Source Separation Using NMF

Sebastian Ament and Aarti Bagul

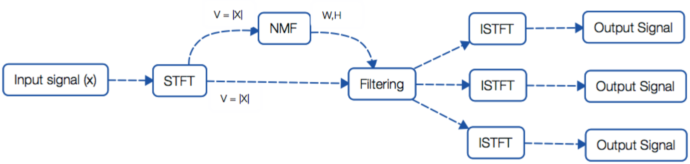
December 16, 2015

Table of contents

Introduction

- **Source Separation** - Estimating signal produced by single source from a mixture of sources
- **Unsupervised** learning
- eg: cocktail party problem, separating sounds of musical instruments
- Using Non-negative Matrix factorization and Ikatura-Saito divergence

Pipeline


Input signal (x)
STFT
V = |X|
NMF
W,H
V = |X|
Filtering
ISTFT
ISTFT
ISTFT
Output Signal
Output Signal
Output Signal

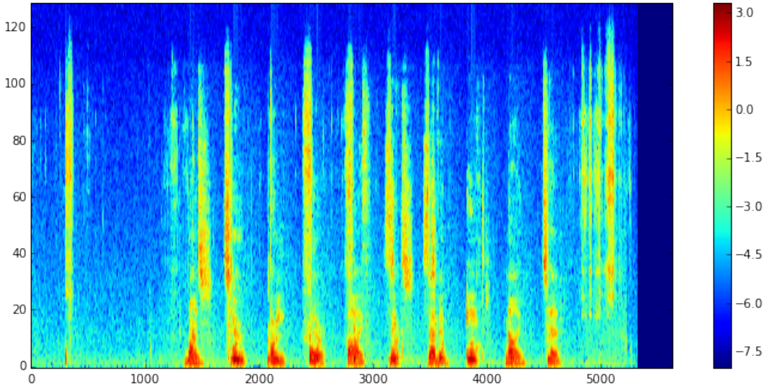
Data set

1. Piano recording of ”Mary had a little lamb, 5 seconds long”,
2. Recording of 2 people talking at the same time, 20 seconds long
3. 2 Jazz recordings with multiple instruments, 2 minutes long

Signal Processing

1. Input is a .wav soundfile
2. Apply Short-Time Fourier Transform
3. Sine window with a width of 256 samples and 128 samples overlap
4. Compute magnitude squared of Fourier coefficients to get power spectrogram

Spectrogram



NMF

- **Non-negative matrix factorization:**

$$V = WH$$

where $V \in \mathbb{R}_+^{F \times N}$, $W \in \mathbb{R}_+^{F \times K}$, and $H \in \mathbb{R}_+^{K \times N}$.

- **Itakura-Saito Divergence**

$$d_{IS}(x, y) = \sum_i \frac{x_i}{y_i} - \log \frac{x_i}{y_i} - 1$$

We want to minimize

$$d(v, Wh) = \sum_i \frac{v_i}{(Wh)_i} - \log \frac{v_i}{(Wh)_i} - 1$$

Online Algorithm

Online Algorithm for IS-NMF

Input Training set, $W^{(0)}$, $A^{(0)}$, $B^{(0)}$, ρ, β, η, ϵ

repeat

    t $\leftarrow$ t+1

    draw point v_t from training set.

    $h_t \leftarrow \arg\min_h d_{IS}(\epsilon + v_t, \epsilon + Wh)$

    $a^{(t)} \leftarrow (\frac{\epsilon+v_t}{(\epsilon+Wh_t)^2} h_t^T).W^2$

    $b^{(t)} \leftarrow \frac{1}{(\epsilon+Wh_t)^2} h_t^T$

    if $t \equiv 0[\beta]$

        $A^{(t)} \leftarrow A^{(t-\beta)} + \rho \sum_{s=t-\beta+1}^{t} a^{(s)}$

        $B^{(t)} \leftarrow B^{(t-\beta)} + \rho \sum_{s=t-\beta+1}^{t} b^{(s)}$

        $W^{(t)} \leftarrow \sqrt{\frac{A^{(t)}}{B^{(t)}}}$

        for k = 1...K

            $s \leftarrow \sum_f W_{fk}$,

            $W_{fk} \leftarrow W_{fk}/s$

            $A_{fk} \leftarrow A_{fk}/s$,

            $B_{fk} \leftarrow B_{fk}$ x s

        end for

Gradient Backtracking

- Gradient:

$$\nabla_h d_{IS}(v_t, Wh) = \frac{1}{Wh}\left(\vec{1} - \frac{1}{Wh}\right) \cdot W$$

- For gradient descent with backtracking, we want:

$$\epsilon = \beta^m \eta$$

  where, m = max number of backtracking steps, η = the initial step size,β = backtracking factor, and ϵ = the precision of the optimization.

- This yields

$$m(\beta) = \frac{\log \epsilon - \log \eta}{\log \beta}$$

Gradient Backtracking

β	m	t	$\mathbb{E}(backsteps)$	$\mathbb{E}(s)$	max s
.1	16	3.5×10^{-4}	9	1.0×10^{-10}	1.0
.2	23	2.1×10^{-4}	11.5	9.1×10^{-10}	.5
.5	53	4.9×10^{-4}	26.5	1.4×10^{-9}	.2
.8	165	1.4×10^{-3}	82.5	1.9×10^{-9}	.125

s is computed backtracking constant and
$\mathbb{E}(\cdot)$ is the empirical mean of "$\cdot$".
m is the maximum number of backtracking steps
β is the backtracking factor,

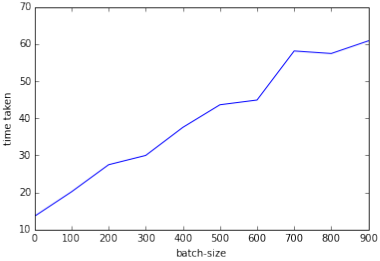
Analysis of parameters

Impact of β parameter on performance of NMF.

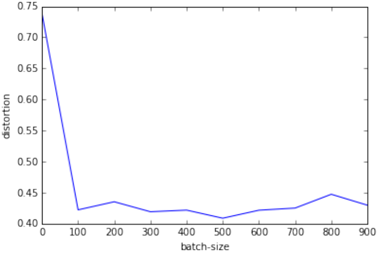
β	$t[sec]$	$d_{IS}(V, WH)$
2^6	3.9×10^0	3.8×10^9
2^7	7.6×10^0	7.4×10^8
2^8	1.4×10^1	2.5×10^6
2^9	2.0×10^1	2.4×10^6
2^{10}	3.9×10^1	2.2×10^6
2^{11}	9.1×10^1	2.3×10^6

t is the time until the stopping criterion is met, $d_{IS}(V, WH)$ the IS-divergence of the last iteration.

Mini-batch KMeans: Varying batch size

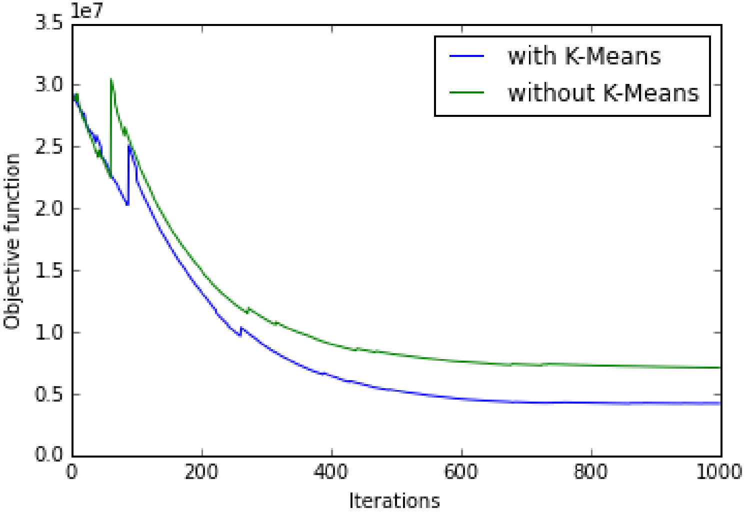


Batch size v/s time



Batch size vs k-means objective

Mini-batch KMeans



Smoothing

Smoothness term measures the deviation to the adjacent columns of H

$$S_{\lambda,t}(h) = \lambda \left[d_{IS}(H_{\cdot(t-1)}, h) + d_{IS}(H_{\cdot(t+1)}, h) \right]$$

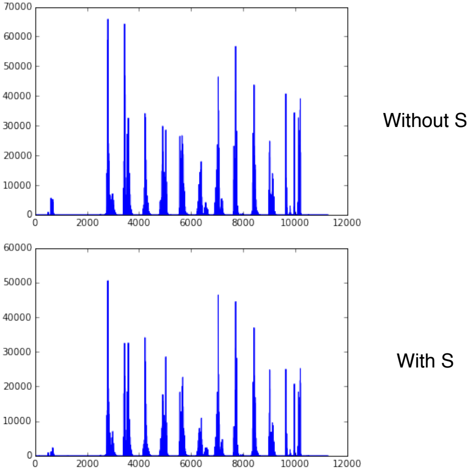
with corresponding gradient step

$$\nabla_h S_{\lambda,t}(h) = \lambda \left[\frac{1}{H_{\cdot(t-1)}} + \frac{1}{H_{\cdot(t+1)}} - 2\frac{1}{h} \right]$$

Our new minimization problem is given by

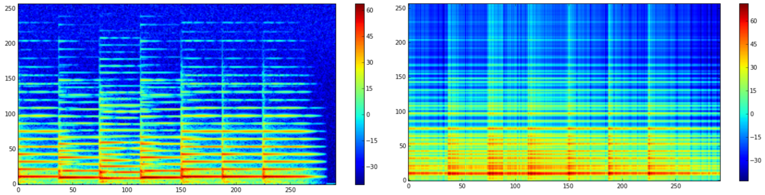
$$h_t = \arg\min_h \left[d_{IS}(v_t, Wh) + S_{\lambda,t}(h) \right]$$

Local Smoothing



One row of H

Approximated Spectrogram with $K = 3$



- IS divergence after last iteration$= 2x10^5$
- While the reproduction of the spectrogram works to a satisfactory degree, meaningful source separation was not possible

Conclusion

- Our implementation of the online IS-NMF exhibits higher performance compared to batch IS-NMF
- The initialization of W using k-means clustering of the input spectrogram speeds up the convergence of the NMF algorithm
- We proposed a new objective function for online IS-NMF, including a locally acting smoothness term
- This might be extended to a mini-batch version of the online algorithm, with a term that considers the smoothness of a whole batch instead of just one sample