MINIPROJECT ON SPARSITY AWARE LEARNING

Anders S. Olsen (s154043) & Lena M. Nilsson (s153554)

DTU Health Tech

ABSTRACT

In this project, sparse representations of sound signals are explored. A sound signal was created using two strings on a guitar played after one another. We determine the power spectrum and perform Least Absolute Shrinkage Selector Operator (LASSO) regression to represent the spectrum sparsely. We attempt to reconstruct the sound signal using the sparse representation and find a loss of time dependence, resulting in the two frequencies being present constantly in the signal. Next, a short time Fourier transform (STFT) is computed enabling a reconstruction with proper time resolution.

1. INTRODUCTION

In machine learning, a branch of methods used for distinguishing signal from no signal is called sparse learning. This branch utilizes so-called sparse representation of signals, where a dictionary is constructed including only the non-zero elements of the signal.

In this miniproject, aspects of sparse learning using LASSO regression are explored using a sound signal recorded from a guitar. More specifically, we are looking for sparse representations of this signal using the Fourier transform.

2. METHODS

The Fourier transform provides explicit information on the frequency content of a signal. The discrete Fourier transform (DFT) is given in matrix form as

$$\mathbf{y} = M\mathbf{x}, \tag{1}$$

where $\mathbf{x}$ is the time signal, $\mathbf{y}$ is the corresponding Fourier transform and the elements of matrix M are given as $M_{kn} = e^{-i2\pi kn/N}$. The matrix M is often very large, and for implementation purposes the MATLAB function *fft* is used instead.

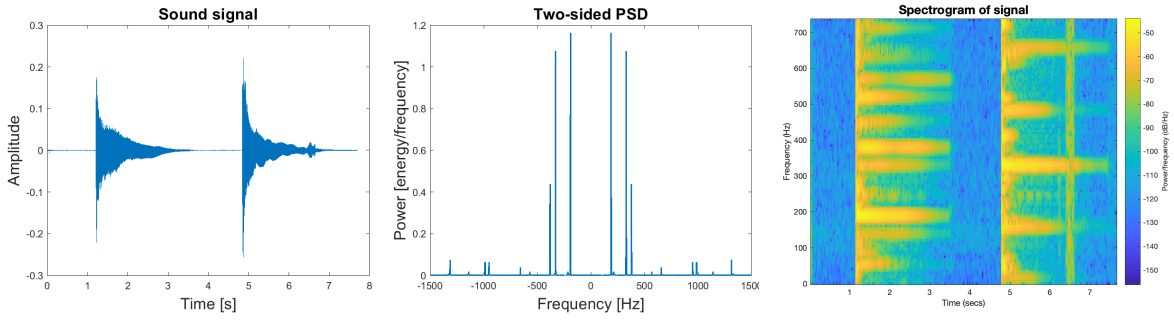
Using the standard linear regression model $\mathbf{y} = X\boldsymbol{\theta} + \boldsymbol{\eta}$, a simple least squares solution is defined as:

$$\hat{\boldsymbol{\theta}}_{LS} = \underset{\theta}{\operatorname{argmin}} \sum_{n=0}^{N} (y_n - \boldsymbol{\theta}^T \mathbf{x}_n)^2 \tag{2}$$

This model has the closed-form solution $\hat{\theta}_{LS} = (X^T X)^{-1} X^T \mathbf{y}$, but often leads to overfitting. This inspired the development of the Ridge regression model that penalizes large values of the L2-norm of the parameter vector $\boldsymbol{\theta}$. This forces the model to shrink the weights, albeit not to zero. Therefore, individual features will have less impact in the model, but will remain nonzero. In sparsity aware learning, solutions where a majority of elements are zero are the goal. A possible remedy is to penalize the L1-norm instead:

$$\hat{\boldsymbol{\theta}}_1 = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \sum_{i=0}^{n} (y_i - \boldsymbol{\theta}^T \mathbf{x}_i)^2 + \lambda ||\boldsymbol{\theta}||_1 \tag{3}$$

This forces the model to shrink the weights for the least useful features to zero, thereby acting as a feature selection method. The LASSO has no closed-form solution, however, since the 1-norm is non-differentiable in $\boldsymbol{\theta} = \mathbf{0}$. Instead, the subgradient, which is equal to the gradient in all differentiable points and then a random value between -1



(a) (b) (c)

Fig. 1: (a) Signal created with a guitar. (b) Two-sided power spectrum of audio signal. (c) Spectrogram of audio signal.

and 1 in $\boldsymbol{\theta} = \mathbf{0}$, is used. In the case $X^T X = I$, the following expression is valid:

$$\hat{\theta}_{1,i} = \text{sgn}(\hat{\theta}_{LS,i})(|\hat{\theta}_{LS,i}| - \frac{\lambda}{2})_+, \qquad (4)$$

where $i = 1, 2, ..., l$. In practice, Equation 4 works as a thresholding function for the least squares regression. In the case $\boldsymbol{\theta} = \mathbf{0}$, the second term in Equation 4 will always be negative for $\lambda > 0$, thus eliminating the need for random values in this point. Also, since the L1 norm is not strictly convex, the need for iterative regularization methods is also bypassed using Equation 4. Other types of norms that could be used for sparsity include those with p<1 that, like LASSO, are good for shrinking weights to zero, but lack the trait of convexity, which is desired in order to be able to optimize the weights.

3. PROCESSING

The signal used in this miniproject was created using a guitar and a laptop microphone with sampling rate $F_s = 44100Hz$. The G-string on the guitar was played and then brought to a standstill, after which the high E-string was played and brought to a standstill. The recorded signal is loaded into MATLAB, and visualized as seen in Figure 1a.

The two-sided power spectrum of the signal can be seen in Figure 1b, and immediately, three peaks on each side of the origin are noticed. The two highest ones are thought to be equivalent to the two dominant frequencies provoked by the guitar strings, while the lowest peak most likely is a harmony from the first peak. More specifically, the dominant frequencies are located at 189.6 Hz and $329.2Hz$. These correspond to the G and E strings, which typically have the frequencies $196.0Hz$ and $329.6Hz$ (*source*), which fits quite well with the frequencies identified. To investigate the frequency evolution over time, a spectrogram of the signal is likewise created, and can be seen in Figure 1c. Here, the same dominant frequencies as in Figure 1b are visible, with the first guitar string being around 190 Hz with visible harmonics, while the second guitar string has a central frequency of around 330 Hz with visible harmonics.

Next, LASSO regression is performed on the Fourier transformed signal using $X = I$ to implement a sparse version of the audio signal. The resulting power spectrum after implementing LASSO regression with $\lambda = 10^3$ can be seen in Figure 2, while the corresponding reconstruction using *ifft* can be seen in Figure 3.

In Figure 2 it is clear that the LASSO regression is successful, as the resulting power spectrum contains only the two desired frequencies, while the remaining frequency contents have been shrunk to

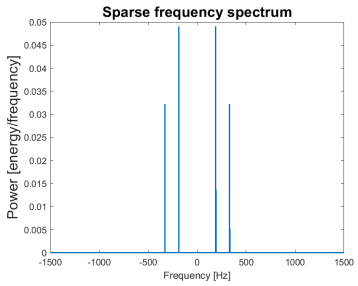


Fig. 2: Two-sided power spectrum after LASSO regression.

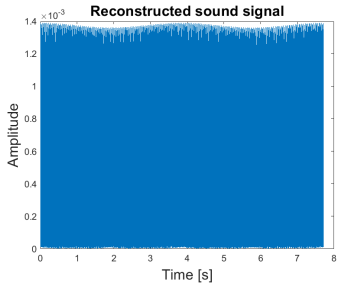


Fig. 3: Reconstructed sound signal after LASSO regression.

zero. Mind that the spectrum is two-sided. However, as evident in Figure 3, the two identified frequencies are both constantly present in the signal. If played, it sounds as if the two strings were plucked at the same time with the amplitude held constant. This is due to the removed frequencies that combined constitute the shape of the input signal. Some amplitude difference is however still present in this reconstructed signal, which may be due to low-frequency spectral leakage caused by the fact that the sparse solution represents ideal window-filters in the frequency domain, which then become sinc functions when converted to the time domain.

In order to reconstruct a signal that gives the desired time-dependent output, namely the sound of two guitar strings separately, LASSO regression is instead performed on the short-time Fourier transform of the signal. The signal is reconstructed using *ifft* on each time window, and the result using a window size of $\frac{Fs}{10}$ with a skip size of $\frac{Fs}{20}$ can be seen in Figure 4.

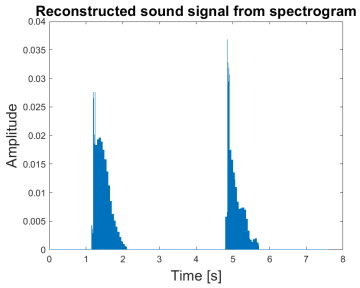


Fig. 4: Reconstructed audio signal after LASSO regression on spectrogram.

In order to achieve an even better reconstruction, prior knowledge of the properties of the signal can be taken into account. In order to limit the sparse spectrum for each time window to only approximately two frequencies, the λ-value would need to be quite high, penalizing the smaller frequencies. However, in many of the time windows, this would result in a shrinkage of the desired frequencies to zero as well, shortening the time period, where each of the strings are audible. To avoid this, the prior knowledge that only one frequency is desired in each frequency spectrum, it is possible to select the largest frequency value in the spectrum and set the rest to zero, while maintaining a low λ-value. Naturally, as the spectrum is two-sided, the same frequency is chosen from each side.

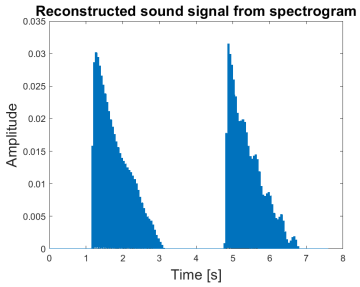


Fig. 5: Reconstructed audio signal after LASSO regression on spectrogram using prior knowledge.

As a result, it is possible to distinguish the two guitar strings played from the reconstructed signal.