A Variance Modeling Framework Based on Variational Autoencoders for Speech Enhancement

Simon Leglaive[1] Laurent Girin[1,2] Radu Horaud[1]

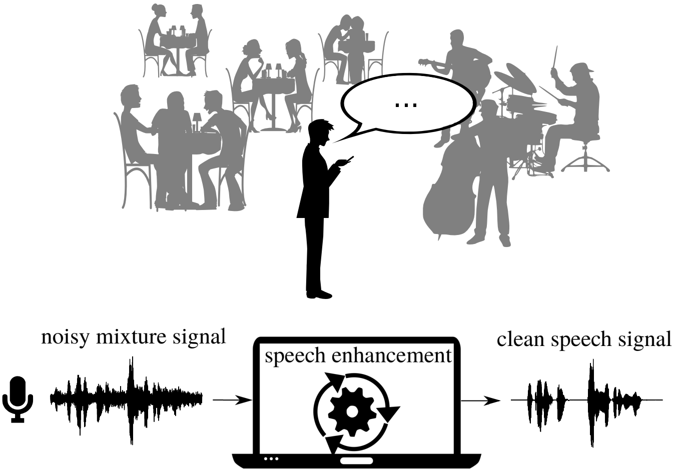
1: Inria Grenoble Rhône-Alpes 2: Univ. Grenoble Alpes, Grenoble INP, GIPSA-lab

IEEE International Workshop on Machine Learning for Signal Processing (MLSP)

September 18, 2018 – Aalborg, Denmark

Introduction

Speech enhancement



- ▷ Preprocessing step for various speech information retrieval tasks (e.g. automatic speech recognition, voice activity detection, etc.).

Speech enhancement as a source separation problem

In the short-term Fourier transform (STFT) domain, for all $(f, n) \in \mathbb{B} = \{0, ..., F-1\} \times \{0, ..., N-1\}$, we observe:

$$x_{fn} = s_{fn} + b_{fn}, \tag{1}$$

- ▷ $s_{fn} \in \mathbb{C}$ is the clean speech signal.
- ▷ $b_{fn} \in \mathbb{C}$ is the noise signal.
- ▷ f is the frequency index and n the time-frame index.

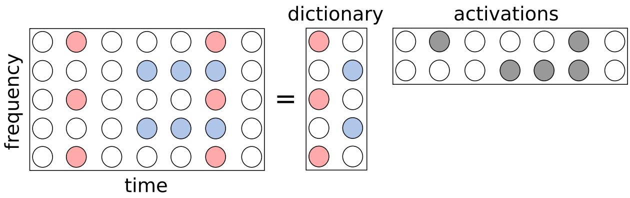
Objective: Separate the speech and noise signals from the observed mixture signal (under-determined problem).

Variance modeling with non-negative matrix factorization (NMF)

From [1], independently for all $(f, n) \in \mathbb{B}$:

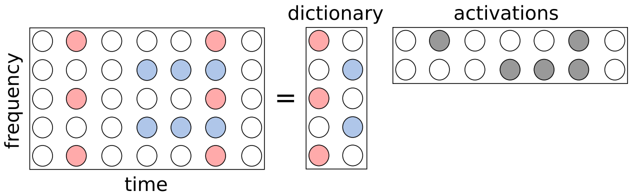
$$s_{fn} \sim \mathcal{N}_c\Big(0, (\mathbf{W}_s \mathbf{H}_s)_{f,n}\Big), \tag{2}$$

- $\mathbf{W}_s \in \mathbb{R}_+^{F \times K_s}$ is a dictionary matrix of spectral templates.
- $\mathbf{H}_s \in \mathbb{R}_+^{K_s \times N}$ is the activation matrix.
- K_s is the rank of the factorization (usually $K_s(F + N) \ll FN$).



[1] C. Févotte, N. Bertin and J. L. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis", *Neural computation*, 2009.

Supervised NMF



- Supervised setting:
  - $\mathbf{W}_s$ is learned on a dataset of clean speech signals.
  - $\mathbf{H}_s$ is estimated from the noisy mixture signal.

- Pros and cons:
  - Easy to interpret.
  - Linear variance model $\qquad \mathbb{E}[|s_{fn}|^2] = (\mathbf{W}_s\mathbf{H}_s)_{f,n} = \mathbf{w}_{s,f}^\top \mathbf{h}_{s,n}.$
  - Limited number of trainable parameters.

In this work...

... we explore the use of neural networks as an alternative to this supervised NMF-based variance model.

Model

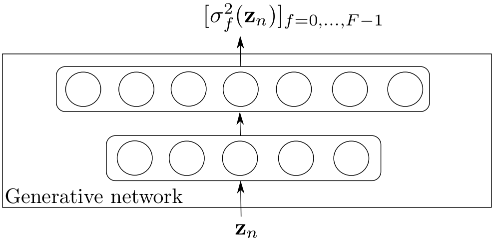
Speech variance modeling with neural networks

From [2, 3], independently for all $(f, n) \in \mathbb{B}$:

$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_L); \tag{3}$$

$$s_{fn} \mid \mathbf{z}_n \sim \mathcal{N}_c(0, \sigma_f^2(\mathbf{z}_n)), \tag{4}$$

- $\mathbf{z}_n \in \mathbb{R}^L$ is a latent random vector with $L \ll F$.
- $\sigma_f^2 : \mathbb{R}^L \mapsto \mathbb{R}_+$ is a non-linear function parametrized by $\boldsymbol{\theta}_s$.



[2] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes", *Proc. of ICLR*, 2014.
[3] Y. Bando *et al.*, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization", *Proc. of IEEE ICASSP*, 2018.

Noise and mixture models

▷ **Unsupervised noise model**: Independently for all $(f, n) \in \mathbb{B}$,

$$b_{fn} \sim \mathcal{N}_c\left(0, (\mathbf{W}_b\mathbf{H}_b)_{f,n}\right), \tag{5}$$

where $\mathbf{W}_b \in \mathbb{R}_+^{F\times K_b}$ and $\mathbf{H}_b \in \mathbb{R}_+^{K_b\times N}$.

Noise and mixture models

▷ **Unsupervised noise model**: Independently for all $(f, n) \in \mathbb{B}$,

$$b_{fn} \sim \mathcal{N}_c \left(0, (\mathbf{W}_b \mathbf{H}_b)_{f,n}\right), \tag{5}$$

where $\mathbf{W}_b \in \mathbb{R}_+^{F \times K_b}$ and $\mathbf{H}_b \in \mathbb{R}_+^{K_b \times N}$.

▷ **Mixture model**: For all $(f, n) \in \mathbb{B}$,

$$x_{fn} = \sqrt{g_n} s_{fn} + b_{fn}, \tag{6}$$

where $g_n \in \mathbb{R}_+$ is a gain parameter.

Noise and mixture models

$\triangleright$ **Unsupervised noise model**: Independently for all $(f, n) \in \mathbb{B}$,

$$b_{fn} \sim \mathcal{N}_c \left(0, (\mathbf{W}_b \mathbf{H}_b)_{f,n}\right), \tag{5}$$

where $\mathbf{W}_b \in \mathbb{R}_+^{F \times K_b}$ and $\mathbf{H}_b \in \mathbb{R}_+^{K_b \times N}$.

$\triangleright$ **Mixture model**: For all $(f, n) \in \mathbb{B}$,

$$x_{fn} = \sqrt{g_n} s_{fn} + b_{fn}, \tag{6}$$

where $g_n \in \mathbb{R}_+$ is a gain parameter.

$\triangleright$ **Conditional mixture distribution**:

$$x_{fn} \mid \mathbf{z}_n \sim \mathcal{N}_c \left(0, g_n \sigma_f^2(\mathbf{z}_n) + (\mathbf{W}_b \mathbf{H}_b)_{f,n}\right). \tag{7}$$

Inference

Parameters estimation

- For now, we assume that the speech parameters $\boldsymbol{\theta}_s$ have been learned during a training phase.

- Unsupervised model parameters:

$$\boldsymbol{\theta}_u = \left\{ \mathbf{W}_b \in \mathbb{R}_+^{F \times K_b}, \mathbf{H}_b \in \mathbb{R}_+^{K_b \times N}, \mathbf{g} = [g_0, ..., g_{N-1}]^\top \in \mathbb{R}_+^N \right\}$$

- Observed data: $\mathbf{x} = \{x_{fn} \in \mathbb{C}\}_{(f,n) \in \mathbb{B}}$

<u>Direct maximum likelihood estimation is intractable</u>

Parameters estimation

- For now, we assume that the speech parameters $\boldsymbol{\theta}_s$ have been learned during a training phase.

- Unsupervised model parameters:

$$\boldsymbol{\theta}_u = \left\{ \mathbf{W}_b \in \mathbb{R}_+^{F \times K_b}, \mathbf{H}_b \in \mathbb{R}_+^{K_b \times N}, \mathbf{g} = [g_0, ..., g_{N-1}]^\top \in \mathbb{R}_+^N \right\}$$

- Observed data: $\mathbf{x} = \{x_{fn} \in \mathbb{C}\}_{(f,n) \in \mathbb{B}}$

<u>Direct maximum likelihood estimation is intractable</u>

- Latent data: $\mathbf{z} = \{\mathbf{z}_n \in \mathbb{R}^L\}_{n=0}^{N-1}$

- Expectation-maximization (EM) algorithm.

Monte Carlo EM algorithm

- **E-Step.** From the current value of the parameters $\boldsymbol{\theta}_u^\star$, compute:

$$Q(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star) = \mathbb{E}_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}\left[\ln p(\mathbf{x},\mathbf{z};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)\right]$$

Monte Carlo EM algorithm

- **E-Step.** From the current value of the parameters $\boldsymbol{\theta}_u^\star$, compute:

$$\begin{aligned} Q(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star) &= \mathbb{E}_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}\left[\ln p(\mathbf{x},\mathbf{z};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)\right] \\ &\approx \frac{1}{R}\sum_{r=1}^{R} \ln p\left(\mathbf{x},\mathbf{z}^{(r)};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u\right), \end{aligned} \tag{8}$$

where the samples $\left\{\mathbf{z}^{(r)}\right\}_{r=1,\ldots,R}$ are asymptotically drawn from $p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)$ using a Markov chain Monte Carlo method.

Monte Carlo EM algorithm

- **E-Step.** From the current value of the parameters $\boldsymbol{\theta}_u^\star$, compute:

$$
\begin{aligned}
Q(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star) &= \mathbb{E}_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)} [\ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u)] \\
&\approx \frac{1}{R} \sum_{r=1}^{R} \ln p \left(\mathbf{x}, \mathbf{z}^{(r)}; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u \right), \qquad (8)
\end{aligned}
$$

where the samples $\left\{\mathbf{z}^{(r)}\right\}_{r=1,\dots,R}$ are asymptotically drawn from $p(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u^\star)$ using a Markov chain Monte Carlo method.

- **M-Step.**

$$
\boldsymbol{\theta}_u^\star \leftarrow \arg\max_{\boldsymbol{\theta}_u} \; Q(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star), \qquad (9)
$$

with $\boldsymbol{\theta}_u = \{\mathbf{H}_b \in \mathbb{R}_+^{K_b \times N}, \mathbf{W}_b \in \mathbb{R}_+^{F \times K_b}, \mathbf{g} \in \mathbb{R}_+^N\}$.

Speech estimation

Let $\tilde{s}_{fn} = \sqrt{g_n} s_{fn}$ be the scaled speech STFT coefficients.

Posterior mean estimation

For all $(f, n) \in \mathbb{B}$,

$$\begin{aligned}\hat{\tilde{s}}_{fn} &= \mathbb{E}_{p(\tilde{s}_{fn}|x_{fn};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)}[\tilde{s}_{fn}] \\ &= \mathbb{E}_{p(\mathbf{z}_n|\mathbf{x}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)}\left[\frac{g_n\sigma_f^2(\mathbf{z}_n)}{g_n\sigma_f^2(\mathbf{z}_n) + (\mathbf{W}_b\mathbf{H}_b)_{f,n}}\right] x_{fn}. \end{aligned} \tag{10}$$

Intractable expectation $\rightarrow$ Markov chain Monte Carlo.

Training the generative model with variational autoencoders

Problem setting

- **Training dataset** of STFT speech time frames: $\mathbf{s} = \{\mathbf{s}_n \in \mathbb{C}^F\}_{n=0}^{N_{tr}-1}$.

- **Generative model** (reminder): Independently for all $(f, n) \in \mathbb{B}$:

$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_L);$$
$$s_{fn} \mid \mathbf{z}_n \sim \mathcal{N}_c(0, \sigma_f^2(\mathbf{z}_n)),$$

  where $\mathbf{z}_n \in \mathbb{R}^L$ and in the following $\mathbf{z} = \{\mathbf{z}_n\}_{n=0}^{N_{tr}-1}$.

- **Problem**: Learn the parameters $\boldsymbol{\theta}_s$ of this generative model (weights and biases of the neural network).

- Maximum likelihood is intractable $\rightarrow$ variational autoencoders [2].

[2] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes", *Proc. of ICLR*, 2014.

Variational inference

- ▷ Find $q(\mathbf{z} \mid \mathbf{s}; \phi)$ which approximates $p(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\theta}_s)$.

Variational inference

- Find $q(\mathbf{z} \mid \mathbf{s}; \phi)$ which approximates $p(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\theta}_s)$.

- Kullback-Leibler divergence as a measure of fit:

$$D_{KL}\Big(q(\mathbf{z} \mid \mathbf{s}; \phi) \parallel p(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\theta}_s)\Big) = \ln p(\mathbf{s}; \boldsymbol{\theta}_s) - \mathcal{L}(\phi, \boldsymbol{\theta}_s), \tag{11}$$

where

$$\mathcal{L}(\phi, \boldsymbol{\theta}_s) = \underbrace{\mathbb{E}_{q(\mathbf{z}|\mathbf{s};\phi)}\left[\ln p\left(\mathbf{s} \mid \mathbf{z}; \boldsymbol{\theta}_s\right)\right]}_{\text{Reconstruction accuracy}} \underbrace{-D_{KL}\left(q(\mathbf{z} \mid \mathbf{s}; \phi) \parallel p(\mathbf{z})\right)}_{\text{Regularization}}. \tag{12}$$

- We would like to maximize $\mathcal{L}(\phi, \boldsymbol{\theta}_s)$ with respect to both ϕ and $\boldsymbol{\theta}_s$.

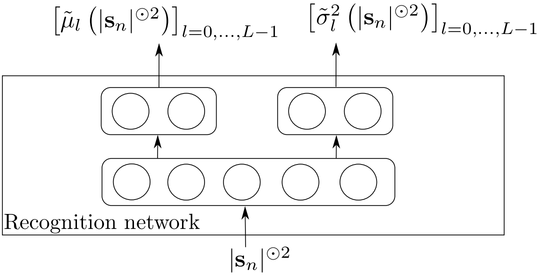
- **We need to define** $q(\mathbf{z} \mid \mathbf{s}; \phi)$.

Variational distribution

Independently for all $n \in \{0, ..., N_{tr} - 1\}$ and $l \in \{0, ..., L - 1\}$:

$$(\mathbf{z}_n)_l \mid \mathbf{s}_n \sim \mathcal{N}\left(\tilde{\mu}_l\left(|\mathbf{s}_n|^{\odot 2}\right), \tilde{\sigma}_l^2\left(|\mathbf{s}_n|^{\odot 2}\right)\right), \quad (13)$$

- $\odot$ denotes element-wise exponentiation;
- $\tilde{\mu}_l : \mathbb{R}_+^F \mapsto \mathbb{R}$ and $\tilde{\sigma}_l^2 : \mathbb{R}_+^F \mapsto \mathbb{R}_+$ are non-linear functions parametrized by ϕ.



Variational free energy

$$\mathcal{L}(\boldsymbol{\theta}_s, \phi) \stackrel{c}{=} -\sum_{f=0}^{F-1}\sum_{n=0}^{N_{tr}-1} \mathbb{E}_{q(\mathbf{z}_n|\mathbf{s}_n;\phi)}\left[d_{IS}\left(\left|s_{fn}\right|^2; \sigma_f^2(\mathbf{z}_n)\right)\right]$$
$$+\frac{1}{2}\sum_{l=1}^{L}\sum_{n=0}^{N_{tr}-1}\left[\ln \tilde{\sigma}_l^2\left(\left|\mathbf{s}_n\right|^{\odot 2}\right) - \tilde{\mu}_l\left(\left|\mathbf{s}_n\right|^{\odot 2}\right)^2 - \tilde{\sigma}_l^2\left(\left|\mathbf{s}_n\right|^{\odot 2}\right)\right], \tag{14}$$

where $d_{IS}(x; y) = x/y - \ln(x/y) - 1$ is the Itakura-Saito (IS) divergence.

- ▷ Intractable expectation approximated by a sample average ("reparametrization trick").
- ▷ Differentiable with respect to both $\boldsymbol{\theta}_s$ and ϕ (backpropagation).
- ▷ Optimized using gradient-ascent-based algorithm.

Experiments

Dataset

- **Clean speech signals**: TIMIT database 🔊.
- **Noise signals**: DEMAND database (domestic environment, nature, office, indoor public spaces, street and transportation) 🔊.
- **Training**:
  - training set of TIMIT database;
  - $\sim$ 4 hours of speech;
  - 462 speakers.
- **Testing**:
  - 168 noisy mixtures at 0 dB signal-to-noise ratio;
  - 1 sentence/speaker in the test set of TIMIT.

Semi-supervised NMF baseline

- Independently for all $(f, n) \in \mathbb{B}$:

$$s_{fn} \sim \mathcal{N}_c(0, (\mathbf{W}_s\mathbf{H}_s)_{f,n}) \quad \text{and} \quad b_{fn} \sim \mathcal{N}_c(0, (\mathbf{W}_b\mathbf{H}_b)_{f,n}).$$

- **Training**: From the observed clean speech signals

$$\min_{\mathbf{W}_s \in \mathbb{R}_+^{F \times K_s}, \mathbf{H}_s \in \mathbb{R}_+^{K_s \times N}} \sum_{(f,n) \in \mathbb{B}} d_{IS}\left(|s_{fn}|^2; (\mathbf{W}_s\mathbf{H}_s)_{f,n}\right).$$

- **Inference**: From the observed mixture signal $x_{fn} = s_{fn} + b_{fn}$,

$$\min_{\mathbf{H}_s \in \mathbb{R}_+^{K_s \times N}, \mathbf{W}_b \in \mathbb{R}_+^{F \times K_b}, \mathbf{H}_b \in \mathbb{R}_+^{K_b \times N}} \sum_{(f,n) \in \mathbb{B}} d_{IS}\left(|x_{fn}|^2; (\mathbf{W}_s\mathbf{H}_s + \mathbf{W}_b\mathbf{H}_b)_{f,n}\right).$$

- **Speech reconstruction**: $\hat{s}_{fn} = \dfrac{(\mathbf{W}_s\mathbf{H}_s)_{f,n}}{(\mathbf{W}_s\mathbf{H}_s + \mathbf{W}_b\mathbf{H}_b)_{f,n}} x_{fn}$

Fully-supervised deep-learning reference method

- Fully-supervised deep-learning approach proposed in [4].
- A deep neural network is trained to map noisy speech log-power spectrograms to clean speech log-power spectrograms.
- Trained with more that 100 different noise types $\rightarrow$ effective in handling unseen noise types.

---

[4] Y. Xu *et al.*, "A regression approach to speech enhancement based on deep neural networks", *IEEE Transactions on Audio, Speech and Language Processing*, 2015.

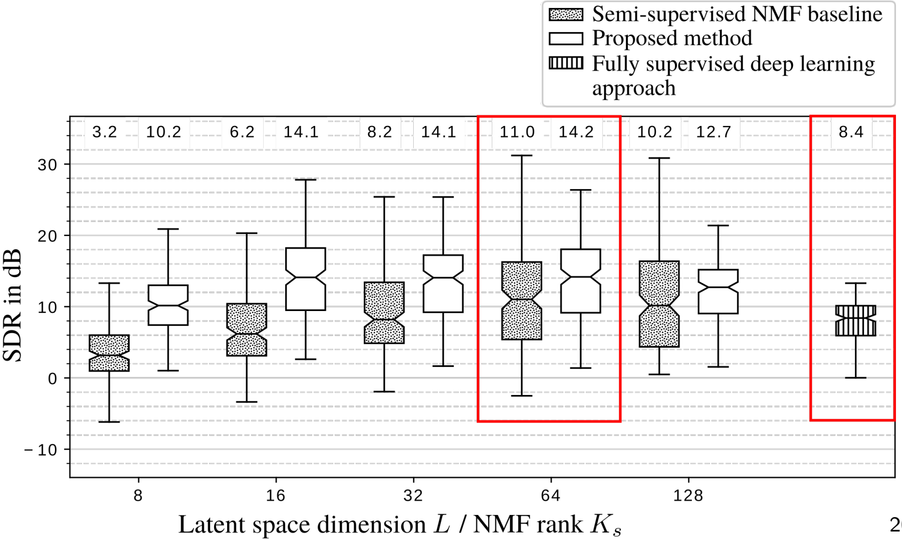
Experiments

- The enhanced speech quality is evaluated in terms of:
  - Signal-to-distortion ratio (SDR) in decibels (dB).
  - Perceptual evaluation of speech quality (PESQ) measure in between -0.5 and 4.5.
  - The higher, the better.
- Different values for the latent dimension L and speech NMF rank K_s:

$$8, 16, 32, 64 \text{ or } 128.$$

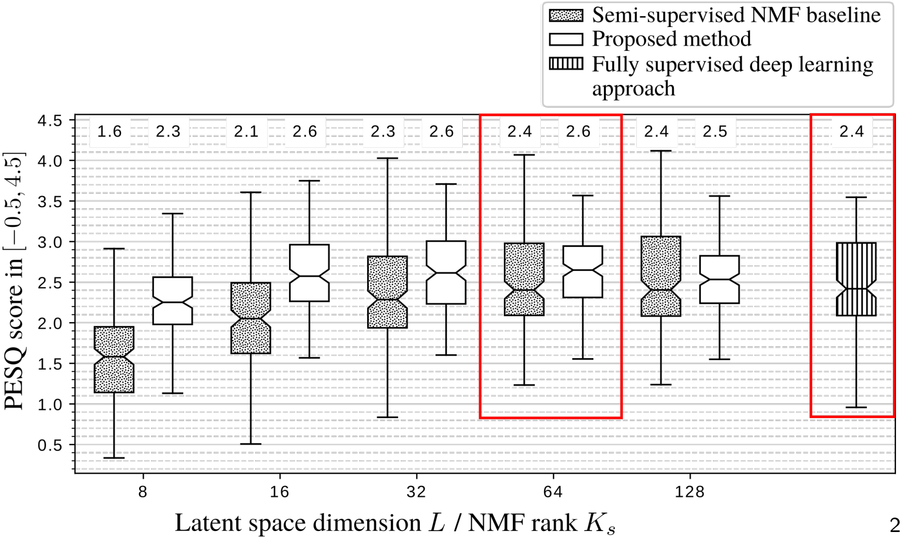
Experimental results (SDR)

Median value indicated above each boxplot.



Experimental results (PESQ)

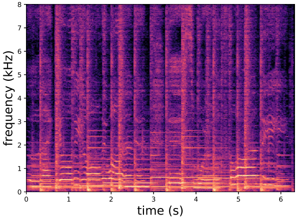
Median value indicated above each boxplot.



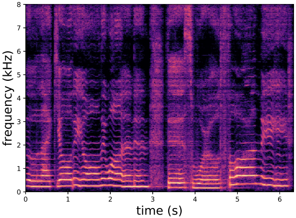
Musical audio example

▷ All models have been trained on speech (not singing voice).

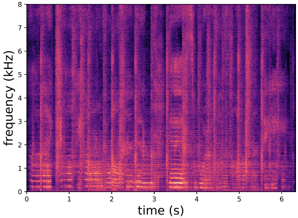
mixture



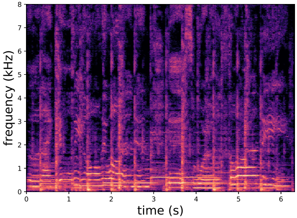
original voice



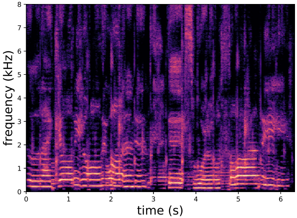
fully-supervised DNN



semi-supervised NMF



proposed



Conclusion

Conclusion

Variational autoencoders are an interesting alternative to supervised NMF models.

Some perspectives:

- Monte Carlo EM is slow $\rightarrow$ variational inference;
- Temporal model on the latent variables;
- Multi-microphone extension;
- Uncertainty propagation for speech information retrieval.

Thank you

Audio examples and code available online:

`https://sleglaive.github.io`