Chapter

Speech Enhancement Using an Iterative Posterior NMF

Sunnydayal Vanambathina

Abstract

Over the years, miscellaneous methods for speech enhancement have been proposed, such as spectral subtraction (SS) and minimum mean square error (MMSE) estimators. These methods do not require any prior knowledge about the speech and noise signals nor any training stage beforehand, so they are highly flexible and allow implementation in various situations. However, these algorithms usually assume that the noise is stationary and are thus not good at dealing with nonstationary noise types, especially under low signal-to-noise (SNR) conditions. To overcome the drawbacks of the above methods, nonnegative matrix factorization (NMF) is introduced. NMF approach is more robust to nonstationary noise. In this chapter, we are actually interested in the application of speech enhancement using NMF approach. A speech enhancement method based on regularized nonnegative matrix factorization (NMF) for nonstationary Gaussian noise is proposed. The spectral components of speech and noise are modeled as Gamma and Rayleigh, respectively. We propose to adaptively estimate the sufficient statistics of these distributions to obtain a natural regularization of the NMF criterion.

Keywords: nonnegative matrix factorization (NMF), speech enhancement, signal-to-noise ratio (SNR), expectation maximization (EM) algorithms, posterior regularization (PR)

1. Introduction

Over the past several decades, there has been a large research interest in the problem of single-channel sound source separation. Such work focuses on the task of separating a single mixture recording into its respective sources and is motivated by the fact that real-world sounds are inherently constructed by many individual sounds (e.g., human speakers, musical instruments, background noise, etc.). While source separation is difficult, the topic is highly motivated by many outstanding problems in audio signal processing and machine learning, including the following:

1. Speech denoising and enhancement—the task of removing background noise (e.g., wind, babble, etc.) from recorded speech and improving speech intelligibility for human listeners and/or automatic speech recognizers

2. Content-based analysis and processing—the task of extracting and/or processing audio based on semantic properties of the recording such as tempo, rhythm, and/or pitch

3. Music transcription—the task of notating an audio recording into a musical representation such as a musical score, guitar tablature, or other symbolic notations

4. Audio-based forensics—the task of examining, comparing, and evaluating audio recordings for scientific and/or legal matters

5. Audio restoration—the task of removing imperfections such as noise, hiss, pops, and crackles from (typically old) audio recordings

6. Music remixing and content creation—the task of creating a new musical work by manipulating the content of one or more previously existing recordings

2. Nonnegative matrix factorization

2.1 NMF model

Nonnegative matrix factorization is a process that approximates a single nonnegative matrix as the product of two nonnegative matrices. It is defined by

$$V \approx WH \tag{1}$$

$V \in R_+^{N_f \times N_t}$ is a nonnegative input matrix. $W \in R_+^{N_f \times N_z}$ is a matrix of basis vectors, basis functions, or dictionary elements; $H \in R_+^{N_z \times N_t}$ is a matrix of corresponding activations, weights, or gains; and N_f is the number of rows of the input matrix. N_t is the number of columns of the input matrix; N_z is the number of basis vectors [1].

$V \in R_+^{N_f \times N_t}$—original nonnegative input data matrix

- Each column is an N_f -dimensional data sample.
- Each row represents a data feature.

$W \in R_+^{N_f \times N_z}$ —matrix of basis vectors, basis functions, or dictionary elements.

- A column represents a basis vector, basis function, or dictionary element.
- Each column is not orthonormal, but commonly normalized to one.

$H \in R_+^{N_z \times N_t}$ —matrix of activations, weights, encodings, or gains.

- A row represents the gain of a corresponding basis vector.
- Each row is not orthonormal, but sometimes normalized to one.

When used for audio applications, NMF is typically used to model spectrogram data or the magnitude of STFT data [2]. That is, we take a single-channel recording, transform it into the time-frequency domain using the STFT, take the magnitude or power V, and then approximate the result as $V \approx WH$. In doing so, NMF approximates spectrogram data as a linear combination of prototypical frequencies or spectra (i.e., basis vectors) over time.

This process can be seen in **Figure 1** [3], where a two-measure piano passage of “Mary Had a Little Lamb” is shown alongside a spectrogram and an NMF factorization. Notice how **W** captures the harmonic content of the three pitches of the passage and **H** captures the time onsets and gains of the individual notes. Also note that N_z is typically chosen manually or using a model selection procedure such as cross-validation and N_f and N_t are a function of the overall recording length and STFT parameters (transform length, zero-padding size, and hop size).

This leads to two related interpretations of how NMF models spectrogram data. The first interpretation is that the columns of V (i.e., short-time segments of the mixture signal) are approximated as a weighted sum of basis vectors as shown in **Figure 2** and Eq. (2):

$$V \approx \begin{bmatrix} | & | & | & & | \\ V1 & V2 & V_3 & \ldots\ldots & V_{N_t} \\ | & | & | & & | \end{bmatrix} \approx \left[\sum_{j=1}^{N_z} H_{j1} W_j \;\; \sum_{j=1}^{K} H_{j2} W_j \;\; \sum_{j=1}^{K} H_{jN_t} W_j\right] \qquad (2)$$

The second interpretation is that the entire matrix V is approximated as a sum of matrix “layers,” as shown in **Figure 3** and Eq. (3).

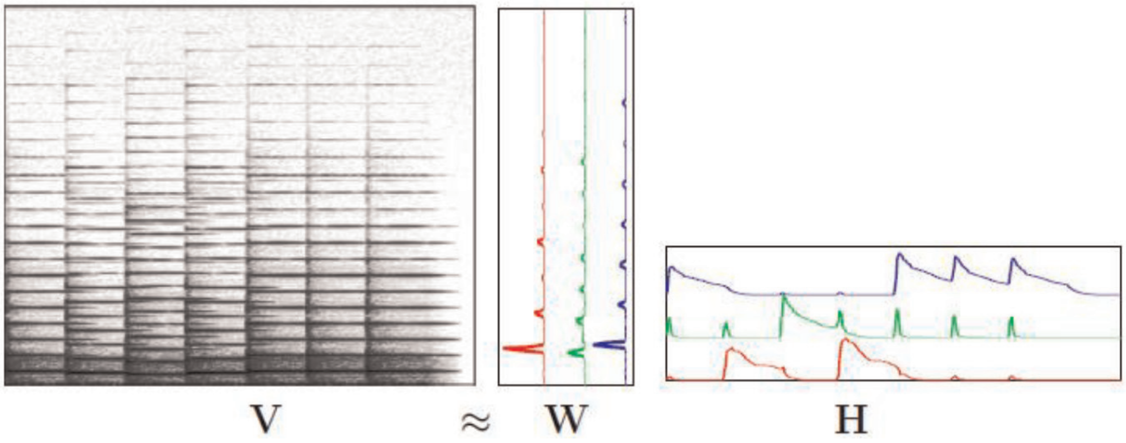


Figure 1.
NMF of a piano performing “Mary had a little lamb” for two measures with N_z = 3. Notice how matrix ***W*** *captures the harmonic content of the three pitches of the passage and matrix* ***H*** *captures the time onsets and gains of the individual notes [3].*

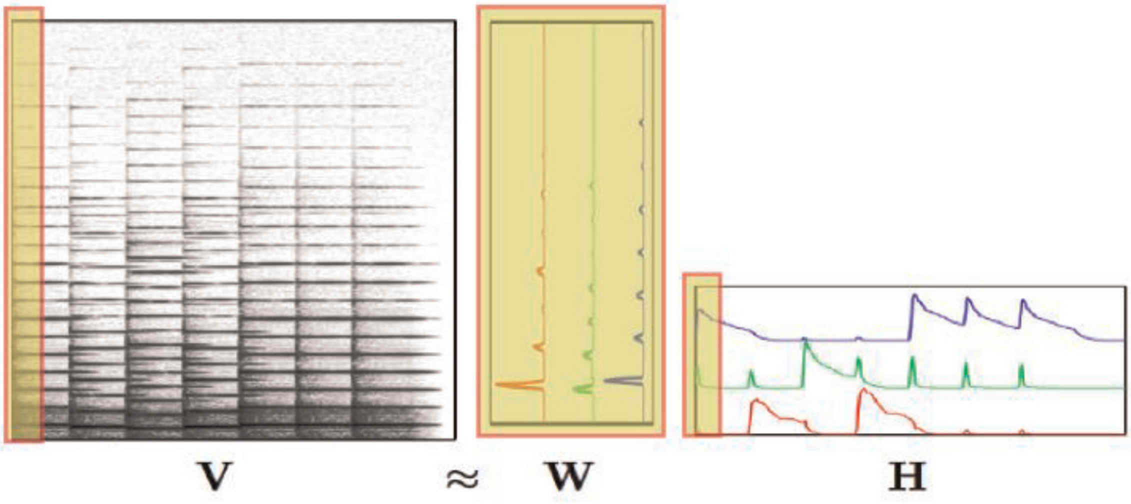


Figure 2.
NMF interpretation I. the columns of ***V*** *(i.e., short-time segments of the mixture signal) are approximated as a weighted sum or mixture of basis vectors* ***W*** *[3].*

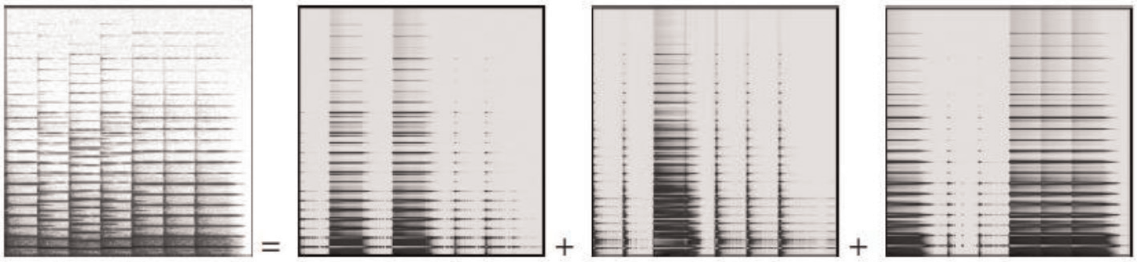


Figure 3.
*NMF interpretation II. The matrix **V** (i.e., the mixture signal) is approximated as a sum of matrix "layers" [3].*

$$V \approx \begin{bmatrix} | & | & | & | \\ V1 & V2 & V_3 \ldots\ldots & V_{N_t} \\ | & | & | & | \end{bmatrix} \approx \begin{bmatrix} | & | & | & | \\ W1 & W2 & W_3 \ldots\ldots & W_{N_z} \\ | & | & | & | \end{bmatrix} \begin{bmatrix} h_1^T \\ h_2^T \\ h_3^T \\ . \\ h_{N_z}^T \end{bmatrix}$$

$$V \approx W_1 h_1^T + W_2 h_2^T + W_3 h_3^T + \ldots\ldots + W_{N_z} h_{N_z}^T \tag{3}$$

The application of NMF on noisy speech can be seen in **Figure 4**.

2.2 Optimization formulation

To estimate the basis matrix **W** and the activation matrix **H** for a given input data matrix **V**, NMF algorithm is formulated as an optimization problem. This is written as:

$$\underset{\mathrm{W,\,H}}{argmin}\, D(V|\mathrm{WH})$$
$$W \geq 0, H \geq 0 \tag{4}$$

where $D(V|\mathrm{WH})$ is an appropriately defined cost function between **V** and **W H** and the inequalities $\geq$ are element-wise. It is also common to add additional equality constraints to require the columns of **W** to sum to one, which we enforce. When $D(V|\mathrm{WH})$ is additively separable, the cost function can be reduced to

$$D(V|\mathrm{WH}) = \sum_{f=1}^{N_f} \sum_{t=1}^{N_t} d\left(V_{ft} \middle| [\mathrm{WH}]_{ft}\right) \tag{5}$$

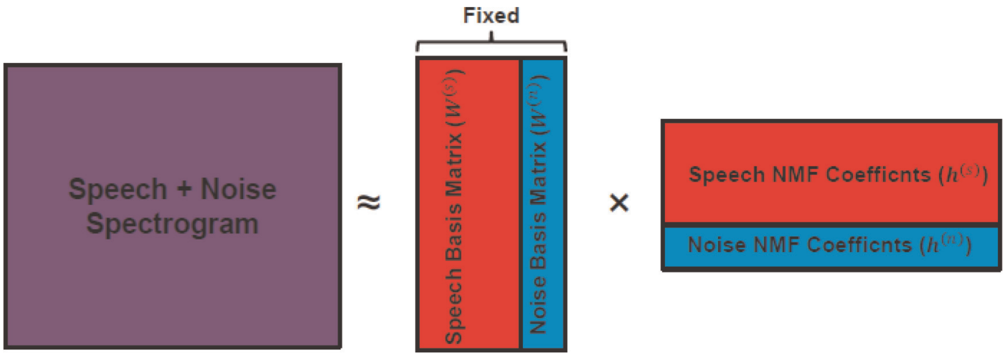

Fixed
Speech + Noise Spectrogram
≈
Speech Basis Matrix ($W^{(s)}$)
Noise Basis Matrix ($W^{(n)}$)
×
Speech NMF Coefficnts ($h^{(s)}$)
Noise NMF Coefficnts ($h^{(n)}$)

Figure 4.
Applying NMF on noisy speech.

where $[]_{ft}$ indicates its argument at row f and column t and $D(V|\mathrm{WH})$ is a scalar cost function measured between V and WH.

Popular cost functions include the Euclidean distance metric, Kullback-Liebler (KL) divergence, and Itakura-Saito (IS) divergence. Both the KL and IS divergences have been found to be well suited for audio purposes. In this work, we focus on the case where $d(q|p)$ is generalized (non-normalized) KL divergence:

$$d_{KL}(q|p) = q\, ln\frac{q}{p} - q + p \tag{6}$$

where $[]_{ft}$ indicates its argument at row f and column t and $d(q|p)$ is a scalar cost function measured between q and p.

This results in the following optimization formulation:

$$\underset{\mathrm{W,\,H}}{argmin} \sum_{f=1}^{N_f}\sum_{t=1}^{N_t} -V_{ft}\ \ln\Big|[\mathrm{WH}]_{ft} + \Big|[\mathrm{WH}]_{ft} + const$$

Subject to

$$W \geq 0, H \geq 0 \tag{7}$$

Given this formulation, we notice that the problem is not convex in **W** and **H**, limiting our ability to find a globally optimal solution to Eq. (7). It is, however, biconvex or independently convex in **W** for a fixed value of **H** and convex in **H** for a fixed value of **W**, motivating the use of iterative numerical methods to estimate locally optimal values of **W** and **H**.

2.3 Parameter estimation

To solve Eq. (7), we must use an iterative numerical optimization technique and hope to find a locally optimal solution. Gradient descent methods are the most common and straightforward for this purpose but typically are slow to converge. Other methods such as Newton's method, interior-point methods, conjugate gradient methods, and similar [4] can converge faster but are typically much more complicated to implement, motivating alternative approaches.

The most popular alternative that has been proposed is by Lee and Seung [1, 5] and consists of a fast, simple, and efficient multiplicative gradient descent-based optimization procedure. The method works by breaking down the larger optimization problem into two subproblems and iteratively optimizes over **W** and then **H**, back and forth, given an initial feasible solution. The approach monotonically decreases the optimization objective for both the KL divergence and Euclidean cost functions and converges to a local stationary point.

The approach is justified using the machinery of majorization-minimization (MM) algorithms [6]. MM algorithms are closely related to expectation maximization (EM) algorithms. In general, MM algorithms operate by approximating an optimization objective with a lower bound auxiliary function. The lower bound is then maximized instead of the original function, which is usually more difficult to optimize.

Algorithm 1 shows the complete iterative numerical optimization procedure applied to Eq. (7) with the KL divergence, where the division is element-wise, $\otimes$ is an element-wise multiplication, and 1 is a vector or matrix of ones with appropriately defined dimensions [5].

Algorithm 1 KL-NMF parameter estimation

```
Procedure KL-NMF ($V \in R_{+}^{N_f \times N_t}$//input data matrix.
$N_z$//number of basic vectors.
)
Initialize: $W \in R_{+}^{N_f \times N_z}$, $H \in R_{+}^{N_z \times N_{t.}}$
repeat
Optimize over W
    $W \leftarrow W \otimes \frac{\left(\frac{V}{\mathrm{WH}}\right)H^T}{1H^T}$    (8)
Optimize over H
    $H \leftarrow H \otimes \frac{W^T\left(\frac{V}{\mathrm{WH}}\right)}{W^T 1}$    (9)
until convergence
return: W and H
```

NMF is an optimization technique using EM algorithm in terms of matrix, whereas probabilistic latent component analysis (PLCA) is also an optimization technique using EM algorithm in terms of probability. In PLCA, we are going to incorporate probabilities of time and frequency. In the next section, the development of PLCA-based algorithm to incorporate time-frequency constraints is discussed.

3. A probabilistic latent variable model with time-frequency constraints

Considering this approach, we now develop a new PLCA-based algorithm to incorporate the time-frequency user-annotations. For clarity, we restate the form of the symmetric two-dimensional PLCA model we use:

$$p(f,t) = \sum_z p(z)p(f|z)p(t|z) \tag{10}$$

Compared to a modified NMF formulation, incorporating optimization constraints as a function of time, frequency, and sound source into the factorized PLCA model is particularly interesting and motivating to our focus.

Incorporating prior information into this model, and PLCA in general, can be done in several ways. The most commonly used methods are by direct observations (i.e., setting probabilities to zero, one, etc.) or by incorporating Bayesian prior probabilities on model parameters. Direct observations do not give us enough control, so we consider incorporating Bayesian prior probabilities. For the case of Eq. (10), this would result in independently modifying the factor terms $p(f|z)$, $p(t|z)$, or $p(z)$. Common prior probability distributions used for this purpose include Dirichlet priors [7], gamma priors [8], and others.

Given that we would like to incorporate the user-annotations as a function of time, frequency, and sound source, however, we notice that this is not easily accomplished using standard priors. This is because the model is factorized, and each factor is only a function of one variable and (possibly) conditioned by another, making it difficult to construct a set of prior probabilities that, when jointly applied to $p(f|z)$, $p(t|z)$, and/or $p(z)$, would encourage or discourage one source or another to explain a given time-frequency point. We can see this more clearly when we consider PLCA to be the following simplified estimation problem:

$$X(f,t) \approx \varphi(z)\varphi(f,z)\varphi(t,z) \tag{11}$$

where $X(f,t)$ is the observed data that we model as the product of three distinct functions or factors $\varphi(z)$, $\varphi(f,z)$, and $\varphi(t,z)$. Note, each factor has different input arguments and each factor has different parameters that we wish to estimate via EM. Also, forget for the moment that the factors must be normalized probabilities.

Given this model, if we wish to incorporate additional information, we could independently modify:

- $\varphi(z)$ to incorporate past knowledge of the variable z

- $\varphi(f,z)$ to incorporate past knowledge of the variable f and z

- $\varphi(t,z)$ to incorporate past knowledge of the variable t and z

This way of manipulation allows us to maintain our factorized form and can be thought of as prior-based regularization. If we would like to incorporate additional information/regularization that is a function of all three variables *z, f, and t,* then we must do something else. The first option would be to try to simultaneously modify all factors together to impose regularization that is a function of all three variables. This is unfortunately very difficult—both conceptually difficult to construct and practically difficult to algorithmically solve.

This motivates the use of posterior regularization (PR). PR provides us with an algorithmic mechanism via EM to incorporate constraints that are complementary to prior-based regularization. Instead of modifying the individual factors of our model as we saw before, we directly modify the posterior distribution of our model. The posterior distribution of our model, very loosely speaking, is a function of all random variables of our model. It is natively computed within each E step of EM and is required to iteratively improve the estimates of our model parameters. In this example, the posterior distribution would be akin to $\varphi(z,f,t)$, which is a function of t, f, and z, as required. We now formally discuss PR below, beginning with a general discussion and concluding with the specific form of PR we employ within our approach.

3.1 Posterior regularization

The framework of posterior regularization, first introduced by Graca, Ganchev, and Taskar [9, 10], is a relatively new mechanism for injecting rich, typically data-dependent constraints into latent variable models using the EM algorithm. In contrast to standard Bayesian prior-based regularization, which applies constraints to the model parameters of a latent variable model in the maximization step of EM, posterior regularization applies constraints to the posterior distribution (distribution over the latent variables, conditioned on everything else) computation in the expectation step of EM. The method has found success in many natural language processing tasks, such as statistical word alignment, part-of-speech tagging, and similar tasks that involve latent variable models.

In this case, what we do is constrain the distribution q in some way when we maximize the auxiliary bound $F(q,\Theta)$ with respect to q in the expectation step of an EM algorithm, resulting in

$$q^{n+1} = \underset{q}{argmin}\, KL(q\|p) + \Omega(q) \tag{12}$$

where $\Omega(q)$ constrains the possible space of q.

Note, the only difference between Eq. (12) and our past discussion on EM is the added term $\Omega(q)$. If $\Omega(q)$ is set to zero, we get back the original formulation and easily solve the optimization by setting q = p without any computation (except computing the posterior p). Also note to denote the use of constraints in this context, the term "weakly supervised" was introduced by Graca [11] and is similarly adopted here.

This method of regularization is in contrast to prior-based regularization, where the modified maximization step would be

$$\Theta^{n+1} = \underset{\Theta}{argmax}\, F\left(q^{n+1}, \Theta\right) + \Omega(\Theta) \tag{13}$$

where $\Omega(\Theta)$ constrains the model parameter Θ.

3.2 Linear grouping expectation constraints

Given the general framework of posterior regularization, we need to define a meaningful penalty $\Omega(q)$ for which we map our annotations. We do this by mapping the annotation matrices to linear grouping constraints on the latent variable z. To do so, we first notice that Eq. (12) decouples for each time-frequency point for our specific model. Because of this, we can independently solve Eq. (12) for each time-frequency point, making the optimization much simpler. When we rewrite our E step optimization using vector notation, we get

$$\underset{q}{argmin} -q_{ft}^{T} lnp_{ft} + q_{ft}^{T} lnq_{ft}$$

subject to

$$q_{ft}^{T}1 = 1, q_{ft} \geq 0 \tag{14}$$

where q and $p(z|f,t)$ for a given value of f and t is written as q_{ft} and p_{ft} without any modification; we note q is optimal when equal to $p(z|f,t)$ as before.

We then apply our linear grouping constraints independently for each time-frequency point:

$$\underset{q}{argmin} -q_{ft}^{T} lnp_{ft} + q_{ft}^{T} lnq_{ft} + q_{ft}^{T}\lambda_{ft}$$

Subject to

$$q_{ft}^{T}1 = 1, q_{ft} \geq 0, \tag{15}$$

where we define $\lambda_{ft} = \left[\Lambda_{ft1}........\Lambda_{ft1}\Lambda_{ft2}.........\Lambda_{ft2}\right]^{T} \in R^{N_z}$ as the vector of user-defined penalty weights, T is a matrix transpose, $\geq$ is element-wise greater than or equal to, and **1** is a column vector of ones. In this case, positive-valued penalties are used to decrease the probability of a given source, while negative-valued coefficients are used to increase the probability of a given source. Note the penalty weights imposed on the group of values of z that correspond to a given source s are identical, linear with respect to the z variables, and applied in the E step of EM, hence the name "linear grouping expectation constraints."

To solve the above optimization problem for a given time-frequency point, we form the Lagrangian

$$L\left(q_{ft}, \gamma\right) = -q_{ft}^T \ln p_{ft} + q_{ft}^T \ln q_{ft} + q_{ft}^T \lambda_{ft} + \gamma\left(1 - q_{ft}^T 1\right) \tag{16}$$

With γ being a Lagrange multiplier, take the gradient with respect to q and γ:

$$\nabla_{q_{ft}} L\left(q_{ft}, \gamma\right) = -\ln p_{ft} + 1 + \ln q_{ft} + \lambda_{ft} - \gamma 1 = 0 \tag{17}$$

$$\nabla_a L\left(q_{ft}, \gamma\right) = \left(1 - q_{ft}^T 1\right) = 0 \tag{18}$$

set Eqs. (17) and (18) equal to zero, and solve for q_{ft}, resulting in

$$q_{ft} = \frac{P_{ft} \otimes exp\left(-\lambda_{ft}\right)}{P_{ft}^T exp\left(-\lambda_{ft}\right)} \tag{19}$$

where exp{} is an element-wise exponential function.

Notice the result is computed in closed form and does not require any iterative optimization scheme as may be required in the general posterior regularization framework [9], minimizing the computational cost when incorporating the constraints. Also note, however, that this optimization must be solved for each time-frequency point of our spectrogram data for each E step iteration of our final EM parameter estimation algorithm.

3.3 Parameter estimation

Now knowing the posterior-regularized expectation step optimization, we can derive a complete EM algorithm for a posterior-regularized two-dimensional PLCA model (PR-PLCA):

$$p(z|f, t) \leftarrow \frac{p(z)p(f|z)p(t|z)\overline{\Lambda}_{ftz}}{\sum_{z'} p(z')p(f|z')p(t|z')\overline{\Lambda}_{ftz'}} \tag{20}$$

where $\overline{\Lambda} = \exp\{-\Lambda\}$. The entire algorithm is outlined in Algorithm 2. Notice we continue to maintain closed-form E and M steps, allowing us to optimize further and draw connections to multiplicative nonnegative matrix factorization algorithms.

Algorithm 2 PR-PLCA with linear grouping expectation constraints

```
Procedure PLCA (
$V \in R_+^{N_f \times N_t}$//observed normalized data
$N_z$//number of basis vectors
$N_s$//number of sources
$\Lambda \in R^{N_f \times N_t \times N_z}$//penalties
)
Initialize: feasible $p(z)$,$p(f|z)$ and $p(t|z)$

Precompute:        $\overline{\Lambda} \leftarrow \exp(-\Lambda)$        (21)
```

```
repeat
    Expectation step
    for all $z, f, t$ do
        $p(z|f,t) \leftarrow \frac{p(z)p(f|z)p(t|z)\overline{\Lambda}_{ftz}}{\sum_{z'}p(z')p(f|z')p(t|z')\overline{\Lambda}_{ftz'}}$    (22)
    end for
        Maximization step
        for all $z, f, t$ do
            $p(f|z) = \frac{\sum_t V_{ft}p(z|f,t)}{\sum_{f'}\sum_{t'}V_{f't'}p(z|f',t')}$    (23)
            $p(t|z) = \frac{\sum_f V_{ft}p(z|f,t)}{\sum_{f'}\sum_{t'}V_{f't'}p(z|f',t')}$    (24)
            $p(z) = \frac{\sum_f\sum_t V_{ft}p(z|f,t)}{\sum_{z'}\sum_{f'}\sum_{t'}V_{f't'}p(z|f',t')}$    (25)
        end for
until convergence
return:$p(f|z)$, $p(t|z)$,$p(z)$ and $p(z|f,t)$
```

- ***Multiplicative Update Equations***

We can rearrange the expressions in Algorithm 2 and convert to a multiplicative form following similar methodology to Smaragdis and Raj [12].

Rearranging the expectation and maximization steps, in conjunction with Bayes' rule, and

$$Z(f,t) = \sum_z p(z)p(f|z)p(t|z)\overline{\Lambda}_{ftz},$$

we get

$$p(z|f,t) = \frac{p(f|z)p(t,z)\overline{\Lambda}_{ftz}}{Z(f,t)} \tag{26}$$

$$p(t,z) = \sum_f V_{ft}q(z|f,t) \tag{27}$$

$$p(f|z) = \frac{\sum_t V_{ft}q(z|f,t)}{\sum_t p(t,z)} \tag{28}$$

$$p(z) = \sum_t p(t,z) \tag{29}$$

Rearranging further, we get

$$p(f|z) = \frac{p(f|z)\sum_t \frac{V_{ft}\overline{\Lambda}_{ftz}}{Z(f,t)}p(t,z)}{\sum_t p(t,z)} \tag{30}$$

$$p(t,z) = p(t,z)\sum_f p(f|z)\frac{V_{ft}\overline{\Lambda}_{ftz}}{Z(f,t)} \tag{31}$$

which fully specifies the iterative updates. By putting Eqs. (30) and (31) in matrix notation, we specify the multiplicative form of the proposed method in Algorithm 3.

Algorithm 3. PR-PLCA with linear grouping expectation constraints in matrix notation

```
Procedure PLCA (
$V \in R_{+}^{N_f \times N_t}$//observed normalized data
$N_z$//number of basis vectors
$N_s$//number of sources
$\Lambda_s \in R^{N_f \times N_t}, \forall \in \{1, ....N_s\}$//penalties
)
Initialize: $W \in R_{+}^{N_f \times N_z}, H \in R_{+}^{N_z \times N_t}$
Precompute:
        For all s do
                $\overline{\Lambda}_s \leftarrow \exp\{-\Lambda_s\}$    (32)
                $X_s \leftarrow V \otimes \overline{\Lambda}_s$    (33)
        End for
Repeat
                $\Gamma \leftarrow \sum_s (W_s H_s) \otimes \overline{\Lambda}_s$    (34)
        For all s do
                $Z_s \leftarrow \frac{X_s}{\Gamma}$    (35)
                $W_{(s)} \leftarrow W_s \otimes \frac{Z_s H_s^T}{1 H_s^T}$    (36)
                $H_{(s)} \leftarrow H_s \otimes \left(W_s^T Z_s\right)$    (37)
        End for
until convergence
return: W and H
```

4. An iterative posterior NMF method for speech enhancement in the presence of additive Gaussian noise (proposed algorithm)

Over the past several years, research has been carried out in single-channel sound source separation methods. This problem is motivated by speech denoising, speech enhancement [13], music transcription [14], audio-based forensic, and music remixing. One of the most effective approach is nonnegative matrix factorization (NMF) [5]. The user-annotations can be used to obtain the PR terms [15]. If the number of sources is more, then it is difficult to identify sources in the spectrogram. In such cases, the user interaction-based constraint approaches are inefficient.

In order to avoid the previous problem, in the proposed method, an automatic iterative procedure is introduced. The spectral components of speech and noise are modeled as Gamma and Rayleigh, respectively [16].

4.1 Notation and basic concepts

Let noisy speech signal x[n] be the sum of clean speech s[n] and noise d[n] and their corresponding magnitude spectrogram be represented as

$$|\boldsymbol{X}(\boldsymbol{f},\boldsymbol{t})| = |\boldsymbol{S}(\boldsymbol{f},\boldsymbol{t})| + |\boldsymbol{D}(\boldsymbol{f},\boldsymbol{t})| \tag{38}$$

where f represents the frequency bin and t the frame number. The observed magnitudes in time-frequency are arranged in a matrix $\mathbf{X} \in R_+^{f\times t}$ of nonnegative elements. The source separation algorithms based on NMF pursue the factorization of $\mathbf{X}$ as a product of two nonnegative matrices, $\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_R] \in R_+^{f\times R}$ in which the columns collect the basis vectors and $\boldsymbol{H} = \left[h_1^T, h_2^T, \dots., h_R^T\right]^T \in R_+^{R\times t}$ that collects their respective weights, i.e.,

$$X = \mathrm{WH} = \sum_{z=1}^{R} W_z H_z \tag{39}$$

where R denotes the number of latent components.

4.2 Proposed regularization

There are several ways to incorporate the user-annotations into latent variable models, for instance, by using the suitable regularization functions. For expectation maximization (EM) algorithms, posterior regularization was introduced by [9, 11]. This method is data dependent. This method gives richness and also gives the constraints on the posterior distributions of latent variable models. The applications of this method is used in many natural language processing tasks like statistical word alignment, part-of-speech tagging. The main idea is to constrain on the distribution of posterior, when computing expectation step in EM algorithm.

The prior distributions for the magnitude of the noise spectral components are modeled as Rayleigh probability density function (PDF) with scale parameter σ, which is fitted to the observations by a maximum likelihood procedure [16, 17], i.e.,

$$f(x;\sigma) = \frac{x}{\sigma^2} e^{-x^2/2\sigma^2}\, \mathit{for}\ x \geq 0\ \mathit{with}\ \sigma^2 = \frac{1}{2N}\sum_{i=1}^{N} x_i^2 \tag{40}$$

The above equation can be written as

$$f(x;\sigma) = e^{\log\left(\frac{x}{\sigma^2}\right)} e^{-x^2/2\sigma^2} = e^{\log\left(\frac{x}{\sigma^2}\right) - \frac{x^2}{2\sigma^2}} \tag{41}$$

By applying negative logarithm on both sides of (41), we will get

$$-\log\left(f(x;\sigma)\right) = -\log\left(e^{\log\left(\frac{x}{\sigma^2}\right) - \frac{x^2}{2\sigma^2}}\right) = \frac{x^2}{2\sigma^2} - \log\left(\frac{x}{\sigma^2}\right) \tag{42}$$

Then, the regularization term for the noise is defined as

$$\Lambda_N \equiv \Lambda_{S_1} = -\log f(x;\sigma) = \frac{x^2}{2\sigma^2} - \log\frac{x}{\sigma^2}. \tag{43}$$

The spectral components of speech modeled as Gamma probability density function [16, 18]

$$f(x;k,\theta)=\frac{x^{k-1}e^{\frac{-x}{\theta}}}{\theta^{k}\Gamma(k)} \tag{44}$$

with shape parameter $k>0$ and scale parameter $\theta>0$:

$$\theta=\frac{1}{kN}\sum_{i=1}^{N}x_i \textit{ and } k\approx\frac{3-s+\sqrt{(s-3)^2+24s}}{12s} \tag{45}$$

where the auxiliary variable s is defined as $s=\ln\left(\frac{1}{N}\sum_{i=1}^{N}x_i\right)-\frac{1}{N}\sum_{i=1}^{N}\ln(x_i)$.

The regularization term for the speech samples is defined as (by applying negative logarithm in both sides of (44))

$$\Lambda_{SP}\equiv\Lambda_{S_2}=-\log f(x;k,\theta)=\frac{x}{\theta}-\log\left(\frac{x^{k-1}}{\theta^{k}\Gamma(k)}\right), x\geq 0 \tag{46}$$

Special case: When we fix $k = 1$, the Gamma density simplifies to the exponential density and

$$f(x;1,\theta)=\frac{1}{\theta}e^{\frac{-x}{\theta}}, \Lambda_{SP}\equiv\Lambda_{S_2}=\frac{x}{\theta}, x\geq 0 \tag{47}$$

The proposed multiplicative nonnegative matrix factorization method is summarized in Algorithm 4 [16]. In general, like in the specific case of Algorithm 4, one can only guarantee the monotonous descent of the iteration through a majorization-minimization approach [19] or the convergence to a stationary point [20].

The subscript(s) with parenthesis represents corresponding columns or rows of the matrix assigned to a given source. **1** is an approximately sized matrices of ones, and ⊗ represents element-wise multiplication.

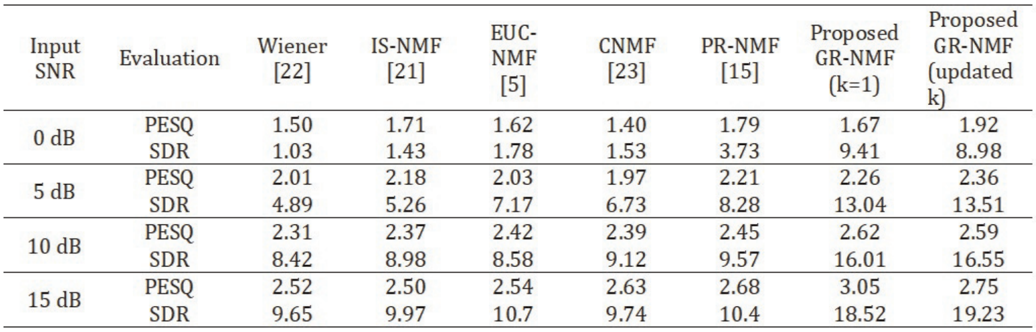
Algorithm 4: Gamma-Rayleigh regularized PLCA method (GR-NMF)

```
Procedure
        $X\in R_{+}^{f\times t}$ % Observed normalized data
        $\Lambda_S\in R_{+}^{f\times t}, s\in\{1,\ldots..,N_S\}$ % $\Lambda_S$-Penalties, $N_S$-Number of sources
        $\Lambda_{s(NEW)}=0$
        $\Lambda_{S_1}=\Lambda_{N(OLD)}=\frac{X^2}{2\sigma^2}-\log\frac{X}{\sigma^2}$ and $\Lambda_{S_2}=\Lambda_{SP(OLD)}=\frac{X}{\theta}-\log\left(\frac{X^{k-1}}{\theta^{k}\Gamma(k)}\right)$    (48)
        $\widetilde{\Lambda}_{s(OLD)}\leftarrow\exp\{-\Lambda_s\}$    (49)
Repeat
    For all s do
        $\widetilde{\Lambda}_s=(1-\mu)\Lambda_{s(OLD)}+\mu\Lambda_{s(NEW)}$ %Update penalties using LMS    (50)
        $\Lambda_{s(OLD)}=\Lambda_{s(NEW)}$    (51)
```

```
            $X_s \leftarrow X \otimes \widetilde{\Lambda}_S$    (52)
        End for
        $\Gamma \leftarrow \sum_s \left(W_{(s)}H_{(s)}\right) \otimes \widetilde{\Lambda}_S$    (53)
        $Z_s \leftarrow \frac{X_s}{\Gamma}$    (54)
        For all s do
            $W_{(s)} \leftarrow W_{(s)} \otimes \frac{Z_s H_{(s)}^T}{1\mathrm{H}_{(s)}^T}$    (55)
            $H_{(s)} \leftarrow H_{(s)} \otimes \left(W_{(s)}^T Z_s\right)$    (56)
        End for
    Reconstruction
        For all s do
            $M_s \leftarrow \frac{W_{(s)}H_{(s)}}{\mathrm{WH}}$ % Compute Filter    (57)
            $\hat{X}_s \leftarrow M_s \otimes X$ % Filter Mixture    (58)
            $x_s \leftarrow ISTFT\left(\hat{X}_s, \angle \mathrm{X}, P\right)$ % P - STFT parameters    (59)
                if update k % Gamma model
                    $s = \ln\left(\frac{1}{N}\sum \hat{X}_s\right) - \frac{1}{N}\sum \ln\left(\hat{X}_s\right), k \approx \frac{3 - s + \sqrt{(s-3)^2 + 24s}}{12s}$    (60)
    else % Exponential model
            k = 1,
    end
            $\theta = \frac{1}{kN}\sum \hat{X}_s$    (61)
            $\Lambda_{S_1} = \Lambda_{N(OLD)} = \frac{\hat{X}_{s_1}^2}{2\sigma^2} - \log \frac{\hat{X}_{s_1}}{\sigma^2}$
            $\Lambda_{S_2} = \Lambda_{SP(OLD)} = \frac{\hat{X}_{s_2}}{\theta} - \log\left(\frac{\hat{X}_{s_2}^{\mathrm{k}-1}}{\theta^k \Gamma(k)}\right)$    (62)
    $\Lambda_{s(NEW)} = \exp\left(-\Lambda_{s(OLD)}\right)$ % $\Lambda_{s(OLD)}$ represents both $\Lambda_{SP}$ and $\Lambda_N$    (63)
        End for
    Until Convergence
    Return: Time domain signals $x_s$
```

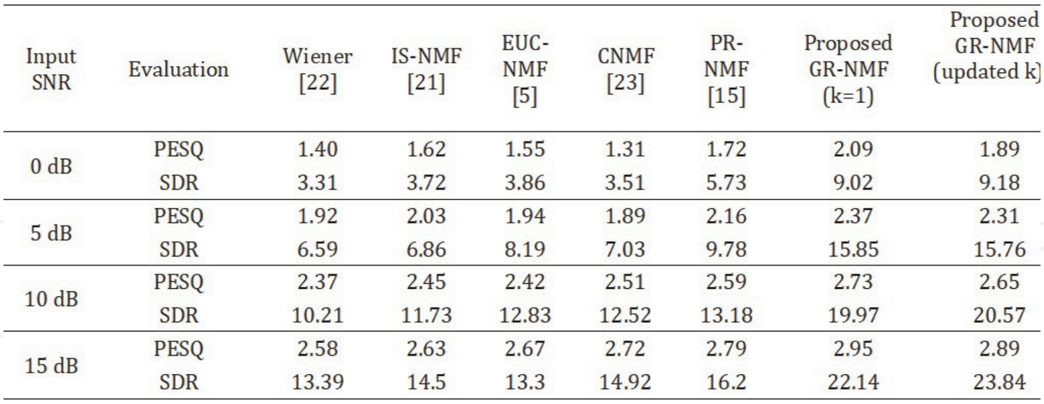
5. Experimental results

The speech and noise audio samples were taken from NOIZEUS [21]. Sampling frequency is 8 KHz. The algorithm is iterated until convergence [16]. The proposed method was compared with Euclidean NMF (EUC-NMF) [5], Itakura-Saito NMF (IS-NMF) [22], posterior regularization NMF (PR-NMF) [15], Wiener filtering [23], and constrained version of NMF (CNMF)[24]. These methods are implemented by considering nonstationary noise, babble noise and street noise. The performance of proposed method was evaluated by using perceptual evaluation of speech quality (PESQ) [25] and source-to-distortion ratio (SDR) [26]. SDR gives the average quality of separation on dB scale and considers signal distortion as well as noise distortion. For PESQ and SDR, the higher value indicates the better performance. **Tables 1** and **2** show the PESQ and SDR values of different NMF algorithms evaluated. The experimental results show that proposed method performs better than other existing methods in terms of the PESQ and SDR indices.



Input SNR	Evaluation	Wiener [22]	IS-NMF [21]	EUC-NMF [5]	CNMF [23]	PR-NMF [15]	Proposed GR-NMF (k=1)	Proposed GR-NMF (updated k)
0 dB	PESQ	1.50	1.71	1.62	1.40	1.79	1.67	1.92
	SDR	1.03	1.43	1.78	1.53	3.73	9.41	8..98
5 dB	PESQ	2.01	2.18	2.03	1.97	2.21	2.26	2.36
	SDR	4.89	5.26	7.17	6.73	8.28	13.04	13.51
10 dB	PESQ	2.31	2.37	2.42	2.39	2.45	2.62	2.59
	SDR	8.42	8.98	8.58	9.12	9.57	16.01	16.55
15 dB	PESQ	2.52	2.50	2.54	2.63	2.68	3.05	2.75
	SDR	9.65	9.97	10.7	9.74	10.4	18.52	19.23

Table 1.
PESQ and SDR for babble noise.



Input SNR	Evaluation	Wiener [22]	IS-NMF [21]	EUC-NMF [5]	CNMF [23]	PR-NMF [15]	Proposed GR-NMF (k=1)	Proposed GR-NMF (updated k)
0 dB	PESQ	1.40	1.62	1.55	1.31	1.72	2.09	1.89
	SDR	3.31	3.72	3.86	3.51	5.73	9.02	9.18
5 dB	PESQ	1.92	2.03	1.94	1.89	2.16	2.37	2.31
	SDR	6.59	6.86	8.19	7.03	9.78	15.85	15.76
10 dB	PESQ	2.37	2.45	2.42	2.51	2.59	2.73	2.65
	SDR	10.21	11.73	12.83	12.52	13.18	19.97	20.57
15 dB	PESQ	2.58	2.63	2.67	2.72	2.79	2.95	2.89
	SDR	13.39	14.5	13.3	14.92	16.2	22.14	23.84

Table 2.
PESQ and SDR for street noise.

6. Conclusion

A novel speech enhancement method based on an iterative and regularized NMF algorithm for single-channel source separation is proposed. The clean speech and noise magnitude spectra are modeled as Gamma and Rayleigh distributions, respectively. The corresponding log-likelihood functions are used as penalties to regularize the cost function of the NMF. The estimation of basis matrices and excitation matrices are calculated by using proposed regularization of multiplicative update rules. The experiments reveal that the proposed speech enhancement method outperforms other existing benchmark methods in terms of SDR and PESQ values.

Author details

Sunnydayal Vanambathina
Department of Electronics and Communication Engineering, VIT-AP University, Amaravati, India

*Address all correspondence to: sunny.conference@gmail.com

References

[1] Lee DD, Seung HS. Learning the parts of objects by non-negative matrix factorization. Nature. 1999;**401**:788-791

[2] Smaragdis P. Non-negative matrix factorization for polyphonic music transcription. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics; 19–22 October 2003. Mohonk Mountain; 2013. pp. 177-180

[3] Bryan NJ, Mysore GJ. An efficient posterior regularized latent variable model for interactive sound source separation. In: International Conference on Machine Learning (ICML); June 2013

[4] Boyd S, Vandenberghe L. Convex Optimization. New York, NY, USA: Cambridge University Press; 2004

[5] Lee DD, Seung HS. Algorithms for Non-negative Matrix Factorization. NIPS Proceedings. 2001

[6] Hunter DR, Lange K. A tutorial on MM algorithms. The American Statistician. 2004;**58**:30-37

[7] Paltz N. Separation by 'Humming': user-guided sound extraction from monophonic mixtures. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA); 2009. pp. 69–72

[8] Fitzgerald D. User assisted separation using tensor factorisations. In: European Signal Processing Conference (EUSIPCO). 2012. pp. 2412–2416

[9] Graca J, Ganchev K, Taskar B. Expectation maximization and posterior constraints. Advances in Neural Information Processing Systems. 2008; **20**:1-8

[10] Ganchev K, Gillenwater J. Posterior regularization for structured latent variable models. Journal of Machine Learning Research. 2010;**11**:2001-2049

[11] Graça J, Ganchev K, Taskar B, Pereira F. Posterior vs. parameter sparsity in latent variable models. NIPS–Advances in Neural Information Processing Systems. 2009:664-672

[12] Smaragdis P, Raj B. Shift-invariant probabilistic latent component analysis. Journal of Machine Learning Research. Technical Report TR2007009, MERL; December, 2007:5

[13] Mysore GJ, Smaragdis P. A non-negative approach to semi-supervised separation of speech from noise with the use of temporal dynamics Gautham J. Mysore Advanced Technology Labs Adobe Systems Inc, University of Illinois at Urbana-Champaign, Adobe Systems Inc. IEEE International Conference on Acoustics, Speech and Signal Processing–ICASSP 2011; 2011. pp. 17–20

[14] Bertin N, Badeau R, Vincent E. Enforcing harmonicity and smoothness in Bayesian non-negative matrix factorization applied to polyphonic music transcription. IEEE Transactions on Audio, Speech and Language Processing. 2010;**18**:538-549

[15] Bryan NJ, Mysore GJ. An Efficient Posterior Regularized Latent Variable Model for Interactive Sound Source Separation. in Icml, 2013

[16] Sunnydayal K k, Cruces-Alvarez SA. An iterative posterior NMF method for speech enhancement in the presence of additive Gaussian noise. Neurocomputing. 2017;**230**:312-315

[17] Cruces-Alvarez SA, Cichocki A, ichi Amari S. From blind signal extraction to blind instantaneous signal separation: Criteria, algorithms, and stability. IEEE Transactions on Neural Networks. 2004;**15**:859-873

[18] Erkelens JS, Hendriks RC, Heusdens R, Jensen J. Minimum mean-square

error estimation of discrete Fourier coefficients with generalized gamma priors. IEEE Transactions on Audio, Speech and Language Processing. 2007; **15**(6):1741-1752

[19] Cichocki A, Cruces S, ichi Amari S. Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization. Entropy;**13**: 134-170

[20] Lin C-J. On the convergence of multiplicative update for nonnegative matrix factorization. IEEE Transactions on Neural Networks. 2007;**18**:1589-1596

[21] https://ecs.utdallas.edu/loizou/speech/noizeus/ [Online]

[22] Févotte C, Bertin N, Durrieu J-L. Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. Neural Computation. 2009;**21**:793-830

[23] Ephraim Y, Malah D. Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator. IEEE Transactions on Acoustics. 1984;**32**:1109-1121

[24] Berry MW, Browne M, Langville AN, Pauca VP, Plemmons RJ. Algorithms and applications for approximate nonnegative matrix factorization. Computational Statistics and Data Analysis. 2007;**52**(1):155-173

[25] Hu Y, Loizou PC. Evaluation of objective quality measures for speech enhancement. IEEE Transactions on Acoustics, Speech, and Signal Processing. 2008;**16**(1):229-238

[26] Vincent E, Gribonval R, Fevotte C. Performance measurement in blind audio source separation. IEEE Transactions on Audio, Speech and Language Processing. 2006;**14**: 1462-1469