1 Compressive sensing

The reason that extreme compression of photographs is possible is the corresponding images are often sparse in the correct basis (e.g., the Fourier or wavelet basis). Thus one can take a very detailed photo and then zero out all the small Fourier coefficients, vastly compressing the image while also preserving the bulk of the important information.

Problematically, despite only recording a small amount of information at the end (say, s large Fourier coefficients), in order to figure out which coefficients to save, we had to perform a very detailed measurement (making our camera pretty expensive). Compressive sensing is the idea that, if we do a few random linear measurements, then we capture the large coefficients without first knowing what they are.

Sparse recovery. Let us formalize the sparse recovery problem. Our signal will be a point $x \in \mathbb{R}^n$, and we will have a linear measurement map $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ that makes m linear measurements, where hopefully $m \ll n$.

Say that a signal $x \in \mathbb{R}^n$ is s-sparse if $\|x\|_0 \leqslant s$, where $\|\cdot\|_0$ denotes the number of non-zero coordinates in its argument. For s-sparse signals x to be uniquely recoverable from the measurements $\Phi(x)$, the following property is necessary and sufficient: For every pair of s-sparse vectors $x, y \in \mathbb{R}^n$, it holds that $\Phi(x) \neq \Phi(y)$.

On the other hand, given the measurements $M = \Phi(x)$, we might want to *recover* the unique corresponding s-sparse vector x. It would be natural to solve the following optimization: $\min \|y\|_0$ subject to $\Phi(y) = M$. Clearly $\|y\|_0 \leqslant \|x\|_0$, so by the unique decoding property for s-sparse vectors and the fact that $\Phi(x) = \Phi(y)$, it must be that $x = y$. Unfortunately, ℓ_0 optimization subject to linear constraints is an NP-hard problem.

Instead, one often solves the problem $\min \|y\|_1$ subject to $\Phi(y) = M$. This is a linear program and can thus be solved efficiently. It is often referred to as *basis pursuit.* Remarkably, if we choose the map Φ appropriately, then the optimum solution y^* will satisfy $x = y^*$, yielding an efficient procedure for sparse recovery.

1.1 The restricted isometry property

We will now formalize the properties of the map $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ that makes efficient sparse recovery possible. For $s \geqslant 1$, let $\delta_s = \delta_s(\Phi)$ be the smallest number such that for every s-sparse vector $x \in \mathbb{R}^n$, we have

$$(1-\delta_s)^2\|x\|_2^2 \leqslant \|\Phi(x)\|_2^2 \leqslant (1+\delta_s)^2\|x\|_2^2 \,. \tag{1.1}$$

It will help to think about this parameter in a slightly different way as well. Let $T \subseteq [n]$ be a subset of s columns of Φ (thought of as an $m \times n$ matrix). Let $\Phi_T : \mathbb{R}^s \to \mathbb{R}^m$ be the linear map corresponding to the matrix which consists of the columns of Φ indexed by T. Then the above property is equivalent to the property that, for every $|T| = s$ and $x \in \mathbb{R}^s$, we have

$$(1-\delta_s)^2\|x\|_2^2 \leqslant \|\Phi_T(x)\|_2^2 \leqslant (1+\delta_s)^2\|x\|_2^2 \,. \tag{1.2}$$

Theorem 1.1. *If $\delta_{2s}(\Phi) < 1$, then Φ has the unique recovery property for s-sparse vectors. If $\delta_{2s}(\Phi) < \sqrt{2}-1$, then the basis pursuit algorithm performs s-sparse recovery.*

Proof. We will only prove the first assertion. If x and x' are s-sparse vectors with $\Phi(x) = \Phi(x')$, then $x - x'$ is a $2s$-sparse vector with $\Phi(x - x') = 0$. But (1.1) yields

$$0 = \|\Phi(x - x')\|_2^2 \geqslant (1 - \delta_{2s})^2 \|x - x'\|_2^2 \,.$$

So for $\delta_{2s} < 1$, we conclude that $x = x'$. □

1.2 Random construction of RIP matrices

Let us define the $m \times n$ random matrix Φ by setting $\Phi_{ij} = \frac{1}{\sqrt{m}} X_i^{(j)}$ where $\{X_i^{(j)}\}$ is a family of i.i.d. $N(0,1)$ random variables. With high probability, this matrix will have the RIP for the parameters chosen appropriately.

Theorem 1.2. *For every $n \geqslant s \geqslant 1$ and $0 < \delta < 1$, there is an $m = O\left(\frac{s}{\delta} \log\left(\frac{n}{\delta s}\right)\right)$ such that with high probability, $\delta_s(\Phi) \leqslant \delta$.*

Proof. Fix a subset $T \subseteq [n]$ with $|T| = s$. We will show that

$$\mathbb{P}\left(\forall x \in \mathbb{R}^s \text{ s.t. } \|x\|_2 = 1, \|\Phi_T(x)\|_2 \in [1 - \delta, 1 + \delta]\right) \geqslant 1 - 2\left(\frac{16}{\delta}\right)^m e^{-\delta m/12} \,. \tag{1.3}$$

Assuming this is true, we can take a union bound over all $|T| = s$, yielding

$$\mathbb{P}\left(\forall s\text{-sparse } x \in \mathbb{R}^n \text{ s.t. } \|x\|_2 = 1, \|\Phi(x)\|_2 \in [1 - \delta, 1 + \delta]\right) \geqslant 1 - 2\left(\frac{16}{\delta}\right)^m e^{-\delta m/12} \binom{n}{s} .$$

Using the fact that $\log\binom{n}{s} = O(s \log(n/s))$, we can conclude that choosing $m = O\left(\frac{s}{\delta} \log\left(\frac{n}{\delta s}\right)\right)$, this probability is at least, say, $1 - 1/n$, and in this case $\delta_s(\Phi) \leqslant \delta$.

Thus we are left to prove (1.3). Let N be a $\delta/4$-net on the unit sphere in $\mathbb{R}^s$. This means that for every $x \in \mathbb{R}^n$ with $\|x\|_2 = 1$, there is an $x' \in N$ with $\|x - x'\| \leqslant \delta/4$. A simple volume argument shows that we can choose such a net N with $|N| \leqslant (4/\delta)^s$.

Now using Claim 1.2 from Lecture 11 (and a union bound over N), we have

$$\mathbb{P}\left(\forall x \in N, \|\Phi_T(x)\|_2 \in \left[1 - \frac{\delta}{4}, 1 + \frac{\delta}{4}\right]\right) \geqslant 1 - 2\left(\frac{4}{\delta}\right)^m e^{-\delta m/48} \,.$$

We are left to show that $\|\Phi_T(x)\|_2 \in \left[1 - \frac{\delta}{4}, 1 + \frac{\delta}{4}\right]$ for all $x \in N$ implies $\|\Phi_T(x)\|_2 \in [1 - \delta, 1 + \delta]$ for all $x \in \mathbb{R}^s$ with $\|x\|_2 = 1$.

This involves another very clever trick. We will define a sequence of points $\{x_i\} \subseteq N$. For any $y \in \mathbb{R}^s$, let $\Gamma(y) = y' \|y\|_2$ where $y' \in N$ is the closest point to $y/\|y\|_2$. Note that by the net property, we have $\|y - \Gamma(y)\|_2 \leqslant \frac{\delta}{4} \|y\|_2$.

Consider $\|x\|_2 = 1$. Define $x_0 = \Gamma(x)$. We can then write $x = x_0 + (x - x_0)$. Now $x_0 \in N$, so we can control $\Phi(x_0)$. Also, $\|x - x_0\|_2 \leqslant \delta/4$. But problematically, we don't have any control on $\Phi(x - x_0)$ (maybe the map Phi expands distances a lot in the direction of $x - x_0$). But the idea is that we can again use the net: Let $x_1 = \Gamma(x - x_0)$. Then $x_1/\|x_1\|_2 \in N$, so we can control $\Phi(x_1)$. Moreover, we have made progress: $\|(x - x_0) - x_1\|_2 \leqslant (\delta/4)^2$.

In the line with the preceding sketch, inductively put $x_{i+1} = \Gamma(x - (x_0 + x_1 + \cdots + x_i))$. Note that

$$x = x_0 + (x - x_0) = x_0 + x_1 + (x - x_0 - x_1) = \cdots = \sum_{i=0}^{\infty} x_i \,,$$

and by induction, $\|x_i\|_2 \leqslant \left(\frac{\delta}{4}\right)^i$, and by construction, $x_i/\|x_i\|_2 \in N$.

Now we can use our assumption that $\|\Phi_T(y)\|_2 \in [1-\delta/4, 1+\delta/4]$ for $y \in N$ to say that

$$\|\Phi_T(x)\|_2 \leqslant \sum_{i=0}^{\infty} \|\Phi_T(x_i)\|_2 \leqslant \left(1+\frac{\delta}{4}\right) \sum_{i=0}^{\infty} \left(\frac{\delta}{4}\right)^i = \left(1+\frac{\delta}{4}\right) / \left(1-\frac{\delta}{4}\right) \leqslant 1+\delta\,,$$

where the last inequality follows from $\delta < 1$. On the other hand,

$$\begin{aligned}\|\Phi_T(x)\|_2 \geqslant \|\Phi_T(x_0)\| - \sum_{i=1}^{\infty} \|\Phi_T(x_i)\|_2 &\geqslant \left(1-\frac{\delta}{4}\right) - \frac{\delta}{4}\left(1+\frac{\delta}{4}\right) \sum_{i=0}^{\infty} (\delta/4)^i \\ &= 1 - \frac{\delta}{4} - \frac{\delta(1+\frac{\delta}{4})}{4(1-\delta/4)} \geqslant 1-\delta\,,\end{aligned}$$

where again we have used $\delta < 1$. We have thus confirmed (1.3), completing the proof. □

Remark 1.3. Note that we must always perform s "measurements" even if we know exactly the s important coordinates. The preceding theorems says that we can do unique (and efficient) recovery with only $O(s\log(n/s))$ measurements without knowing anything about the input signal except that it's s-sparse.

Remark 1.4. In a more realistic model, we might expect that our signal is of the form $x = x_s + y$ where x_s is s-sparse and $\|y\|_2 \leqslant \varepsilon\|x\|_2$. In other words, the signal has s large coordinates plus lower order "noise." The RIP and basis pursuit algorithms can also be used to provide guarantees in this setting.