Towards Convolutional Sparse Models

. [Papyan, Sulam, Elad '17]

Recall that so far, we have been employing models for "local" signals. For eg. every patch from an image X, $R_i X$, admits $R_i X \approx D\gamma_i$, $\|\gamma_i\|_0 \ll n$, where $R_i X \in \mathbb{R}^n$, $X \in \mathbb{R}^N$. $(N \gg n)$.

In fact, when deploying these models, we typically take every overlapping patch, process them more or less independently, and average them back together. This can't be optimal, right?

Several ways to improve on this exist:

- Consider patches accross different resolutions
  eg. Multi-scale K-SVD [Sulam, '15].
- Denoise/Process patches Jointly
  eg. Joint sparse coding [Mairal et. al., Romano et. al].
- Better than just averaging the local estimates
  [Sulam et al. (EPLL sparse), Romano et. al, (Boosting), etc].

More importantly, by imposing a local sparse model on every patch $R_i X$, what is the global model imposed on X?

This is what the Convolutional Sparse Coding model aims to ~~[illegible]~~ answer.

The Convolutional model.

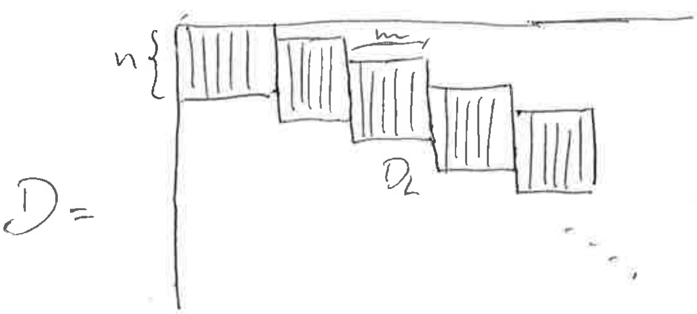
Recall the convolution operation: $(f*g)(t) = \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$; or $(f*g)[n] = \sum_{m=-\infty}^{\infty} f[m]\, g[n-m]$.

Just as before we had $y = D\gamma$, now we will assume that

$$X = \sum_{i=1}^{m} d_i * z_i \in \mathbb{R}^N, \quad d_i \in \mathbb{R}^n \;;\; n \ll N. \quad z_i \in \mathbb{R}^N$$

and where the "feature maps" z_i are sparse: $\|z_i\|_0 \ll N$.
This is equivalent to onstructing the following global dictionary, which will fasilitate some definitions we will need.

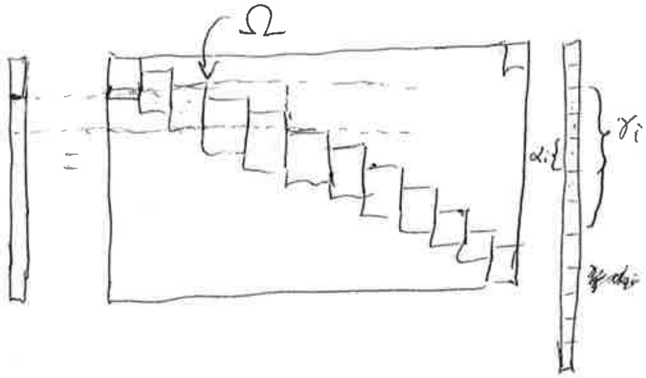
Recalling the operator R_i, define then $D = [R_1^T D_L, R_2^T D_L, \dots, R_N^T D_L]$, where $D_L = [d_1, \dots, d_m]$ is the set of local filters/atoms.



This allows us to write

$$X = D\Gamma$$

where D: global dictionary, and Γ is the global vector that "interlaces" the z_i.



Note that each patch can be expressed as

$$x_i = R_i X = R_i D\Gamma = \underbrace{R_i D S_i^T}\, \underbrace{S_i \Gamma}$$

$$= \Omega\, \gamma_i$$

stripe extractor

$\gamma_i = [\alpha_{i-n+1}, \dots, \alpha_i, \dots, \alpha_{i+n-1}]^T$ stripe dictionary, stripe vector

$\alpha_i \in \mathbb{R}^m$

We can thus write $x_i = \Omega \gamma_i$ for all patches.

$n \times (2n-1)m$

ursuit of Convolutional & Sparse representations.

We now have $X = D\Gamma$, and Γ: sparse.

Recall that all guarantees we've seen depend on some characterization of D. For example, the solution to

$$\min_{\Gamma} \|\Gamma\|_0 \text{ s.t. } X = D\Gamma$$

is unique if $\|\Gamma\|_0 < \frac{1}{2}\left(1 + \frac{1}{\mu(D)}\right)$.

In this convolutional setting, $\mu(D) \geq \sqrt{\frac{m-1}{m(2n-1)-1}} = \Omega(1/\sqrt{n})$.

However, this n is the dimension of the local filters d_i, not the signal dimension! In other words, even if $N = 100$ or 10^{10}, one would allow $\frac{1}{2}\left(1+\frac{1}{\mu(D)}\right)$ (say, 10) non-zeros in Γ. This depicts how all the bounds we've presented lack applicability in this convolutional case, because they allow $O(\sqrt{n})$ non-zeros.

$l_{0,\infty}$ - "norm"

Def: $\|\Gamma\|_{0,\infty} = \max_i \|S_i \Gamma\|_0 = \max_i \|\gamma_i\|_0$.

New Convolut. Pursuit:

$$\min_{\Gamma} \|\Gamma\|_{0,\infty} \text{ s.t. } X = D\Gamma. \qquad (P_{0,\infty})$$

Stripe-Spark: $\sigma_\infty(D) = \min_{\Delta} \|\Delta\|_{0,\infty}$ s.t. $\Delta \in \mathcal{N}(D) \setminus \{0\}$.

We can easily then obtain the corresponding result.

<u>Thm:</u> If $\|\Gamma\|_{0,\infty} < \frac{\sigma_\infty(D)}{2}$, then it is the global optimum of $P_{0,\infty}$.

But so far we don't really know much about $\sigma_\infty(D)$, (let alone computing it). The following lemma resolves this.

<u>Lemma:</u> Given a convolutional dictionary D, with $\mu(D)$, and support S with "$\ell_{0,\infty}$-norm" k (note this is an abuse of notation), consider its Gram $G_S = D_S^T D_S$. Then, its eigenvalues are bounded by

$$1-(k-1)\mu(D) \leq \lambda_i(G_S) \leq 1+(k-1)\mu(D).$$

<u>proof:</u>

From Gershgorin's Theorem, the eigenvalues of G_S lie inside the union of its Gershgorin circles.

The j^{th} circle is centered at G^S_{jj}, and with radius

$r_j = \sum_{i \neq j} |G^S_{j,i}|$. $\quad = 1$ (normalization).

Since all circles will be centered at 1, the eigenvalues of G^S will reside in the circle with largest radius:

$$|\lambda_i(G^S) - 1| \leq \max_j \sum_{i \neq j} |G^S_{j,i}| = \max_j \sum_{\substack{i \neq j \\ i,j \in S}} |d_j^T d_i|.$$

Now, on the one hand, $|d_i^T d_j| \le \mu(D)$. However, this product will be non-zero only for atoms d_i, d_j that overlap with each other. If i, and j are too far, their inner product is zero.

Consider the j^{th} stripe where the maximum (RHS) is atained. Note that only atoms in the j^{th} stripe will have a non-trivially zero inner product. The largest number of atoms in a stripe is $k = \|S\|_{0,\infty}$, thus

$$|\lambda_i(G^S) - 1| \le \max_j \sum_{\substack{i \neq j \\ i,j \in S}} |d_j^T d_i| \le (K-1)\mu(D). \quad //.$$

Note that this now leads to a uniqueness guarante; because we can bound the spark as

$$\sigma_\infty \ge 1 + \frac{1}{\mu(D)}.$$

Why? Because since for the spark we require $D\Delta = 0$, the support of Δ, T, must have a ~~[illegible]~~ enough entries in a stripe so that the Gershgorin's circles include the zero. Thus: $1-(K-1)\mu \le 0 \Rightarrow K = \|\Delta\|_{0,\infty} > 1 + \frac{1}{\mu(D)}$.

And likewise, we can then conclude that

$\ell_{0,\infty}$: $k \le \frac{1}{2}\left(1 + \frac{1}{\mu(D)}\right)$ is sufficient for the uniqueness of a solution $\hat{\Gamma}$.

What about pursuits Algorithm?

Recall that we know that, eg., OMP succeed if the representation is sparse, i.e; if $\|\Gamma\|_0 \leq \frac{1}{2}\left(1 + \frac{1}{\mu(D)}\right)$.

In this convolutional setting we can say something stronger: if $\|\Gamma\|_{0,\infty} \leq \frac{1}{2}\left(1 + 1/\mu(D)\right)$, OMP will recover it from $X = D\Gamma$.

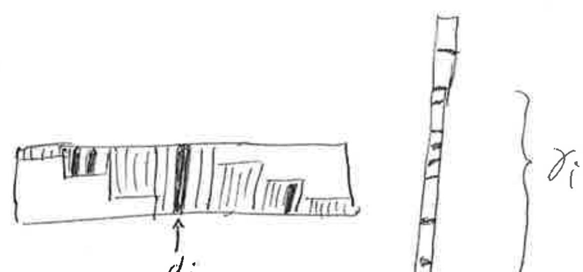
Proof sketch:

Recall that for OMP to succeed, we need it to recover the correct entries at every step. Without loss of generality suppose the max. coeficient (in abs. value) is in the ith coefficient, Γ_i.
For the first step then, we require (sufficient, but not necessary).

$$|d_i^T X| = \left|\sum_{j \in S} \Gamma_j d_j^T d_i\right| > \max_{j \notin S} |d_j^T X| = \max_{j \notin S} \left|\sum_{t \in S} \Gamma_t d_j^T d_t\right|$$

Consider expanding and bounding the left-hand-side,

$$\left|\sum_{j \in S} \Gamma_j d_j^T d_i\right| \geq |\Gamma_i| - |\Gamma_i| \sum_{\substack{j \in S \\ j \neq i}} |d_j^T d_i| .$$



Consider now the stripe centered at the coefficient i, γ_i. One can see that the product $d_i^T d_j$ will be zero for atoms out of the jth stripe. Thus

$$\left|\sum_{j \in S} \Gamma_j d_j^T d_i\right| \geq |\Gamma_i| - |\Gamma_i| \, \|\gamma_i\|_0 \, \mu(D) \geq |\Gamma_i| \left(1 - \|\Gamma\|_{0,\infty} \mu(D)\right)$$

~~What abouts~~

A similar guarantee holds for the Basis Pursuit relaxation of the $(P_{0,\infty})$ problem. We will detail a more general result in the noisy case below.

What about noisy signals? Is the pursuit stable in the convolutional setting?

$$Y = D\Gamma + E. \qquad \text{say } \|E\|_2 \leq \varepsilon.$$

Recall that if we do:

(P_0^ϵ): $\min_\Gamma \|\Gamma\|_0$ s.t. $\|Y - D\Gamma\|_2^2 \leq \varepsilon^2 \rightarrow \hat{\Gamma}$

then $\|\hat{\Gamma} - \Gamma^*\|_2^2 \leq \dfrac{4\epsilon^2}{1 - \mu(D)(2\|\Gamma\|_0 - 1)}$ if $\|\Gamma\|_0 \leq \frac{1}{2}\left(1 + \frac{1}{\mu(D)}\right)$.

Consider instead:

$$(P_{0,\infty}^\epsilon): \min_\Gamma \|\Gamma\|_{0,\infty} \text{ s.t. } \|Y - D\Gamma\|_2^2 \leq \varepsilon^2.$$

How do we $\overset{\text{analyze}}{\text{do}}$ this?

Stripe RIP: Let δ_k be the smallest constant such that

$$(1 - \delta_k)\|\Gamma\|_2^2 \leq \|D\Gamma\|_2^2 \leq (1 + \delta_k)\|\Gamma\|_2^2$$

$$\forall \ \Gamma: \|\Gamma\|_{0,\infty} = k.$$

Guess what?

$$\delta_k \leq (k-1)\mu(D) \ !$$

because
$\|D_s\Gamma\|_2^2 \leq \lambda_{max}(D_s^T D_s)\|\Gamma\|_2^2$
Lemma 1 $\rightarrow \leq (1 + \mu(k-1))\|\Gamma\|_2^2$
$\delta_k \leq \mu(k-1)$.

And so, if $\hat{\Gamma} = \arg\min_\Gamma (P_{0,\infty}^\epsilon)$, if $\|\Gamma\|_{0,\infty} \leq k$, then

$$\|\hat{\Gamma} - \Gamma\|_2^2 \leq \frac{4\epsilon^2}{1 - \mu(D)(2\|\Gamma\|_{0,\infty} - 1)} \ll \frac{4\epsilon^2}{1 - \mu(2\|\Gamma\|_0 - 1)}.$$

In these cases, OMP is also stable in the convolutional setting. We bring here a stability analysis for BPDN:

Consider (P_1^λ): $\min_\Gamma \|Y - D\Gamma\|_2^2 + \lambda \|\Gamma\|_1$.

and assume $Y = D\Gamma^* + E$, $\|E\|_2 \leq \varepsilon$, and $\|\Gamma\|_{0,\infty} = k < \frac{1}{3}\left(1 + \frac{1}{\mu(D)}\right)$.

<u>Thm:</u> If λ is set properly $(\lambda = 4\varepsilon_L = 4 \max_i \|R_i E\|_2)$
then, $\hat{\Gamma} = \arg\min (P_1^\lambda)$,

1) $supp(\Gamma^*) = supp(\hat{\Gamma})$

2) $\|\hat{\Gamma} - \Gamma^*\|_\infty \leq \frac{15}{2} \varepsilon_L$

3) $\hat{\Gamma}$ contains every index i : $|\Gamma_i^*| > \frac{15}{2} \varepsilon_L$.

4) $\hat{\Gamma}$: unique.

Note that these results are a lot more informative in this convolutional setting, as $\varepsilon_L \ll \varepsilon$.

<u>How to compute the solution to (P_1^λ)?</u> Do not compute by D!

the algorithm

$$\Gamma^{k+1} \leftarrow S_{\lambda/c}\left(\Gamma^k - \frac{1}{c} D^T (D\Gamma^k - Y)\right)$$

is equivalent to

$O(N^2 m)$

- local coding:
  $\alpha_i^{k+1} = S_{\lambda/c}\left(\alpha^k + \frac{1}{c} D^T r_i^k\right)$
- $\hat{X}^{k+1} = \sum_i R_i^T D_L \hat{\alpha}^{k+1}$ (aggregation)
- $r_i^{k+1} = R_i (X^{k+1} - Y)$

-local residuals.

$O(Nnm)$.

What about learning the convolutional dictionaries? (5)

We are after $\min_{D,\Gamma} \frac{1}{2}\|X - D\Gamma\|_2^2 + \lambda\|\Gamma\|_1$

But recall that $D: N \times mN$, and structured!

First methods employed Fourier type approaches, to circumvent complexity limitations. They are often complicated to implement, and scales as $O(N\log(N))$.

Here we bring an alternative that relies on local processing, thus being able to employ all the Dict. Learning methods we've seen so far, and that scales linearly with global dimension N.

Sliced-Based Convol. Dict. Learning [Papyan et al, '17].

We will employ the Alternating Directions Methods of Multipliers. (ADMM).

(super)
Brief ADMM recap:

if we want to solve: $\min_x f(x) + g(x)$, and its hard, consider the split:

$$\min_{x,z} f(x) + g(z) \quad \text{s.t.} \quad x = z.$$

Lagrange multipliers: $\min_{x,z,\lambda} f(x) + g(z) + \lambda^T(x-z)$.

Augmented Lagrange: $\min_{x,z,\lambda} f(x) + g(z) + \lambda^T(x-z) + \frac{\rho}{2}\|x-z\|_2^2$.

which can be combined

as: $\min_{x,z,u} f(x) + g(z) + \frac{\rho}{2}\|x - z + u\|_2^2 = L(x,z,u)$

The benefit is that now each of the inner problems are much simpler

$$\begin{cases} x^{k+1} = \arg\min_x L(x, z^k, u^k) \\ z^{k+1} = \arg\min_z L(x^{k+1}, z, u^k) \\ u^{k+1} = u^k + (x^{k+1} - z^{k+1}). \end{cases}$$

ADMM converges with minimal assumptions of f, g (convex, closed, proper).

$$x^k - z^k \to 0, \quad f(x^k) + g(z^k) \to F_{opt}.$$

Back to Conv. Dict. Learning:

We'll modify the problem:

$$\min_{\Gamma} \frac{1}{2} \|X - D\Gamma\|_2^2 + \lambda \|\Gamma\|_1$$

$$\min_{\alpha_i} \frac{1}{2} \|X - \sum_{i=1}^{N} R_i^T D_L \alpha_i\|_2^2 + \lambda \sum_{i=1}^{N} \|\alpha_i\|_1$$

$$\min_{\alpha_i, s_i} \frac{1}{2} \|X - \sum_{i=1}^{N} R_i^T s_i\|_2^2 + \lambda \sum_i \|\alpha_i\|_1 \quad \text{s.t.} \quad s_i = D_L \alpha_i$$

slices!

$$\Rightarrow \min_{\alpha_i, s_i, u_i} \frac{1}{2} \|X - \sum_i R_i^T s_i\|_2^2 + \lambda \sum_{i=1}^{N} \|\alpha_i\|_1 + \frac{\rho}{2} \|s_i - D_L \alpha_i + u_i\|_2^2.$$

through ADMM:

$$\begin{cases} \alpha_i^{k+1} = \arg\min_{\alpha_i} \frac{\rho}{2} \|(s_i^k + u_i^k) - D_L \alpha_i\|_2^2 + \lambda \|\alpha_i\|_1 & \rightarrow \text{low dimensional Lasso!} \ \boxtimes \\ \{s_i^{k+1}\} = \arg\min_{\{s_i\}} \frac{1}{2} \|X - \sum_i R_i^T s_i\|_2^2 + \sum_i \frac{\rho}{2} \|s_i^k - (D_L \alpha_i^{k+1} + u^k)\|_2^2 \\ \quad \Rightarrow \text{closed form solution.} \\ u_i^{k+1} = u_i^k + s_i^{k+1} + u_i^k. \end{cases}$$

What's better, we can also update for the dictionary (local) D_L in step $\boxtimes$, which involves only low dimensional patches, and we can use any previous dict. learning method:

$$(\{\alpha_i\}, D_L) = \underset{\alpha_i, D_L}{\operatorname{argmin}} \sum_i \frac{\rho}{2} \| (S_i^k + u_i^k) - D_L \alpha_i \|_2^2 + \lambda \|\alpha_i\|_1 .$$