligned}$$

  There is a long-listed literature on this subject [51–54]; e.g., look into the Rice DSP list of compressed sensing papers (`https://dsp.rice.edu/cs/`). One caveat of this approach is that λ hyperparameter/regularization parameter is not intuitive to be set up correctly (while sparsity k is easier to set up).
- An alternative formulation uses the notion of proximal operators and proximal gradient descent:

  $$\underset{x\in\mathbb{R}^p}{\text{minimize}} \quad \tfrac{1}{2}\|y - Ax\|_2^2 + \rho\|x\|_1.$$

  This formulation "moves" the convex ℓ_1-norm constraint into the objective, and uses the following update rule

  $$x_{t+1} = \text{Prox}_{\rho\|\cdot\|_1}(x_t - \eta\nabla f(x_t)).$$

  In words, the ℓ_1-norm in the objective "biases" the solution towards sparsity (could be seen as an approximation to the exact ℓ_1-norm projection). Similar to the case above, it is not clear how we can select ρ value to achieve good performance.
- Finally, one could keep the non-convexity, and use non-convex projected gradient descent. This leads to

  $$x_{t+1} = \mathcal{H}_k(x_t - \eta\nabla f(x_t)).$$

  This is perhaps somewhat like sorting the input with respect to magnitude and selecting the k largest ones. Similarly to the above cases, it is not trivial to set up k; yet, in many cases, choosing k is more intuitive (remember, this is an integer value), than selecting a continuous-valued regularization parameter like λ and ρ.

In general, the ℓ_0-pseudonorm introduces hardness in the problem definition, since it suggests we solve the problem in a *combinatorial* way: i.e., we are looking for two things: the active support set with k elements, and the values for the corresponding active entries. If we try to select the k size subsets from p, we experience combinatorial explosion. However, the key to focus on here is the word "in general": In this chapter, we will focus on problem cases where randomness is "enough" such that will lead an exception to this rule and admit polynomial complexity.

For the purposes of this chapter, we will mostly focus on the *iterative hard thresholding algorithm* [55–64] or, IHT for short. In IHT, we have:

$$x_{t+1} = H_k(x_t - \eta\nabla f(x_t)),$$

where

$$H_k(z) = \underset{\|x_0\|\leq k}{\text{argmin}}\|x - z\|_2^2.$$

Before continuing discussion on this algorithm, let us first get a sense of what the hyperparameters we are dealing with are:

- Starting point x_0;
- Step size selection η;
- Sparsity level choice k.

For a second, let's imagine that we were dealing with the simple case of $A = I$, i.e., A is the identity matrix in $\mathbb{R}^{p\times p}$. Note that this is an oversimplification of the problem: in this case $n = p$ by definition of the identity matrix. Then, we would end up with a new problem formulation:

$$\begin{aligned} \underset{x\in\mathbb{R}^p}{\text{minimize}} \quad & f(x) = \|y - x\|_2^2 \\ \text{subject to} \quad & \|x\|_0 \leq k. \end{aligned}$$

We have seen this problem before in an earlier homework assignment—this is not something that is difficult to solve. Actually, in this scenario, the whole problem boils down to the simple projection step $H_k(\cdot)$ as defined above. What this problem reformulation tells is the following: given *enough* data y (in this particular case, also non-perturbed data since we do not observe $y = Ax^\star$, but $y = x^\star$), the problem is easy to solve in closed form solution, *even if the problem involves a combinatorially-hard operation; that of a sparse projection.* I.e., we know that $H_k(\cdot)$ introduces some complexity to the overall problem, but there are cases where this does not create issues always.

Isometry and restricted isometries.. Where we should direct our attention is when one deviates from $A = I$ and starts *i*) perturbing the measurements as in $y = Ax^\star$, and *ii*) even more importantly, what happens when $n \ll p$, i.e., we do not have enough measurements to solve the problem with a matrix inversion.

Focus on the following expression: it always holds for $x_1, x_2 \in \mathbb{R}^p$ and for all $\delta \in [0, 1)$:

$$(1-\delta)\|x_1 - x_2\|_2^2 \leq \|I(x_1 - x_2)\|_2^2 \leq (1+\delta)\|x_1 - x_2\|_2^2.$$

It is actually true that the inequalities above hold with equality for $\delta = 0$. What is the purpose of these inequalities? They show *how much the geometry of the vector $x_1 - x_2$ changes when someone applies the operator I on the vector $x_1 - x_2$*. To see this clearly, the left and right hand sides of the above expressions indicate by how much "energy" we deviate from the true image $x_1 - x_2$ (when $\delta > 0$), when we apply $x_1 - x_2$ on I. For this toy example, of course as we mentioned above, we do not lose anything: the above expressions hold with equality for $\delta = 0$.

The above lead to the notion of *isometry*: Intuitively, this means that the matrix I does not perturb the distance between x_1 and x_2 too much, in the sense that the resulting image $I(x_1 - x_2)$ is identical to that of $x_1 - x_2$. The question of course becomes interesting when one deviates from I; e.g., under which conditions the above expressions hold for some A matrix and some δ. Also, does this hold for any vectors x_1, x_2 or should they satisfy some constraints?

The above lead us to the definition of the *restricted isometry property* for sparse vectors.

Definition 31. (Restricted Isometry Property (RIP) [65]) *A matrix $A \in \mathbb{R}^{n\times p}$ where $n \leq p$ satisfies the RIP with constant $\delta_k \in (0, 1)$ if and only if:*

$$(1-\delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1+\delta_k)\|x\|_2^2,$$

$\forall x \in \mathbb{R}^p$ such that $\|x\|_0 \leq k$.

In the literature, other properties like nullspace properties and eigenvalues properties are considered, but for this lecture we will be focusing on the restricted isometry property (henceforth RIP), and the analysis we can do based on this. Note that verifying if a matrix satisfies the RIP is NP-hard. Therefore, let us take for granted we have such a a matrix and now attempt to prove convergence given this restriction; later in the chapter, we will provide a proof that this holds with high probability for general classes of random matrices.

The geometric interpretation of RIP matrices lies the following two key observations: *i*) one difficulty for a matrix A to satisfy RIP is the fact that A might be adversarially picked such that there is no small constant δ that satisfies these two inequalities; *ii*) more importantly, even if A is "nice" enough, it might be the case that the rows of A, n is so much smaller than the dimension p. In other words, A "squeezes" the information/"energy" in x when one applies Ax, making it hard to guarantee that the energy $\|Ax\|_2$ will be comparable to that of the original $\|x\|_2$ for a small δ. What RIP guarantees is that there might exist matrices A that preserve the "energy" (i.e., distances) of high dimensional vectors $x \in \mathbb{R}^p$, when "projected" onto lower-dimensional subspaces $\mathbb{R}^n$, such that $n \ll p$, when x satisfy some interesting properties (here, sparsity).

Convergence proof of non-convex IHT algorithm. For the moment, we will assume that $A \in \mathbb{R}^{n\times p}$ satisfies the RIP, for some $n \ll p$. In order to set up the background, we remind that we consider the following problem:

$$\begin{aligned} \underset{x\in\mathbb{R}^p}{\text{minimize}} \quad & f(x) := \tfrac{1}{2}\|y - Ax\|_2^2 \\ \text{subject to} \quad & \|x\|_0 \leq k. \end{aligned}$$

The IHT algorithm solves this problem with the following gradient-based recursion:

$$x_{t+1} = H_k(x_t - \eta\nabla f(x_t)).$$

This is nothing else but *projected gradient descent*, but the projection step is *non-convex*. Thus, any arguments originating from convex analysis breaks (see Chapter 3). Consider the following example: we will try to prove whether the fact

$$\|H_k(x_1) - H_k(x_2)\|_2 \leq \|x_1 - x_2\|_2$$

holds, which is one of the basic properties of projections onto convex sets. Here, we prove that this is not true anymore. Consider the following two vectors:

$$x_1 = \begin{bmatrix} 1 \\ 10 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 10 \\ 1 \end{bmatrix}$$

Consider the case of $k = 1$. We could use the analysis of convex projected gradient descent if we could have:

$$\begin{aligned} &\|H_1(x_1) - H_1(x_2)\|_2 \leq \|x_1 - x_2\|_2 \Rightarrow \\ &\left\|H_1\left(\begin{bmatrix} 1 \\ 10 \end{bmatrix}\right) - H_1\left(\begin{bmatrix} 10 \\ 1 \end{bmatrix}\right)\right\|_2 \leq \left\|\begin{bmatrix} 1 \\ 10 \end{bmatrix} - \begin{bmatrix} 10 \\ 1 \end{bmatrix}\right\|_2 \Rightarrow \\ &\left\|\begin{bmatrix} 0 \\ 10 \end{bmatrix} - \begin{bmatrix} 10 \\ 0 \end{bmatrix}\right\|_2 \leq \left\|\begin{bmatrix} 1 \\ 10 \end{bmatrix} - \begin{bmatrix} 10 \\ 1 \end{bmatrix}\right\|_2 \Rightarrow \\ &10\sqrt{2} \leq 9\sqrt{2}, \end{aligned}$$

which is not true; thus, we cannot use this property.

We will start by recalling some relevant details to the proof. For the linear regression problem, the gradient of the function satisfies:

$$\nabla f(x_t) = -A^\top(y - Ax_t).$$

Therefore, the IHT recursion for this particular problem can be simplified into:

$$x_{t+1} = H_k(x_t + \eta A^\top(y - Ax_t))$$

We will make the assumption that we know $k = \|x^\star\|_0$. Also, for the moment assume $\eta = 1$; this assumption will be broken in other variants of IHT.[12]

But, even if the projection is non-convex, what can we say about our projection? Denote $\tilde{x}_t = x_t + A^\top(y - Ax_t)$. Also, we know that $x_{t+1} = H_k(\tilde{x}_t)$, i.e., x_{t+1} is the best k-sparse projection of $\tilde{x}_t$, based on the ℓ_2-norm distance. With this notation, this implies:

$$\begin{aligned} &\|x_{t+1} - \tilde{x}_t\|_2^2 \leq \|x^\star - \tilde{x}_t\|_2^2 \Rightarrow \\ &\|(x_{t+1} - x^\star) + (x^\star - \tilde{x}_t)_2^2\| \leq \|x^\star - \tilde{x}_t\|_2^2 \Rightarrow \\ &\|x_{t+1} - x^\star\|_2^2 + \|x^\star - \tilde{x}_t\|_2^2 + 2\langle x_{t+1} - x^\star, x^\star - \tilde{x}_t\rangle \leq \|x^\star - \tilde{x}_t\|_2^2 \Rightarrow \\ &\|x_{t+1} - x^\star\|_2^2 \leq 2\langle x_{t+1} - x^\star, x^\star - \tilde{x}_t\rangle \end{aligned}$$

Now, we have an expression that includes $\|x_{t+1} - x^\star\|_2^2$ on the left hand side, and an inner product that involves (as we will see) x_t and $x^\star$ on the right hand side. To proceed, we will

[12] As we will see, this step size is valid based on the strict assumption that A satisfies the RIP *with symmetry*. I.e., the RIP inequalities $(1-\delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1+\delta_k)\|x\|_2^2$ are centered in the interval $[(1-\delta_k)\|x\|_2^2, (1+\delta_k)\|x\|_2^2]$. However, this symmetry breaks in reality, so step size selection should be completed more carefully.

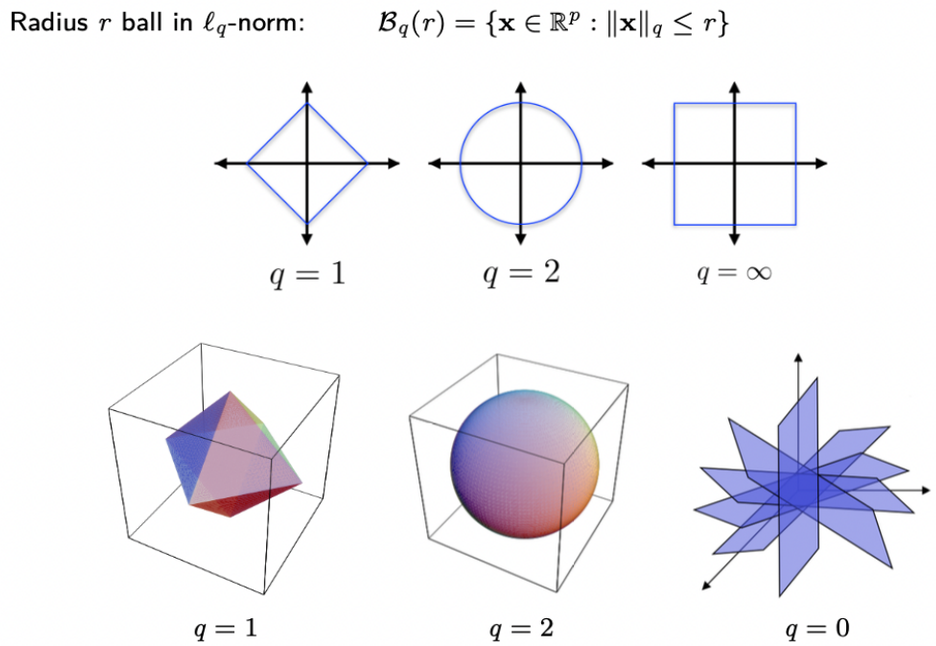

Fig. 47. 2D and 3D representations of some unit norms, both convex and non-convex. The ℓ_0-pseudonorm represents the hyperplanes that span the coordinate system, based on the level of sparsity k.

define $\mathcal{U} := \text{supp}(x_t) \cup \text{supp}(x_{t+1}) \cup \text{supp}(x^\star)$, where $\text{supp}(\cdot)$ is the support function that, given an argument vector, returns the index set of non-zero elements. In words, the set $\mathcal{U}$ contains the union of the support set of the vectors x_t, x_{t+1}, as well as the optimal set $x^\star$ (we will not use any information of the index set of $x^\star$ in the proof, just the fact that it is a k-sparse set).

Since by definition $y = Ax^\star$ and the fact that $\tilde{x}_t = x_t + A^\top(y - Ax_t)$, we have:

$$\begin{aligned}\tilde{x}_t = x_t + A^\top(y - Ax_t) &= x_t + A^\top(Ax^\star - Ax_t) \\ &= x_t + A^\top A(x^\star - x_t).\end{aligned}$$

We will use this definition of $\tilde{x}_t$ in the inequality above. In particular, we have:

$$\begin{aligned}&\|x_{t+1} - x^\star\|_2^2 \\ &\quad \leq 2\langle x_{t+1} - x^\star, x_t + A^\top A(x^\star - x_t) - x^\star\rangle,\end{aligned}$$

where the RHS equals to:

$$2\langle x_{t+1} - x^\star, (I - A_{\mathcal{U}}^\top A_{\mathcal{U}}) \cdot (x_t - x^\star)\rangle.$$

Here, $A_{\mathcal{U}}$ indicates the matrix A with only columns restricted and indexed by the set $\mathcal{U}$. This selection is based on a key product of the inner product operator to note:

$$\langle x, A^\top y\rangle = x^\top A^\top y = (Ax)^\top y = \langle Ax, y\rangle.$$

I.e., in the quadratic form, the matrix could be "moved" to be applied both on the left and right hand side of the operator $\langle \cdot, \cdot \rangle$. This means that we can safely restrict the active columns of A on the union of the support set of the vectors $x_{t+1} - x^\star$ and $x_t - x^\star$, which are subsets of the superset $\mathcal{U}$.

For the main term in our recursion, we have:

$$\begin{aligned}&\langle x_{t+1} - x^\star, (I - A_{\mathcal{U}}^\top A_{\mathcal{U}})(x_t - x^\star)\rangle \\ &\qquad \leq \|x_{t+1} - x^\star\|_2 \cdot \|(I - A_{\mathcal{U}}^\top A_{\mathcal{U}})(x_t - x^\star)\|_2 \\ &\qquad \leq \|x_{t+1} - x^\star\|_2 \cdot \|I - A_{\mathcal{U}}^\top A_{\mathcal{U}}\|_2 \cdot \|x_t - x^\star\|_2\end{aligned}$$

where, again, we use Cauchy-Schwartz inequality, and by using the RIP bounds, we can show that:

$$\|I - A_{\mathcal{U}}^\top A_{\mathcal{U}}\|_2 \leq \max\{(1 + \delta_k) - 1, 1 - (1 - \delta_k)\} = \delta_k.$$

Using the above in our main expression, we obtain:

$$\begin{aligned}&\|x_{t+1} - x^\star\|_2^2 \leq 2\delta_k \|x_{t+1} - x^\star\|_2 \cdot \|x_t - x^\star\|_2 \implies \\ &\|x_{t+1} - x^\star\|_2 \leq 2\delta_k \|x_t - x^\star\|\end{aligned}$$

Let us define $\rho := 2\delta_k$. One logical expectation for convergence is to assume/require $\rho < 1$, which further assumes $\delta_k \leq \frac{1}{2}$. (*Later on, we will see how the δ_k requirements affect the number of measurements n the matrix A should have in order to guarantee this convergence, and thus the $x^\star$ recovery*).

In what follows, we will unroll our main recursion over t iterations to obtain the following:

$$\begin{aligned}\|x_{t+1} - x^\star\|_2 &\leq \rho \cdot \|x_t - x^\star\|_2 \\ &\leq \rho^t \cdot \|x_0 - x^\star\|_2,\end{aligned}$$

based on $\rho < 1$. To conclude, this implies that we can obtain $\|x_{t+1} - x^\star\|_2 \leq \varepsilon$ by running IHT for $O(\log \frac{\|x_0 - x^\star\|_2}{\varepsilon})$ iterations.

Step size based on convex optimization analysis. In the analysis above, we have used the fact that $\eta = 1$. Yet, this selection does not work well in practice (as we can see in the Demo file of this chapter). The reason for this behavior is that, often in practice, the *symmetry* in the RIP condition is not always satisfied. I.e., bounds:

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2,$$

[13] The selection for this notation is on purpose.

are not "centered". More specifically, there might be some lower and upper bound constants, μ_k and L_k[13], such that we still have:

$$\mu_k\|x\|_2^2 \leq \|Ax\|_2^2 \leq L_k\|x\|_2^2,$$

Can we pick a new step size based on the RIP property?

It not hard to show that actually these lower and upper bound constants are the *minimum and maximum* eigenvalues of the Hessian matrix, when one is restricted to sparse signals. I.e., one can use (μ_k, L_k), if known, to apply step size selection techniques, like the ones we used in convex optimization. E.g.,

- In Convex Optimization, $\eta = \frac{1}{L}$ works well (where L is the Lipschitz constant of the objective function). L is also the upper bound on the eigenvalues of the Hessian of the function.
- In our case, we have L_k (but assumed known for now). In the symmetric version of RIP, $L_k = (1+\delta_k)$.

Let us drive a deeper connection between the above notions. By definition of $f(\cdot)$, for x_1, x_2 that are k-sparse, and using the definition of the L-Lipschitzness, we have:

$$\begin{aligned}\|\nabla f(x_1) - \nabla f(x_2)\|_2 &= \|-A^\top(y - Ax_1) + A^\top(y - Ax_2)\|_2 \\ &= \|A^\top A(x_1 - x_2)\|_2 \\ &\leq \max_{\mathcal{S}:|\mathcal{S}|\leq 2k} \|(A^\top A)_{\mathcal{S}}\|_2 \cdot \|x_1 - x_2\|_2 \\ &\leq (1+\delta_{2k})\|x_1 - x_2\|_2\end{aligned}$$

by definition of RIP on $2k$-sparse vectors. This drives the connection that, similarly to convex optimization that one uses $\eta = \frac{1}{L}$, one could potentially use $\eta = \frac{1}{1+\delta_{2k}}$ as a step size. Yet, the difficulty of this choice is that δ-values are NP-hard to know a priori. So a better strategy should be devised.

Adaptive Step Sizes. To close the IHT section, we will consider adaptive step sizes. We want to consider whether there are efficient adaptive step size selection formulas η_t in $x_{t+1} = H_k(x_t - \eta_t \cdot \nabla f(x_t))$.

To do so, let us start with some observations:

- x_t is k-sparse;
- x_{t+1} is k-sparse;
- x_{t+1} could potentially have intersection with the support set of x_t, as well as the set $H_k(-\nabla f(x_t))$ (outside of $\text{supp}(x_t)$).

Schematically, the above observations lead to the following picture for the IHT recursion:

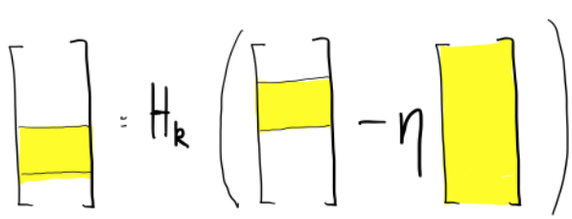

Fig. 48. The above scheme illustrates what happens per iteration in the IHT algorithm. Yellow parts indicate the non-zeros; they are grouped together here, without losing generality.

We will present the idea of *line search*. This is the case where choosing step size is the result of an optimization problem as in:

$$\eta := \underset{\eta}{\text{argmin}} \|y - A(x_t - \eta\nabla f(x_t)\|_2$$

As we will show in the Demo, such approaches generally perform better in practice than any constant step size selection that theory might suggest. Key attribute for line search approaches is for η to be easily computable.

To complete the above, let us define first:

$$\begin{aligned}\mathcal{S}_t &= \text{supp}(x_t) \\ \mathcal{Q}_t &= \mathcal{S}_t \cup \text{supp}(H_k(\nabla_{\mathcal{S}_t^c} f(x_t))) \\ \mathcal{S}_{t+1} &= \text{supp}(x_{t+1}) \subseteq \mathcal{Q}_t,\end{aligned}$$

where $\mathcal{S}^c$ represent the complement of a set. Then, based on the scheme above, observe that:

$$H_k(x_t - \eta\nabla f(x_t)) = H_k(x_t - \eta \cdot \nabla_{\mathcal{Q}_t} f(x_t));$$

i.e., what matters in the gradient $\nabla f(x_t)$ is indexed by the set $\mathcal{Q}_t$. This observation changes the line search problem above as:

$$\eta = \underset{\eta}{\text{argmin}} \|y - A(x_t - \eta \cdot \nabla_{\mathcal{Q}_t} f(x_t))\|_2^2.$$

But, what is the solution to this 1D problem with respect to η? Define the auxiliary objective $g(\eta) := \|y - A(x_t - \eta \cdot \nabla_{\mathcal{Q}_t} f(x_t))\|_2^2$. Taking the derivative and setting it equal to zero:

$$\begin{aligned}0 &= \nabla g(\eta) \\ &= 2\langle A\nabla_{\mathcal{Q}_t} f(x_t), y - Ax_t\rangle + 2\eta\|A\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2 \\ \implies \eta &= \frac{-\langle A\nabla_{\mathcal{Q}_t} f(x_t), y - Ax_t\rangle}{\|A\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2} = \frac{\|\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2}{\|A\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2}\end{aligned}$$

Can we relate η to the RIP? We know that in the original definition, the following holds:

$$1 - \delta \leq \frac{\|Ax\|_2^2}{\|x\|_2^2} \leq 1 + \delta,$$

for all sparse vectors x. In our case above, $\nabla_{\mathcal{Q}_t} f(x_t)$ is still a sparse vector. How much sparse? $2k$-sparse! Thus, the term $\frac{\|\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2}{\|A\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2}$ has A applying on the sparse gradient vector, which further leads to (based on the RIP bounds:

$$1 + \delta \leq \eta \leq \frac{1}{1-\delta}.$$

But is this computed efficiently? Well, it turns out that $\eta_t = \frac{\|\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2}{\|A\nabla_{\mathcal{Q}_t} f(x_t)\|_2^2}$. Here, the gradient vector is already computed per iteration; what we only need to compute is the set $\mathcal{Q}_t$ which depends on sorting the elements of the dense gradient vector and selecting the k-sparse best subset out of the $\mathcal{S}_t$ set. Finally, applying the operation $A\nabla_{\mathcal{Q}_t} f(x_t)$ does not add much in the overall complexity of the algorithm. Thus, computing η_t is efficient! And it comes with nice theoretical properties that we can use!

Proof of Adaptive Step Sizes in IHT. Following the same procedure as in $\eta = 1$, we have[14]

$$\|x_{t+1} - x^\star\|_2 \leq 2\|I - \eta A_{\mathcal{U}}^\top A_{\mathcal{U}}\|_2 \cdot \|x_t - x^\star\|_2$$

[14] We drop the dependence on k in δ_k for ease of exposition.

By RIP along with η inclusion in the equations, we get:

$$\begin{aligned}\|I-\eta A_{\mathcal{U}}^{\top}A_{\mathcal{U}}\|_2 &\le \max\{\eta(1+\delta)-1, 1-\eta(1-\delta)\}\\ &\le \max\left\{\tfrac{1+\delta}{1-\delta}-1, 1-\tfrac{1-\delta}{1+\delta}\right\}\\ &\le \tfrac{2\delta}{1-\delta},\end{aligned}$$

where the last inequality we use the property $1+\delta \le \eta \le \frac{1}{1-\delta}$ of the step size. Then, going back to the original expression:

$$\begin{aligned}\|x_{t+1}-x^\star\|_2 &\le 2\tfrac{2\delta}{1-\delta}\cdot\|x_t-x^\star\|_2\\ &= \tfrac{4\delta}{1-\delta}\|x_t-x^\star\|_2.\end{aligned}$$

Assuming:

$$\delta < \tfrac{1}{5},$$

we get

$$\tfrac{4\delta}{1-\delta} =: \rho < 1.$$

As shown in the proof of convergence of regular IHT (in which $\rho<1$), we get convergence.

$$\text{———————} \infty \text{———————}$$

Graphical Model Selection. Let $x \sim \mathcal{N}(\mu,\Sigma)$. Then its probability density satisfies:

$$f(x) = \frac{1}{(2\pi)^{p/2}\det(\Sigma)^{1/2}}\cdot\exp\left\{-\frac{1}{2}(x-\mu)^\top\Sigma^{-1}(x-\mu)\right\}$$

Define $\Theta=\Sigma^{-1}$ as the inverse covariance matrix or precision matrix. Then:

$$f(x) = \frac{\det(\Theta)^{1/2}}{(2\pi)^{p/2}}\cdot\exp\left\{-\frac{1}{2}(x-\mu)^\top\cdot\Theta\cdot(x-\mu)\right\}$$

We now introduce the problem definition: assume we do not know (μ,Σ), but we have samples $\{x_i\}_{i=1}^n$, $x_i\sim\mathcal{N}(\mu,\Sigma)$. Let's see what we can do with these samples.

Assume independence between the x_i's. The log-likelihood function is:

$$\begin{aligned}l(\mu,\theta) &= \sum_{i=1}^n \log f(x_i)\\ &\propto \sum_{i=1}^n \log\det(\Theta)^{1/2} - \sum_{i=1}^n\frac{1}{2}(x_i-\mu)^\top\Theta(x_i-\mu)\\ &= \frac{n}{2}\log\det(\Theta) - \frac{1}{2}\sum_{i=1}^n(x_i-\mu)^\top\cdot\Theta\cdot(x_i-\mu)\end{aligned}$$

Observe that:

$$\begin{aligned}&-\operatorname{tr}(\Theta\cdot\hat{\Sigma}) - (\mu-\hat{mu})^\top\Theta(\mu-\hat{mu})\\ =&-\operatorname{tr}(\Theta\cdot\frac{1}{n}\sum_{i=1}^n(x_i-\frac{1}{n}\sum_{i=1}^n x_i)(x_i-\frac{1}{n}\sum_{i=1}^n x_i)^\top)\\ &-(\mu-\frac{1}{n}\sum_{i=1}^n x_i)^\top\Theta(\mu-\frac{1}{n}\sum_{i=1}^n x_i)\end{aligned}$$

(where we used $\hat{\mu}=\frac{1}{n}\sum_{i=1}^n x_i$ and $\hat{\Sigma}=\frac{1}{n}\sum_{i=1}^n(x_i-\hat{\mu})(x_i-\hat{\mu})^\top$)

$$\begin{aligned}&= -\frac{1}{n}\sum_{i=1}^n(x_i-\frac{1}{n}\sum_{i=1}^n x_i)^\top\Theta(x_i-\frac{1}{n}\sum_{i=1}^n x_i)\\ &-(\mu-\frac{1}{n}\sum_{i=1}^n x_i)^\top\Theta(\mu-\frac{1}{n}\sum_{i=1}^n x_i)\end{aligned}$$

$$= -\frac{1}{n}\sum_{i=1}^n(x_i-\mu)^\top\Theta(x_i-\mu)$$

Thus our $l(\cdot,\cdot)$ transforms into

$$l(\mu,\Theta) = \frac{n}{2}\left(\log\det(\Theta)-\operatorname{tr}(\Theta\cdot\hat{\Sigma})-(\mu-\hat{\mu})^\top\Theta(\mu-\hat{\mu})\right)$$

Maximum likelihood estimation of (μ,Σ) leads to:

$$\min_{\mu,\theta\succ 0} -\log\det(\Theta)+\operatorname{tr}(\Theta\cdot\hat{\Sigma})+(\mu-\hat{\mu})^\top\Theta(\mu-\hat{\mu})$$

.

Only the last term in the above expression contains μ; and since $\Theta\succ 0$, $\mu^*=\hat{\mu}$. So letting $\mu^*=\hat{\mu}$, we find

$$\min_{\substack{\Theta\succ 0\\ \Theta\in\mathbb{R}^{p\times p}}} -\log\det(\Theta)+\operatorname{tr}(\Theta\cdot\hat{\Sigma}) = -\log\det(\Theta)+\langle\Theta,\hat{\Sigma}\rangle$$

As a side note, the determinant of a squared matrix is (relatively) not an easy object/operation to describe. The geometric way of thinking of it is if we had a unit cube in p dimensions, then $\det(\Theta)$ measures the volume of the cube, after applying the rows/columns of Θ on that cube. Another way to see it is

$\det(\Theta)=\prod_{i=1}^p\lambda_i(\Theta)$, where $\lambda_i(\Theta)$ is the i-th eigenvalue of Θ.

Why do we care about all this? There is a very nice theory connecting undirected graphs under Gaussian assumptions and covariance selection. This theory assumes that variables $x(i),x(j)$ from $x\sim\mathcal{N}(\mu,\Sigma)$ are conditionally independent if and only if $\Theta^*_{ij}=0$. You can see the example drawn out in the slides.

Concretely, we can ask the question: given samples $\{x_i\}_{i=1}^n$, can we infer the underlying undirected graph structure?

Answer #1: We can take many samples, and use them to compute $\hat{\mu}$, $\hat{\Sigma}$. Then we can derive $\hat{\Sigma}^{-1}$. But if p is on the order of 10^5 to 10^6, this is often impossible.

Answer #2: We find the most important part of the graph. Assuming sparsity in Σ^{-1}, we find $\Theta=\Sigma^{-1}$ satisfying:

$$\min_{\Theta\succ 0} -\log\det(\Theta)-\operatorname{tr}(\Theta\cdot\hat{\Sigma})$$

$$\text{s.t. } \|\Theta\|_0 \le k \text{ (Assuming we obey symmetry)}$$

Note that $-\log\det(\Theta)+\operatorname{tr}(\Theta\cdot\hat{\Sigma})$ is locally Lipschitz gradient.

We omit the proof of RIP for Sub-Gaussian matrices.

RIP proof for sub-Gaussian matrices. Matrices that satisfy:

$$\mathbb{P}_{A\sim D^{n\times p}}[\|A_x\|_2^2-\|x\|_2^2] > \epsilon\cdot\|x\|_2^2 (\le 2e^{-\Omega(n)}),$$

will also satisfy the RIP property with probability $1-2e^{-\Omega(n)}$, whenever $n\ge\Omega(\frac{k}{\delta^2}\log\frac{p}{k})$. So this hints at a way to get RIP matrices (which, as we mentioned before, were computationally expensive to verify). Both Gaussian and Bernoulli matrices $A\in\mathbb{R}^{n\times p}$ will satisfy the above property, and therefore make good candidates.

Below, we use the following definitions: a random variable x is called Sub-Gaussian if there exist $\beta,k>0$ such that

$$\mathbb{P}(|x|\ge t)\le\beta e^{-kt^2}, \forall t>0$$

In general, x is called Sub-Exponential if there exist $\beta, k > 0$ such that

$$\mathbb{P}(|x| \geq t) \leq \beta \cdot e^{-kt}, \forall t > 0$$

Finally, a vector $y \in \mathbb{R}^p$ is called isotropic if $\mathbb{E}[|\langle y, x\rangle|^2] = \|x\|_2^2, \forall x \in \mathbb{R}^p$.

Step 1: *Let $A \in \mathbb{R}^{n\times p}$ with independent, isotropic, and Sub-Gaussian rows. Then, $\forall x \in \mathbb{R}^p$ and $\forall t \in (0,1)$:*

$$\mathbb{P}\left(\left|\frac{1}{n}\|AX\|_2^2 - \|x\|_2^2\right| \geq t \cdot \|x\|_2^2\right) \leq 2e^{-ct^2 n}, c \text{ constant}$$

Proof: W.L.O.G., $\|x\|_2 = 1$. Let $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{R}^p$ be the rows of A. Define $z_i = |\langle \alpha_i, x\rangle|^2 - \|x\|_2^2$. Since α_i is isotropic, $\mathbb{E}[z_i] = 0$. Further, z_i is Sub-Exponential, since $\langle \alpha_i, x\rangle$ is Sub-Gaussian; this means

$$\mathbb{P}(|z_i| \geq r) \leq \beta e^{-kr}, \forall r > 0$$

Observe:

$$\frac{1}{n}\|Ax\|_2^2 - \|x\|_2^2 = \frac{1}{n}\sum_{i=1}^n (|\langle \alpha_i, x\rangle|^2 - \|x\|_2^2) = \frac{1}{2}\sum_{i=1}^n z_i$$

Since the α_i's are independent, the z_i's are independent. We will now use the following Berstein inequality: Let $x_1, \ldots, X_M$ be independent, zero-mean, Sub-Exponential random variables, with constants β, k. Then:

$$\mathbb{P}\left(\left|\sum_{i=1}^M x_i\right| \geq t\right) \leq 2e^{-\frac{(kt)^2/2}{2\beta M + kt}}$$

In our case, this translates into

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^n z_i\right| \geq t\right) = \mathbb{P}\left(\left|\sum_{i=1}^n z_i\right| \geq tn\right) \leq 2e^{-\frac{k^2 n^2 t^2/2}{2\beta n + knt}}$$

$$\leq 2e^{-\frac{k^2}{4\beta + 2k} \cdot nt^2} \quad \text{for } t \in (0,1)$$

□

Step 2: *Assume Step 1 holds. Fix a set $S \subset [p]$ with $|S| = k$ and δ, $\xi \in (0,1)$. If*

$$n \geq \frac{c}{\delta^2}\left(7k + 2\ln\left(\frac{2}{\xi}\right)\right), c \text{ constant}$$

Then W.P. at $1 - \xi$:

$$\|A_s^\top A_s - I\|_2 < \delta$$

Proof: We will use the construction of ϵ-nets over unit balls. Let $B = \{x \in \mathbb{R}^p, \|x\|_2 \leq 1\}$. An ϵ-net over B is a set such that, for every point in B, there is a point in the ϵ-net that is ϵ-close by some distance function (e.g., $\|x - y\|_2 \leq \epsilon$). The number of points in such an ϵ-net can be bounded by:

$$\mathcal{N}(B, \|\cdot\|_2, \epsilon) \leq \left(1 + \frac{2}{\epsilon}\right)^p$$

In our case, we generate an ϵ-net on $B = \{x \in \mathbb{R}^p, \text{supp}(x) \subset S, \|x\|_2 \leq 1\}$. In this case:

$$\mathcal{N}(B, \|\cdot\|_2, \epsilon) \leq \left(1 + \frac{2}{\epsilon}\right)^k$$

Then, from Step 1:

$$\mathbb{P}\left(|\|Au\|_2^2 - \|u\|_2^2| \geq t \cdot \|u\|_2^2, \text{for some } u \text{ in } \epsilon\text{-net}\right)$$

$$\leq \sum_{u \text{ in } \epsilon\text{-net}} \mathbb{P}\left(|\|Au\|_2^2 - \|u\|_2^2| \geq t \cdot \|u\|_2^2\right)$$

$$\leq 2 \cdot \left(1 + \frac{2}{\epsilon}\right)^k e^{-ct^2 n}$$

Define $D = A_s^\top A_s - I$. Then:

$$\begin{aligned}|\|Au\|_2^2 - \|u\|_2| &= \left|\langle A_s^\top A_s u, u\rangle - \langle u, u\rangle\right| \\ &= \left|\langle (A_s^\top A_s - I)u, u\rangle\right| \\ &= |\langle Du, u\rangle|\end{aligned}$$

Then, our goal is to prove $|\langle Dx, x\rangle| < t$ (for $x \in B$, and proper t) via $|\langle Du, u\rangle| < t$ where u is in the ϵ-net.

Assume $|\langle Du, u\rangle| < t$. This occurs W.P. $1 - 2\left(1 + \frac{2}{\epsilon}\right)^k e^{-ct^2 n}$. Then, for some $x \in B$, and some u in ϵ-net such that $\|x - u\|_2 \leq \epsilon < \frac{1}{2}$, we get:

$$\begin{aligned}|\langle Du, u\rangle| &= |\langle Du, u\rangle + \langle D(x+u), x-u\rangle| \\ &\leq |\langle Du, u\rangle| + |\langle D(x+u), x-u\rangle| \\ &\leq t + \|D\|_2 \cdot \|x+u\|_2 \cdot \|x-u\|_2 \leq t + 2 \cdot \|D\|_2 \cdot \epsilon\end{aligned}$$

Taking the maximum over $x \in B$:

$$\|D\|_2 < t + 2\|D\|_2 \cdot \epsilon \implies \|D\|_2 \leq \frac{t}{1 - 2\epsilon}$$

Choose $t = (1 - 2\epsilon) \cdot \delta \to \|D\|_2 < \delta$. This means:

$$\mathbb{P}\left(\|A_s^\top A_s - I\|_2 \geq \delta\right) \leq 2\left(1 + \frac{2}{\epsilon}\right)^k e^{-c(1-2\epsilon)^2 \delta^2 n}$$

Choosing $\epsilon = \frac{2}{e^{7/2} - 1}$, we get that $\|A_s^\top A_s - I\|_2 \leq \delta$ with probability $1 - \xi$ provided

$$n \geq \frac{c}{\delta^2}\left(7k + 2\ln\left(\frac{2}{\xi}\right)\right)$$

□

Step 3: *We proved that $\|A_s^\top A_s - I\|_2 < \delta$ for a single s. Taking all $\binom{p}{k}$ subsets $S \subset [p]$ with $|S| = k$, we get:*

$$\begin{aligned}\mathbb{P}\left(\sup_S \|A_s^\top A_s - I\|_2 \geq \delta\right) &\leq \sum_s \mathbb{P}(\|A_s^\top A_s - I\|_2 \geq \delta) \\ &\leq 2\binom{p}{k}\left(1 + \frac{2}{\epsilon}\right)^k \cdot e^{-c(1-2\epsilon)^2\delta^2 n} \\ &\leq 2\left(\frac{ep}{k}\right)^k \left(1 + \frac{2}{\epsilon}\right)^k e^{-c(1-2\epsilon)^2\delta^2 n}\end{aligned}$$

Forcing this probability to be less than ξ, we get

$$n \geq O\left(k\ln\left(\frac{ep}{k}\right) + \frac{14}{3}k + \frac{4}{3}\ln\left(\frac{2}{\xi}\right)\right)$$

□

Appendix

1. J. Nocedal and S. Wright. Numerical optimization. Springer Science & Business Media, 2006.
2. Y. Nesterov. Introductory lectures on convex optimization: A basic course, volume 87. Springer Science & Business Media, 2013.
3. S. Boyd and L. Vandenberghe. Convex optimization. Cambridge university press, 2004.
4. D. Bertsekas. Convex optimization algorithms. Athena Scientific Belmont, 2015.
5. Sébastien Bubeck. Convex optimization: Algorithms and complexity. Foundations and Trends® in Machine Learning, 8(3-4):231–357, 2015.
6. S. Weisberg. Applied linear regression, volume 528. John Wiley & Sons, 2005.
7. T. Hastie, R. Tibshirani, and M. Wainwright. Statistical learning with sparsity: the lasso and generalizations. CRC press, 2015.
8. J. Friedman, T. Hastie, and R. Tibshirani. The elements of statistical learning, volume 1. Springer series in statistics New York, 2001.
9. M. Paris and J. Rehacek. Quantum state estimation, volume 649. Springer Science & Business Media, 2004.
10. M. Daskin. A maximum expected covering location model: formulation, properties and heuristic solution. Transportation science, 17(1):48–70, 1983.
11. I. Goodfellow, Y. Bengio, and A. Courville. Deep learning. MIT press, 2016.
12. L. Trefethen and D. Bau III. Numerical linear algebra, volume 50. Siam, 1997.
13. G. Strang. Introduction to linear algebra, volume 3. Wellesley-Cambridge Press Wellesley, MA, 1993.
14. G. Golub. Cmatrix computations. The Johns Hopkins, 1996.
15. A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.
16. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.
17. S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.
18. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, pages 3111–3119, 2013.
19. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014.
20. Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pages 1243–1252. JMLR. org, 2017.
21. Haşim Sak, Andrew Senior, and Françoise Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In Fifteenth annual conference of the international speech communication association, 2014.
22. Tom Sercu, Christian Puhrsch, Brian Kingsbury, and Yann LeCun. Very deep multilingual convolutional neural networks for LVCSR. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4955–4959. IEEE, 2016.
23. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. page arXiv:1706.03762, 2017.
24. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. page arXiv:1810.04805, 2018.
25. Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and VQA. In AAAI, pages 13041–13049, 2020.
26. Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. arXiv preprint arXiv:2005.14165, 2020.
27. Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using gpu model parallelism. arXiv preprint arXiv:1909.08053, 2019.
28. Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683, 2019.
29. Gary Marcus, Ernest Davis, and Scott Aaronson. A very preliminary analysis of DALL-E 2. arXiv preprint arXiv:2204.13807, 2022.
30. John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with AlphaFold. Nature, 596(7873):583–589, 2021.
31. Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.
32. Or Sharir, Barak Peleg, and Yoav Shoham. The cost of training nlp models: A concise overview. arXiv preprint arXiv:2004.08900, 2020.
33. H. Karimi, J. Nutini, and M. Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 795–811. Springer, 2016.
34. Philip Wolfe. Convergence conditions for ascent methods. SIAM review, 11(2):226–235, 1969.
35. Larry Armijo. Minimization of functions having Lipschitz continuous first partial derivatives. Pacific Journal of mathematics, 16(1):1–3, 1966.
36. Stephen Wright and Jorge Nocedal. Numerical optimization. Springer Science, 35(67-68):7, 1999.
37. B. Polyak. Introduction to optimization. Inc., Publications Division, New York, 1, 1987.
38. Stephen Boyd, Lin Xiao, and Almir Mutapcic. Subgradient methods. lecture notes of EE392o, Stanford University, Autumn Quarter, 2004:2004–2005, 2003.
39. Marguerite Frank, Philip Wolfe, et al. An algorithm for quadratic programming. Naval research logistics quarterly, 3(1-2):95–110, 1956.
40. M. Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In Proceedings of the 30th international conference on machine learning, number CONF, pages 427–435, 2013.
41. J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In Proceedings of the 25th international conference on Machine learning, pages 272–279, 2008.
42. Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. Computer, 42(8):30–37, 2009.
43. A. Mnih and R. Salakhutdinov. Probabilistic matrix factorization. In Advances in neural information processing systems, pages 1257–1264, 2008.
44. T. Booth and J. Gubernatis. Improved criticality convergence via a modified Monte Carlo power iteration method. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.
45. S. Zavriev and F. Kostyuk. Heavy-ball method in nonconvex optimization problems. Computational Mathematics and Modeling, 4(4):336–341, 1993.
46. E. Ghadimi, H. Feyzmahdavian, and M. Johansson. Global convergence of the heavy-ball method for convex optimization. In 2015 European control conference (ECC), pages 310–315. IEEE, 2015.
47. Y. Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. In Dokl. akad. nauk Sssr, volume 269, pages 543–547, 1983.
48. B. O'Donoghue and E. Candes. Adaptive restart for accelerated gradient schemes. Foundations of computational mathematics, 15(3):715–732, 2015.
49. O. Devolder, F. Glineur, and Y. Nesterov. First-order methods of smooth convex optimization with inexact oracle. Mathematical Programming, 146(1-2):37–75, 2014.
50. L. Bottou, F. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. Siam Review, 60(2):223–311, 2018.
51. S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. SIAM review, 43(1):129–159, 2001.
52. R. Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1):267–288, 1996.
53. P. Hoff. Lasso, fractional norm and structured sparse estimation using a Hadamard product parametrization. Computational Statistics & Data Analysis, 115:186–198, 2017.
54. S. Becker, J. Bobin, and E. Candès. NESTA: A fast and accurate first-order method for sparse recovery. SIAM Journal on Imaging Sciences, 4(1):1–39, 2011.
55. T. Blumensath and M. Davies. Iterative hard thresholding for compressed sensing. Applied and computational harmonic analysis, 27(3):265–274, 2009.
56. D. Needell and J. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Applied and computational harmonic analysis, 26(3):301–321, 2009.
57. S. Foucart. Hard thresholding pursuit: an algorithm for compressive sensing. SIAM Journal on Numerical Analysis, 49(6):2543–2563, 2011.
58. J. Tanner and K. Wei. Normalized iterative hard thresholding for matrix completion. SIAM Journal on Scientific Computing, 35(5):S104–S125, 2013.
59. K. Wei. Fast iterative hard thresholding for compressed sensing. IEEE Signal processing letters, 22(5):593–597, 2014.
60. Rajiv Khanna and Anastasios Kyrillidis. Iht dies hard: Provable accelerated iterative hard thresholding. In International Conference on Artificial Intelligence and Statistics, pages 188–198. PMLR, 2018.
61. Jeffrey D Blanchard and Jared Tanner. GPU accelerated greedy algorithms for compressed sensing. Mathematical Programming Computation, 5(3):267–304, 2013.
62. A. Kyrillidis, G. Puy, and V. Cevher. Hard thresholding with norm constraints. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3645–3648. Ieee, 2012.
63. A. Kyrillidis and V. Cevher. Recipes on hard thresholding methods. In 2011 4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), pages 353–356. IEEE, 2011.
64. X. Zhang, Y. Yu, L. Wang, and Q. Gu. Learning one-hidden-layer ReLU networks via gradient descent. In The 22nd International Conference on Artificial Intelligence and Statistics, pages 1524–1534, 2019.
65. Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on information theory, 52(2):489–509, 2006.