Lecture 25: Sparse Signal/image recovery

In the last lecture we saw an application of l_1-regularization (more broadly, sparse models) in the context of machine learning & regression.

In this lecture we consider a different application in the context of signal processing where "sparsity" will help us in solving remarkable problems.

25.1 The signal recovery problem

Consider a signal (or image) $f^\dagger : [0,1] \to \mathbb{R}$ along with some measurements of the signal $\underline{y} \in \mathbb{R}^M$.

A simple example is

$$\underline{y}^\dagger = (y_0^\dagger, \dots, y_{N-1}^\dagger) \text{ where } y_j^\dagger = f^\dagger(t_j)$$

for some points $t_j \in [0,1]$.

Of course we can simply solve this problem using, for ex, polynomial interpolation.

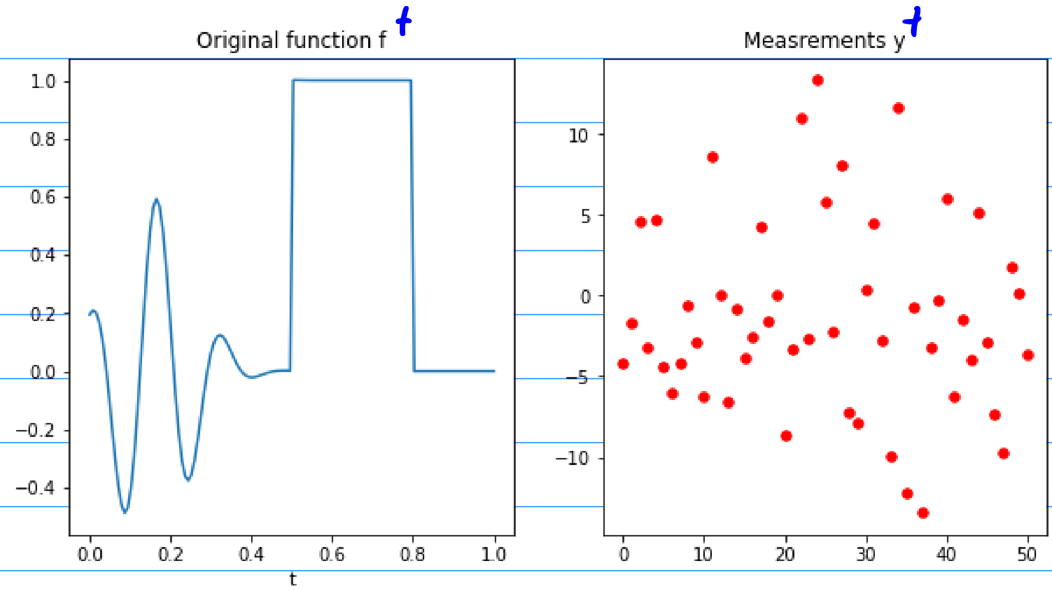
But we want to consider a harder problem specifically one where the measurements $\underline{y} \in \mathbb{R}^M$ are of the form

$$y_j^t = \underline{\xi}_j^T \underline{f}^t,$$

where, $\underline{\xi}_j \sim N(0, I)$ & $\underline{f}^t = (f^t(t_0), f^t(t_1), \ldots, f^t(t_{N-1}))$

Goal: reconstruct f^t given $\underline{y}^t$!

& $M < N$!



Of course the true f^t cannot be found exactly since we have access to limited measurements $\underline{y}^t$. But the question is can we find an approx. $\hat{f}$ that is close to f^t!

Our first step is to assume a model for $\hat{f}$

$$f(t) = \sum_{m,n} \beta_{m,n} \psi_{m,n}(t),$$

where $\psi_{m,n}$ are the Haar wavelet basis.

Reparameterize the wavelet coeffs. into a 1D array,

$$f(t) = \sum_{j=0}^{J-1} \beta_j \psi_j(t).$$

Then, we simply have

$$f(t_n) = \sum_{j=0}^{J-1} \beta_j \psi_j(t_n),$$

In other words,

$$\underline{f} = (f(t_0), \ldots, f(t_{N-1})), \quad \underline{f} = W\underline{\beta}$$

where

$$\mathbb{R}^{N\times J} \ni W = \begin{bmatrix} \psi_0(t_0) & \psi_1(t_0) & & \psi_{J-1}(t_0) \\ \psi_0(t_1) & \psi_1(t_1) & & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ \psi_0(t_{N-1}) & \psi_1(t_{N-1}) & & \psi_{J-1}(t_{N-1}) \end{bmatrix}$$

Finally for such an $\underline{f}$ we have the measurments

$$\underline{y} = S\underline{f}, \quad S = \begin{bmatrix} \underline{\xi}_0^T \\ \hline \underline{\xi}_1^T \\ \hline \vdots \\ \underline{\xi}_{M-1}^T \end{bmatrix} \in \mathbb{R}^{M \times N}$$

In other words

$$\underline{y} = A\underline{\beta} \quad \text{where } A = SW.$$

Then this is nothing but a supervised learning/regression problem!

find $\hat{\beta}$ st. $A\underline{\hat{\beta}} \approx \underline{y}^{+}$!

But there is an important distinction.

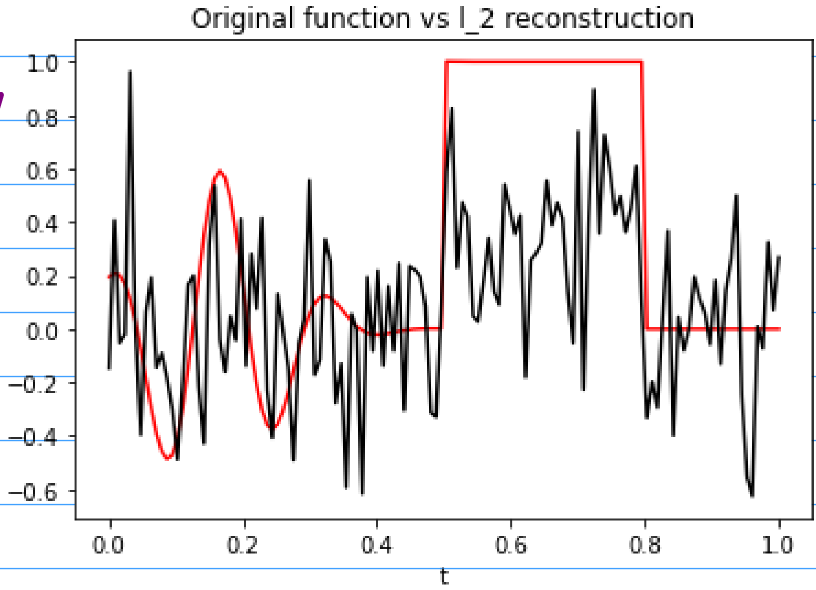
$M < N$, ie, there are more parameters than data! & the data $\underline{y}^{f}$ is correlated!

so there is no point in trying least squares.

First we may try Ridge

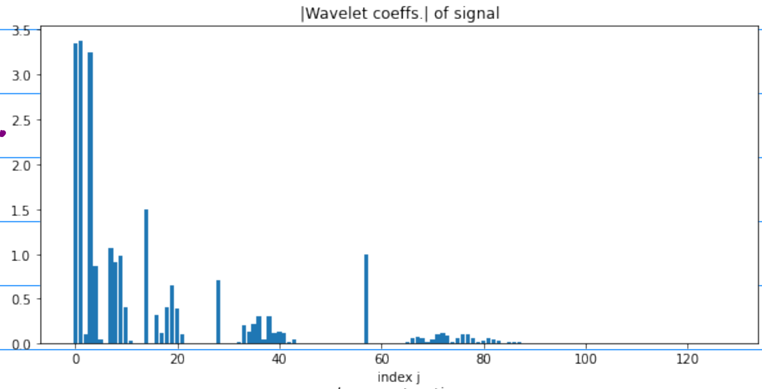
$$\underset{\underline{\beta}\in\mathbb{R}^J}{\text{minimize}} \; \|A\underline{\beta}-\underline{y}\|_2^2+\lambda\|\beta\|_2^2$$

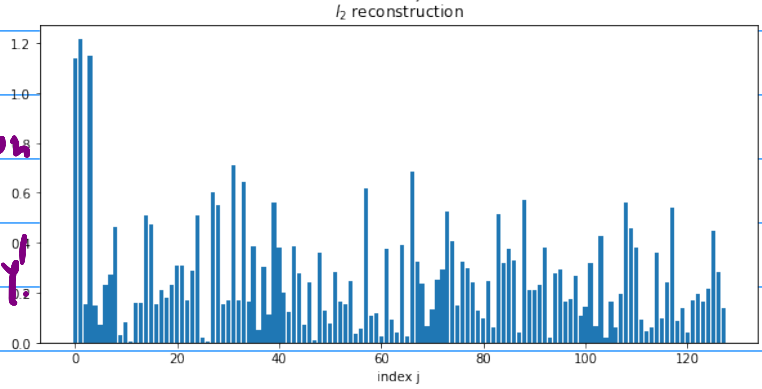
Not good at all! has no bearing to OG function!



Observation:

$f^\dagger$ has sparse coeffs. in the wavelet basis.



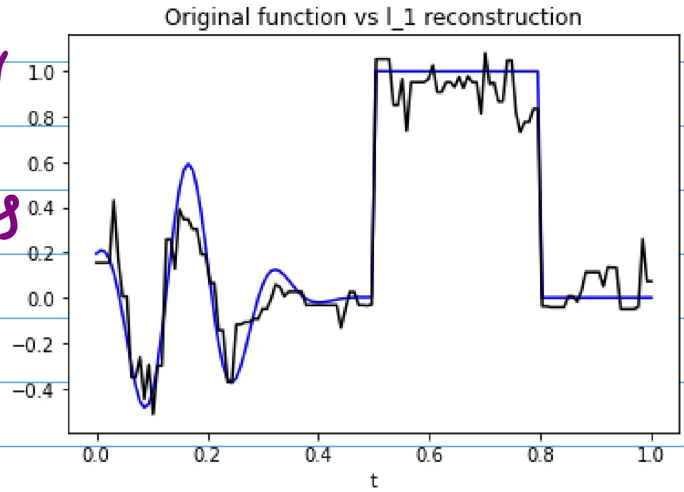


the Ridge (ℓ_2) solution does not exhibit sparsity!

Let us now try a 1-norm penalty. To see if sparsity can help!

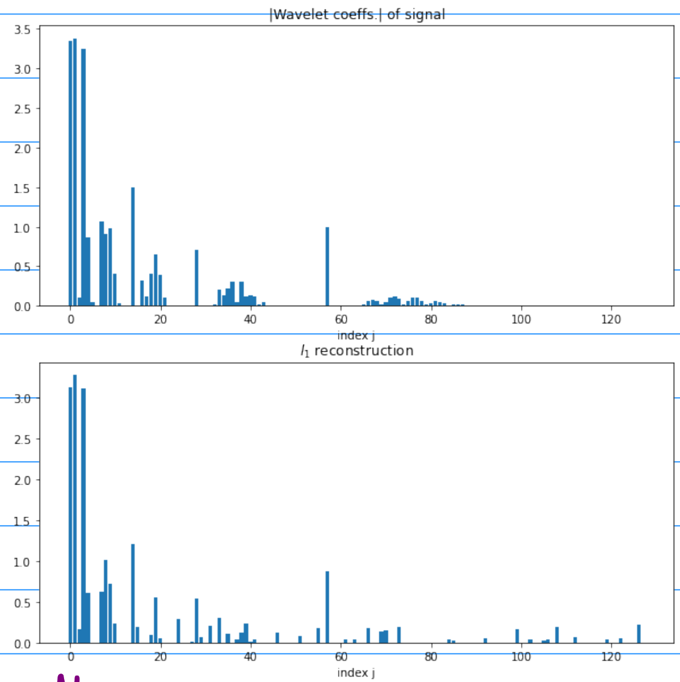
$$\underset{\underline{\beta}}{\text{minimize}} \; \|A\underline{\beta} - \underline{y}\|_2^2 + \lambda \|\underline{\beta}\|_1$$

- much better solution! a bit noisy but captures main trends in f^t.



- Lasso (l_1-penalty) solution has sparse coeffs.

- The problem is solvable if we look for a sparse solution rather than considering all possible solutions!



- sparse recovery is a powerful tool!
It has revolutionized signal processing, medical imaging, & geosciences.

- It is particularly useful if you knew something about the original signal/image f^t. For example most natural images are sparse in wavelet bases. Audio signals are often sparse in Fourier basis, etc.