Major Project (CO - 401)

# Speech Denoising Using Non-Negative Matrix Factorization

Degree of Bachelor of Technology
in
**Computer Engineering (COE)**
Under supervision of
**Dr. Shailender Kumar**
(Associate Professor)

By:
**Ankush Kamboj** (2K15/CO/032)
**Anmol Singh** (2K15/CO/33)
**Deepanshu** (2K15/CO/48)

To:
**Department of Computer Science and Engineering**
**Delhi Technological University**
**(Formerly Delhi College of Engineering)**

# Declaration

We hereby certify that the work which is presented in the Major Project entitles “**Speech Denoising using Non-negative matrix factorization**” in fulfilment of the requirement for the award of the Degree of Bachelor of Technology and submitted to the Department of Computer Engineering, Delhi Technological University (Formerly Delhi College Of Engineering), New Delhi is an authentic record of my own, carried out during a period from August 2018 to November 2018, under the supervision of **Dr. Shailender Kumar (Associate Professor, CSE Department)**.

The matter presented in this report has not been submitted by me for the award of any other degree of this or any other Institute/University.

**Ankush Kamboj** (2K15/CO/032)

**Anmol Singh** (2K15/CO/033)

**Deepanshu** (2K15/CO/048)

# Certificate

This is to certify that this project titled “**Speech Denoising Using Non-Negative Matrix Factorization** ” submitted by Ankush Kamboj (2K15/CO/032), Anmol Singh (2K15/CO/33), Deepanshu (2K15/CO/148) in partial fulfilment for the requirements for the award of Bachelor of Technology Degree in Computer Engineering (COE) at Delhi Technological University is an authentic work carried out by the students under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university or institute for the award of any degree or diploma.

**Dr. Shailender Kumar**
(Associate Professor)
Department of Computer Science and Engineering
Delhi Technological University
Delhi – 110042

# Acknowledgement

Firstly, we express our heartiest gratitude towards the authorities who gave us a chance to
explore the intricacies of various aspects of **Speech signals and non-negative matrix factorization**.

We are grateful to **Dr. Rajni Jindal, HOD** (Department of Computer Science and Engineering), Delhi Technological University, New Delhi and all other faculty members of our department for their astute guidance throughout the project.

We would also sincerely thank our esteemed mentor, **Dr. Shailender Kumar**, who lent a huge helping hand in the process of making this project with her valuable guidance and blessings.

In the end, we would thank our families for their extended support throughout the project.

**Ankush Kamboj** (2K15/CO/032)

**Anmol Singh** (2K15/CO/33)

**Deepanshu** (2K15/CO/048)

# Table of Contents

# Introduction

Speech enhancement is a technique that improves quality of speech signal. The speech signal gets degraded because of various types of noise like background noise, reverberation, babble etc. The clean speech signal is necessary for applications such as speech or speaker recognition, hearing aids, mobile communication. Speech enhancement techniques are used to enhance the corrupted signal by reducing noise. It is assumed that the noise is additive. It is assumed that the noise characteristics change very slowly as compared to the signal. This is the underlying assumption in speech enhancement methods.Speech enhancement deals with processing of noisy speech signals, aiming at improving their perception by human or their correct decoding by machines. The objective of enhancement is improvement in intelligibility and/or overall perceptual quality of degraded speech signal using audio signal processing techniques. Enhancing of speech degraded by noise, or noise reduction, is the most important field of speech enhancement.

Research in musical signal analysis and source separation has historically been technically and philosophically unrelated. Musical signal analysis often involves trying to estimate how traditional musical constructs can explain a musical signal, whereas source separation has been a field using varied approaches to separate sounds. Although most often connections between these two areas are not drawn, recent research has been exposing a converging trend.

Noisy speech signals are a common problem in many applications. For example, Automatic Speech Recognition (ASR). Machine understanding of what was said and how it was said is still far from humans. Well-developed methods of speech denoising will allow to improve different supporting systems dealing with speech processing like Siri, Yandex Maps etc. Speech denoising techniques may also be useful for increasing quality of telephone, skype and other types of conversations. One more possible application is hearing aids devices for people with auditory disabilities.

For defining the scope of this project, we would like to propose the following tasks:

- Problem Formulation and Proposed Solution

  – General formulation of denoising problems.

  – Learn frequency patterns from speech via non negative matrix factorization

  – Decomposing new signal on joint dictionary of patterns

- NMF : How to compute ?

  – Studied about different techniques to implement non negative matrix  factorization.

  – Use techniques like Multiplicative Updates, Alternating non negative Least Squares and compared them.

- Spectrogram and Signal Reconstruction

  – Spectrogram reconstruction from clean speech projection of joint dictionary.

  – Reconstructing speech signal from the reconstructed spectrogram

# Literature Survey

The speech spectral distributions are predominantly harmonic as speech tends to be, whereas the ambient noise spectral distributions are more wideband and noisy, better describing their type of sound. Had we been confronted with a situation where there was a mixture of speech and ambient noise, it is safe to assume that the mixture spectrogram will contain some mix of both the noise and the speech spectral distributions, each describing the presence of each sound type in the signal. This means that if we know these spectral distributions beforehand we can try to reconstruct the now unknown mixture using them. This also means that the subset of spectral distributions that describes the speech would most likely account for the speech part of the mixture, whereas the spectral distributions that describe the ambient noise will do so for the noise. We can therefore create selective reconstructions of the mixture using one spectral basis subset at a time to extract individual sound classes. [6].

In [1], the authors discuss the NMF based method to enhance speech signal when provided with spectral knowledge of the noise has been presented. This method has been applied to the reduction of the non stationary noise produced by the sensors of a robotic assistant. When tested on a corpus of speech signals, the proposed method achieved better performances than well known VAD based denoising.

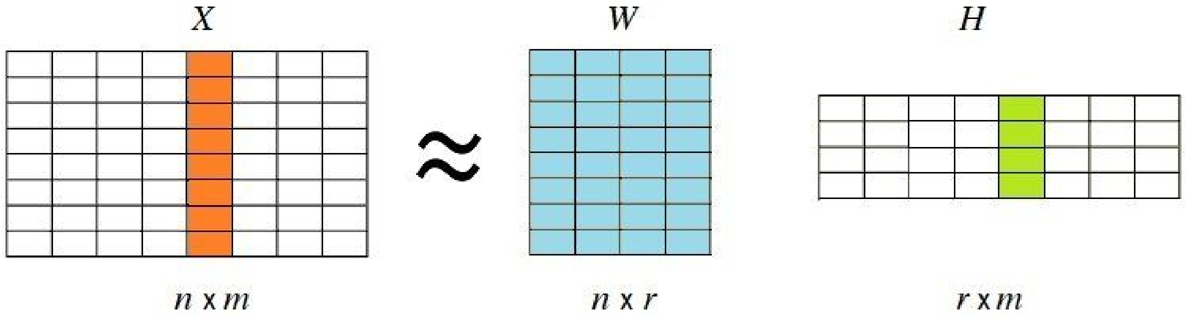
Standard approaches such as spectral subtraction and Wiener filtering require signal and/or noise estimates and therefore are typically restricted to stationary or quasi-stationary noise inpractice. Non-negative matrix factorization, popularized by Lee and Seung [17], finds a locally optimal choice of W and H to solve the matrix equation V = W*H. This provides a way of decomposing a signal into a convex combination of non-negative building blocks. When the signal, V , is a spectrogram and the building blocks, W , are a set of specific spectral shapes, Smaragdis [6] showed how NMF can be used to separate single-channel mixtures of sounds by associating different sets of building blocks with different sound sources [8].

Speech enhancement in presence of background noise is an important problem that exists for a long time and still is widely studied nowadays. The efficient single-channel noise suppression (or noise reduction) techniques are essential for increasing quality and intelligibility of speech, as well as improving noise robustness for automatic speech recognition (ASR) systems [4].

# Algorithms

## Non-Negative Matrix Factorization

Non-negative matrix factorization (NMF or NNMF), also non-negative matrix approximation is a group of algorithms in multivariate analysis and linear algebra where a matrix V is factorized into (usually) two matrices W and H, with the property that all three matrices have no negative elements. This non-negativity makes the resulting matrices easier to inspect. Also, in applications such as processing of audio spectrograms or muscular activity, non-negativity is inherent to the data being considered. Since the problem is not exactly solvable in general, it is commonly approximated numerically. NMF finds applications in such fields as astronomy, computer vision, document clustering, chemometrics, audio signal processing, recommender systems, and bioinformatics.



Matrix Factorization

## NMF Background

Let matrix V be the product of the matrices W and H,

$$V = WH$$

Matrix multiplication can be implemented as computing the column vectors of V as linear combinations of the column vectors in W using coefficients supplied by columns of H. That is, each column of V can be computed as follows:

$$v_i = Wh_i$$

where $v_i$ is the i-th column vector of the product matrix V and $h_i$ is the i-th column vector of the matrix H.

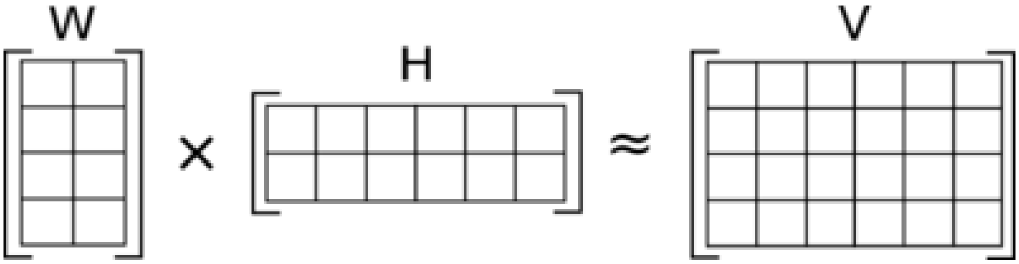
When multiplying matrices, the dimensions of the factor matrices may be significantly lower than those of the product matrix and it is this property that forms the basis of NMF. NMF generates factors with significantly reduced dimensions compared to the original matrix. For example, if V is an m × n matrix, W is an m × p matrix, and H is a p × n matrix then p can be significantly less than both m and n.

Here is an example based on a text-mining application:

- Let the input matrix (the matrix to be factored) be V with 10000 rows and 500 columns where words are in rows and documents are in columns. That is, we have 500 documents indexed by 10000 words. It follows that a column vector v in V represents a document.
- Assume we ask the algorithm to find 10 features in order to generate a features matrix W with 10000 rows and 10 columns and a coefficients matrix H with 10 rows and 500 columns.
- The product of W and H is a matrix with 10000 rows and 500 columns, the same shape as the input matrix V and, if the factorization worked, it is a reasonable approximation to the input matrix V.
- From the treatment of matrix multiplication above it follows that each column in the product matrix WH is a linear combination of the 10 column vectors in the features matrix W with coefficients supplied by the coefficients matrix H.

This last point is the basis of NMF because we can consider each original document in our example as being built from a small set of hidden features. NMF generates these features.

It is useful to think of each feature (column vector) in the features matrix W as a document archetype comprising a set of words where each word's cell value defines the word's rank in the feature: The higher a word's cell value the higher the word's rank in the feature. A column in the coefficients matrix H represents an original document with a cell value defining the document's rank for a feature. We can now reconstruct a document (column vector) from our input matrix by a linear combination of our features (column vectors in W) where each feature is weighted by the feature's cell value from the document's column in H.



Matrix Approximation

## NMF Applications

(i) Astronomy

In astronomy, NMF is a promising method for dimension reduction in the sense that astrophysical signals are non-negative. NMF has been applied to the spectroscopic observations and the direct imaging observations as a method to study the common properties of astronomical objects and post-process the astronomical observations. The advances in the spectroscopic observations by Blanton & Roweis (2007) takes into account of the uncertainties of astronomical observations, which is later improved by Zhu (2016) where missing data are also considered and parallel computing is enabled. Their method is then adopted by Ren et al. (2018) to the direct imaging field as one of the methods of detecting exoplanets, especially for the direct imaging of circumstellar disks.

Ren et al. (2018) are able to prove the stability of NMF components when they are constructed sequentially (i.e., one by one), which enables the linearity of the NMF modeling process; the linearity property is used to separate the stellar light and the light scattered from the exoplanets and circumstellar disks.

### (ii) Text mining

NMF can be used for text mining applications. In this process, a document-term matrix is constructed with the weights of various terms (typically weighted word frequency information) from a set of documents. This matrix is factored into a term-feature and a feature-document matrix. The features are derived from the contents of the documents, and the feature-document matrix describes data clusters of related documents.

### (iii) Non-stationary speech denoising

Speech denoising has been a long lasting problem in audio signal processing. There are lots of algorithms for denoising if the noise is stationary. For example, the Wiener filter is suitable for additive Gaussian noise. However, if the noise is non-stationary, the classical denoising algorithms usually have poor performance because the statistical information of the non-stationary noise is difficult to estimate. Schmidt et al.[63] use NMF to do speech denoising under non-stationary noise, which is completely different from classical statistical approaches. The key idea is that clean speech signal can be sparsely represented by a speech dictionary, but non-stationary noise cannot. Similarly, non-stationary noise can also be sparsely represented by a noise dictionary, but speech cannot.

The algorithm for NMF denoising goes as follows. Two dictionaries, one for speech and one for noise, need to be trained offline. Once a noisy speech is given, we first calculate the magnitude of the Short-Time-Fourier-Transform. Second, separate it into two parts via NMF, one can be sparsely represented by the speech dictionary, and the other part can be sparsely represented by the noise dictionary. Third, the part that is represented by the speech dictionary will be the estimated clean speech.

(iv) Bioinformatics

NMF has been successfully applied in bioinformatics for clustering gene expression and DNA methylation data and finding the genes most representative of the clusters. In the analysis of cancer mutations it has been used to identify common patterns of mutations that occur in many cancers and that probably have distinct causes.

(v) Nuclear imaging

NMF, also referred in this field as factor analysis, has been used since the 1980s to analyze sequences of images in SPECT and PET dynamic medical imaging. Non-uniqueness of NMF was addressed using sparsity constraints.

# Algorithms to compute NMF

The following methods for NMF were implemented:

1. **Multiplicative Update (MU)** method with both KL divergence and Frobenius norm (we used the Nimfa library for that).
   The non-negative matrix factorization problem is non-convex in W and H but it is convex in only W or only H. To optimize the above problem, we use a block coordinate descent scheme where we optimize with respect to W first while keeping H fixed and then vice versa.

   MU is a kind of gradient descent. On each iteration we choose such step that vector h updates multiplicative. We divide gradient on positive and negative part. On each iteration we multiply our vector on $[\nabla_H^-]_{kj}/[\nabla_H^+]_{kj}$

$$[\nabla_H]_{kj} = \frac{\partial D(V, WH)}{\partial h_{kj}} = [\nabla_H^+]_{kj} - [\nabla_H^-]_{kj}$$

$$h_{kj} \leftarrow h_{kj} - \frac{h_{kj}}{[\nabla_H^+]_{kj}}([\nabla_H^+]_{kj} - [\nabla_H^-]_{kj}) = \frac{[\nabla_H^-]_{kj}}{[\nabla_H^+]_{kj}} h_{kj}$$

   This approach allows us not to think about non-negativity of matrices W and H, if they initialized as non-negative matrices, they can't become negative in process of descent.

2. **Alternating Nonnegative Least Squares (ANLS)** with Frobenius norm.
   The so-called alternating nonnegative least squares (ANLS) algorithm for (NMF) minimizes (exactly) the cost function alternatively over factors V and W so that a stationary point of (NMF) is obtained in the limit.

   Structure of the algorithm is the following:

   1) Initialize $W_{ia}^{1} \geq 0$, $W_{ib}^{1} \geq 0$, □ a,i,b,j.

2) For k = 1, 2….

$$W^{k+1} = \arg\min_{W\geq0} D(V, W H^k)$$
$$H^{k+1} = \arg\min_{H\geq0} D(V, W^k H)$$

For solving subproblem we used projected gradient method.

3. **ANLS Frobenius norm and L1 regularization of the matrix H.**

The concept is the same but loss function is a bit changed.

$$W^*, H^* = \arg\min_{W\geq0,\, H\geq0} \left( \|V - WH\|_F^{\,2} + \lambda\|H\|_1 \right)$$

This method was already implemented in sklearn, also we tried to use CVX to solve the subproblem.

4. **Quasi-Newton method for KL divergence.**

Probably the only method of second-order.

On each iteration we should do the following updates:

$$W \leftarrow \max(\varepsilon, W - H_W^{\;-1} \nabla_W D_{KL})$$
$$H \leftarrow \max(\varepsilon, H - H_H^{\;-1} \nabla_H D_{KL})$$

Where $H_W$ and $H_H$ are hessians.

# Short-time Fourier Transform (STFT)

It is a fourier related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time. In practice, the procedure for computing STFTs is to divide a longer time signal into shorter segments of equal length and then compute the Fourier transform separately on each shorter segment. This reveals the Fourier spectrum on each shorter segment. One then usually plots the changing spectra as a function of time.

Simply, in the continuous-time case, the function to be transformed is multiplied by a window function which is non-zero for only a short period of time. The fourier transform (a one-dimensional function) of the resulting signal is taken as the window is slid along the time axis, resulting in a two-dimensional representation of the signal.

Mathematically, this is written as:

$$\textbf{STFT}\ \{x(t)\}(\tau,\ \omega)\ \equiv\ X(\tau,\ \omega)\ =\ \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-jwt}\,dt$$

where *w*(*t*) is the window function, commonly a hann window or gaussian window centered around zero, and

*x*(*t*) is the signal to be transformed (note the difference between *w* and ω).

*X*(τ,ω) is essentially the Fourier Transform of *x*(*t*)*w*(*t*-τ), a complex function representing the phase and magnitude of the signal over time and frequency. Often phase unwrapping is employed along either or both the time axis, τ, and frequency axis, ω, to suppress any jump discontinuity of the phase result of the STFT. The time index τ is normally considered to be "*slow*" time and usually not expressed in as high resolution as time *t*.

STFTs as well as standard Fourier transforms and other tools are frequently used to analyze music. The spectrogram can, for example, show frequency on the horizontal axis, with the lowest frequencies at left, and the highest at the right. The height of each bar (augmented by color) represents the amplitude of the frequencies within that band. The depth dimension represents time, where each new bar was a separate distinct transform. Audio engineers use this kind of visual to gain information about an audio sample, for example, to locate the frequencies of specific noises (especially when used with greater frequency resolution) or to find frequencies which may be more or less resonant in the space where the signal was recorded. This information can be used for equalization or tuning other audio effects.

# Inverse STFT

The STFT is invertible, that is, the original signal can be recovered from the transform by the Inverse STFT. The most widely accepted way of inverting the STFT is by using the overlap-add method (OLA), which also allows for modifications to the STFT complex spectrum. This makes for a versatile signal processing method, referred to as the overlap and add with modifications method.

So the Fourier Transform can be seen as a sort of phase coherent sum of all of the STFTs of *x*(*t*). Since the inverse Fourier transform is

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{+j\omega t} \, d\omega,$$

then *x*(*t*) can be recovered from *X*(τ,ω) as

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(\tau, \omega) e^{+j\omega t} \, d\tau \, d\omega.$$

or

$$x(t) = \int_{-\infty}^{\infty} \left[ \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\tau, \omega) e^{+j\omega t} \, d\omega \right] d\tau.$$

It can be seen, comparing to above that windowed "grain" or "wavelet" of *x*(*t*) is

$$x(t) w(t - \tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\tau, \omega) e^{+j\omega t} \, d\omega.$$

# Dataset

We use the **NOIZEUS** dataset  for evaluation of speech enhancement algorithms. The noisy speech corpus (NOIZEUS) was developed to facilitate comparison of speech enhancement algorithms among research groups. The noisy database contains 30 IEEE sentences (produced by three male and three female speakers) corrupted by eight different real-world noises at different SNRs. The noise was taken from the AURORA database and includes suburban train noise, babble, car, exhibition hall, restaurant, street, airport and train-station noise. This corpus is available to researchers free of charge.

## Speech Material:

Thirty sentences from the IEEE sentence database  were recorded in a sound-proof booth using Tucker Davis Technologies (TDT) recording equipment. The sentences were produced by three male and three female speakers. The IEEE database (720 sentences) was used as it contains phonetically-balanced sentences with relatively low word-context predictability. The thirty sentences  were selected from the IEEE database so as to include all phonemes in the American English language. The list of sentences recorded for NOIZEUS are given in table below. The sentences were originally sampled at 25 kHz and downsampled to 8 kHz.

| Filename | Speaker | Gender | Sentence text |
|---|---|---|---|
| sp01.wav | CH | M | The birch canoe slid on the smooth planks. |
| sp02.wav | CH | M | He knew the skill of the great young actress. |
| sp03.wav | CH | M | Her purse was full of useless trash. |
| sp04.wav | CH | M | Read verse out loud for pleasure. |
| sp05.wav | CH | M | Wipe the grease off his dirty face. |
| sp06.wav | DE | M | Men strive but seldom get rich. |
| sp07.wav | DE | M | We find joy in the simplest things. |
| sp08.wav | DE | M | Hedge apples may stain your hands green. |
| sp09.wav | DE | M | Hurdle the pit with the aid of a long pole. |
| sp10.wav | DE | M | The sky that morning was clear and bright blue. |
| sp11.wav | JE | F | He wrote down a long list of items. |
| sp12.wav | JE | F | The drip of the rain made a pleasant sound. |
| sp13.wav | JE | F | Smoke poured out of every crack. |
| sp14.wav | JE | F | Hats are worn to tea and not to dinner. |
| sp15.wav | JE | F | The clothes dried on a thin wooden rack. |
| sp16.wav | KI | F | The stray cat gave birth to kittens. |
| sp17.wav | KI | F | The lazy cow lay in the cool grass. |

| | | | |
|---|---|---|---|
| sp18.wav | KI | F | The friendly gang left the drug store. |
| sp19.wav | KI | F | We talked of the sideshow in the circus. |
| sp20.wav | KI | F | The set of china hit the floor with a crash. |
| sp21.wav | SI | M | Clams are small, round, soft and tasty. |
| sp22.wav | SI | M | The line where the edges join was clean. |
| sp23.wav | SI | M | Stop whistling and watch the boys march. |
| sp24.wav | SI | M | A cruise in warm waters in a sleek yacht is fun. |
| sp25.wav | SI | M | A good book informs of what we ought to know. |
| sp26.wav | TI | F | She has a smart way of wearing clothes. |
| sp27.wav | TI | F | Bring your best compass to the third class. |
| sp28.wav | TI | F | The club rented the rink for the fifth night. |
| sp29.wav | TI | F | The flint sputtered and lit a pine torch. |
| sp30.wav | TI | F | Let's all join as we sing the last chorus. |

# Filtering

To simulate the receiving frequency characteristics of telephone handsets, the speech and noise signals were filtered by the modified Intermediate Reference System (IRS) filters used in ITU-T P.862 [1] for evaluation of the PESQ measure.

*Ref: ITU P.862 (2000). Perceptual evaluation of speech quality (PESQ), and objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs. ITU-T Recommendation P. 862*

## Adding Noise

Noise is artificially added to the speech signal as follows. The IRS filter is independently applied to the clean and noise signals. The active speech level of the filtered clean speech signal is first determined using the method B of ITU-T P.56 [3]. A noise segment of the same length as the speech signal is randomly cut out of the noise recordings, appropriately scaled to reach the desired SNR level and finally added to the filtered clean speech signal.

Noise signals were taken from the AURORA database and included the following recordings from different places:

- Babble (crowd of people)
- Car
- Exhibition hall
- Restaurant
- Street
- Airport
- Train station
- Train

The noise signals were added to the speech signals at SNRs of 0dB, 5dB, 10dB, and 15dB.

# Work Done

## Problem Formulation

The general formulation of the denoising problem is the following: we need to build transformation function

$$f : R^* \Rightarrow R^*$$

which maps the noisy signal into the clean one with the least possible error. Error can be expressed in many different ways in that case. Let's consider two of them:

- Signal error:

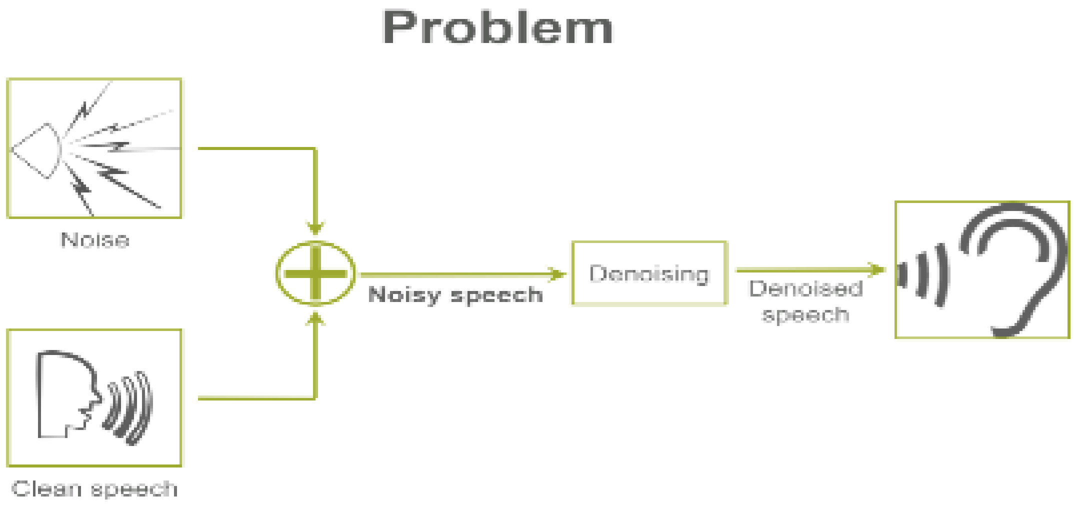
$$f^* = arg\ min_{f \in F} \|s_{clean} - f(s_{noisy})\|$$

It corresponds to the difference of the clean signal and reconstructed one.

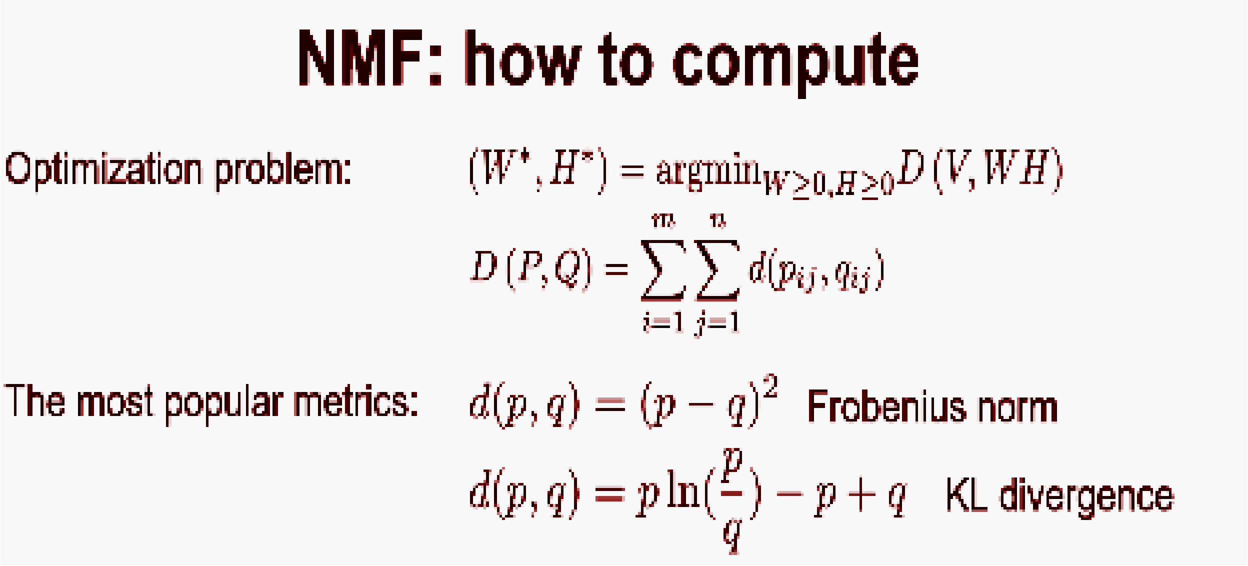
- Spectral error:

$$f^* = arg\ min_{f \in F} \|STFT(s_{clean}) - STFT(f(s_{noisy}))\|$$

In general, the idea is the same - measure the error between the clean and reconstructed signal. But now the distance is calculated between their spectrograms.In this work we're going to use the Spectral error because of the better robustness.



# Comparison of various NMF techniques and their Computation



## NMF: how to compute

Optimization problem: $(W^*, H^*) = \text{argmin}_{W \geq 0, H \geq 0} D(V, WH)$

$$D(P,Q) = \sum_{i=1}^{m} \sum_{j=1}^{n} d(p_{ij}, q_{ij})$$

The most popular metrics: $d(p,q) = (p-q)^2$ Frobenius norm

$d(p,q) = p \ln(\frac{p}{q}) - p + q$ KL divergence

To solve the problem of denoising, nonnegative matrix factorization (NMF) is frequently used.

The goal of NMF is to decompose matrix V ∈ $R^{n*m}$ with non-negative elements into product of two matrices W ∈ $R^{n*k}$ and H ∈ $R^{k*m}$ with nonnegative values and k<min(m, n), such that V ≈ W.H . In other words one should solve the following problem:

$$W^* , H^* = arg\ min_{H>0,W>0} \text{ D(V,WH);}$$

k<min(m, n)

where D(V,WH) is a “distance” between two matrices. Various functions are used for D, which leads to different problems and different solutions for them.

So basically NMF is a group of algorithms in multivariate analysis and linear algebra where a matrix V is factorized into (usually) two matrices W and H, with the property that all three matrices have no negative elements. This non-negativity makes the resulting matrices easier to inspect. Also, in applications such as processing of audio spectrograms or muscular activity, non-negativity is inherent to

the data being considered. Since the problem is not exactly solvable in general, it is commonly approximated numerically.
The common choice here is to use so called separable distance that is when D(P,Q) can be decomposed into elementwise computations. The advantage of such functions is that computations proceed much faster than with usual matrix norms and saved time can be used to do multistart which is much more useful regarding the multi extremality of the problem. In our work we use two following distances:

1. Kullback–Leibler divergence

$$D_{KL}(P,Q) = \sum_{i,j=1} d(p_{ij}, q_{ij}),$$
$$d(p,q) = p \ln(p/q) - p + q.$$

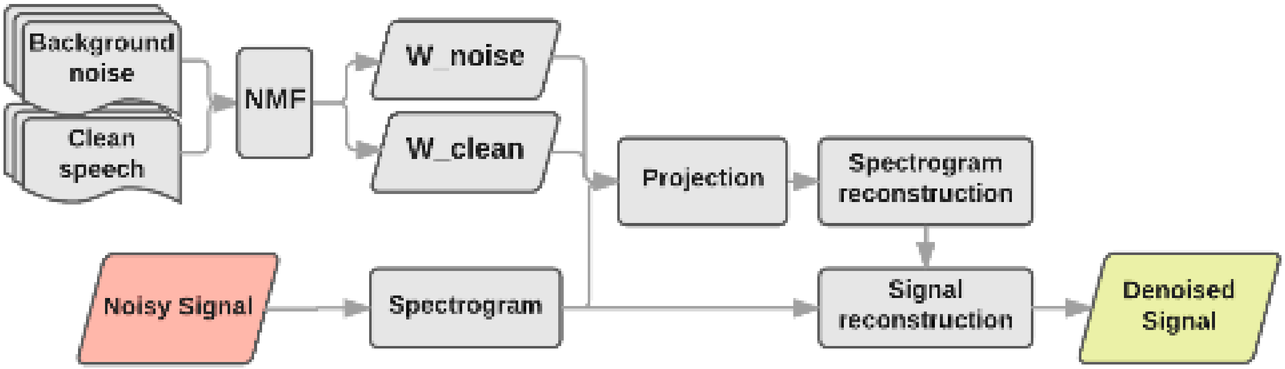
2. Frobenius norm of matrices difference

$$D_{KL}(P,Q) = ||P - Q||_F$$

Also we try to use the regularization to preserve the sparseness of the matrix which should increase the quality of the solution.

The two methods of computing NMF namely Multiplicative Updates and Alternating Least Squares were compared with respect to the two norms Frobenius and KL divergence.

# Proposed Pipeline For Denoising (aka Denoising Workflow)



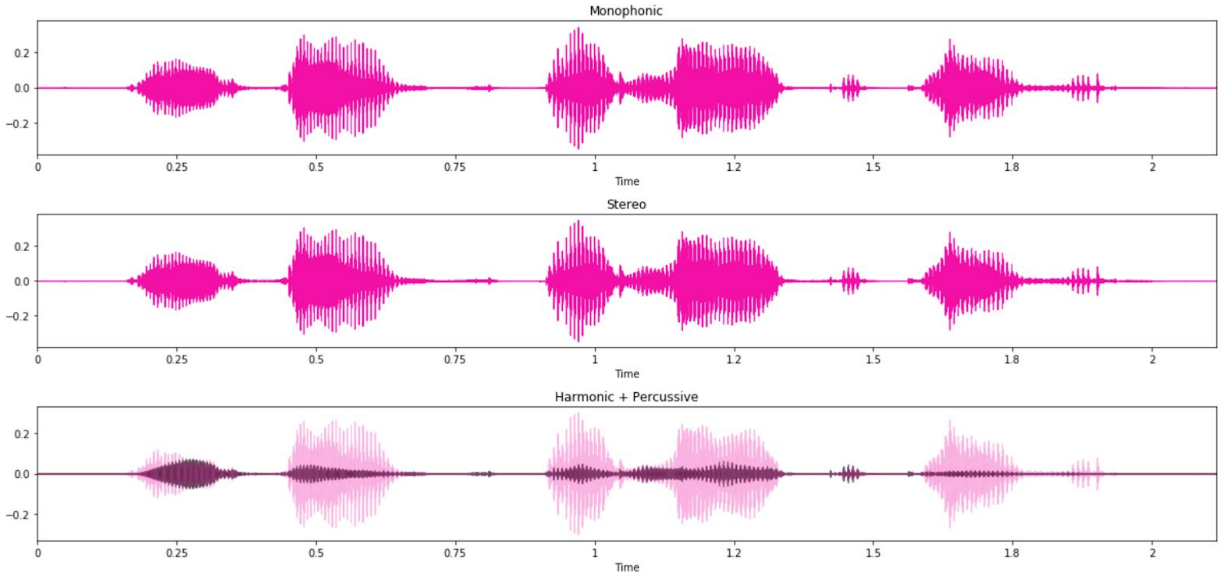
## Spectral Analysis:

Eg: Audio: sp04_16.wav

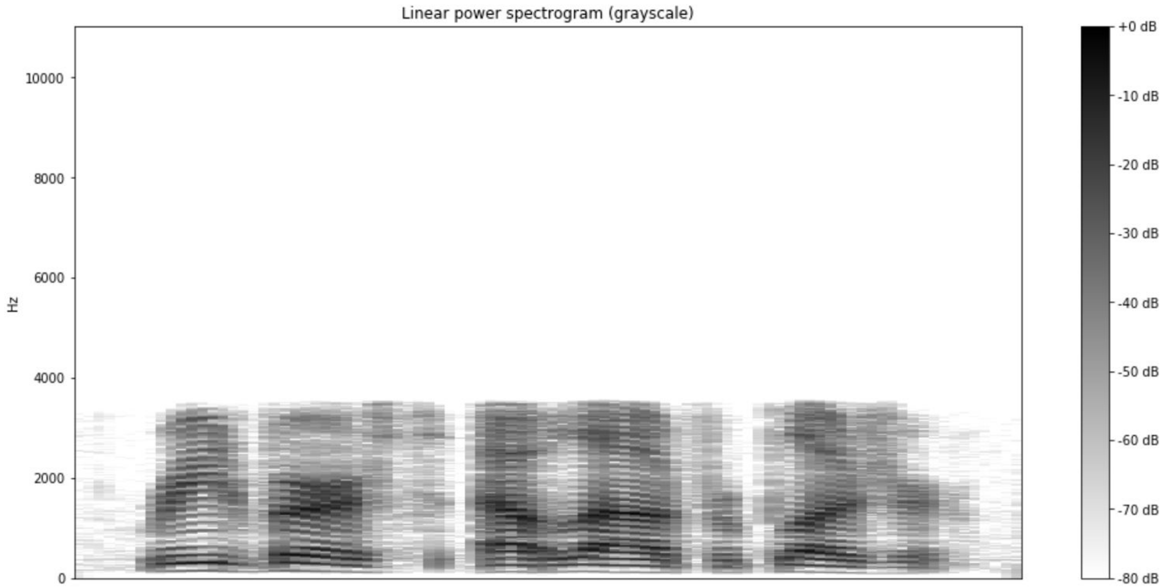
Noise: street_16.wav

### Clean Audio

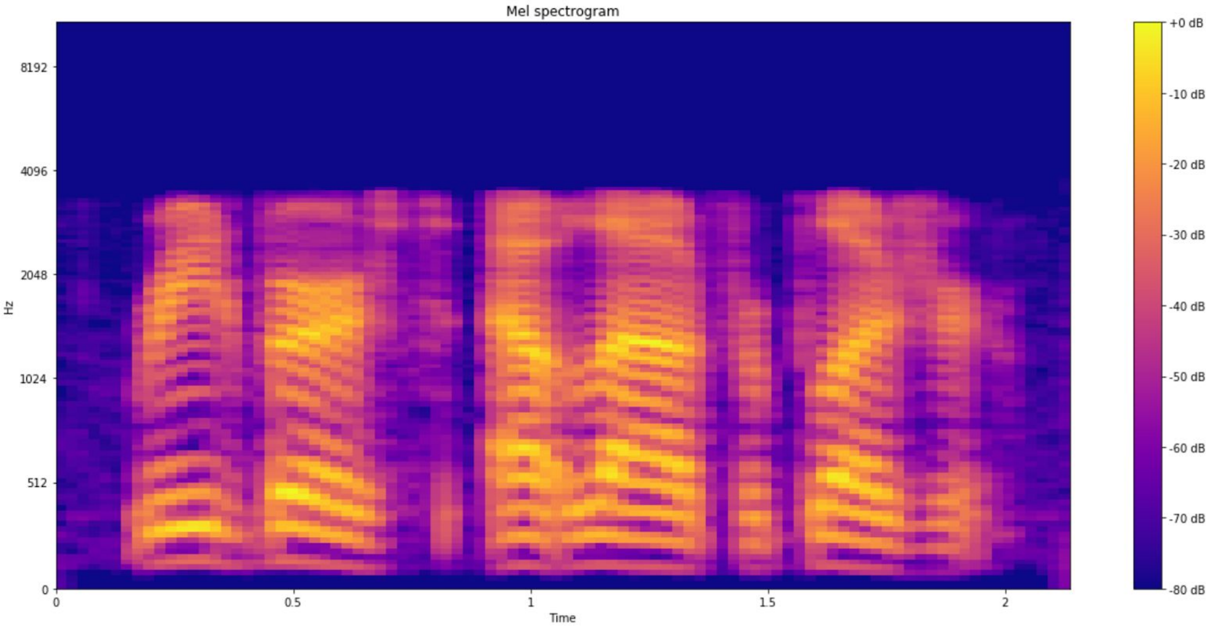
1. **Wave Plots:**



## 2. Linear Power Spectrogram:

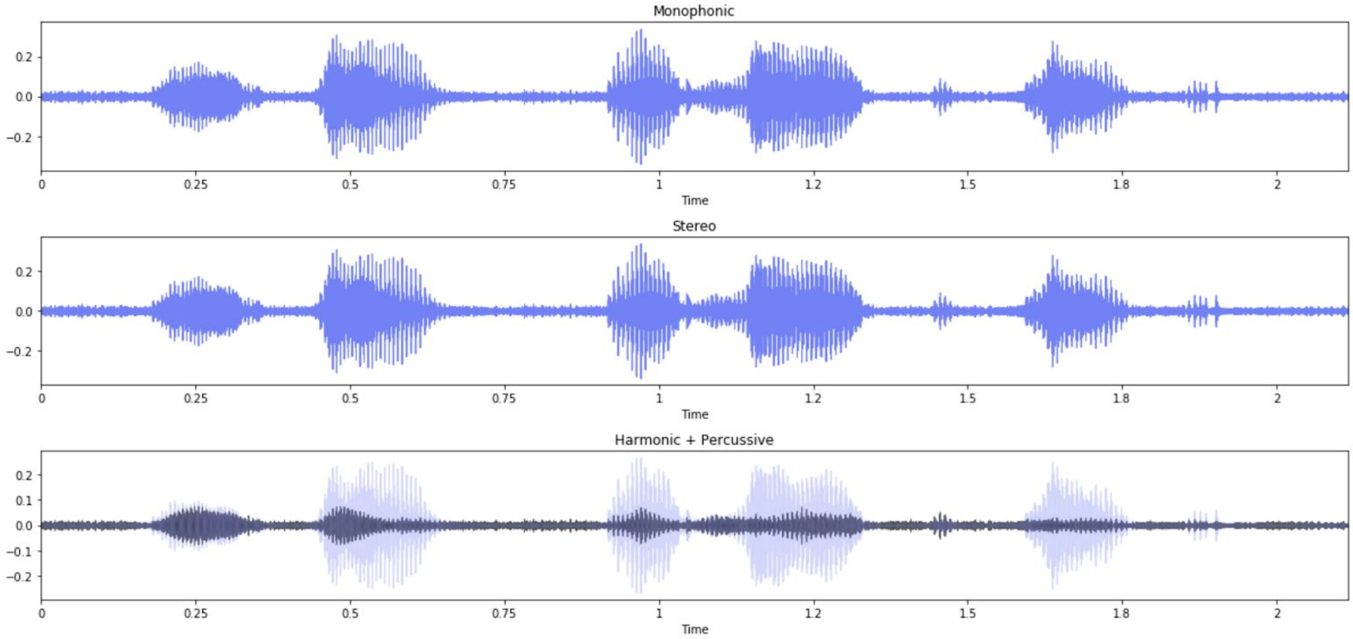


## 3. Mel-spectrogram:

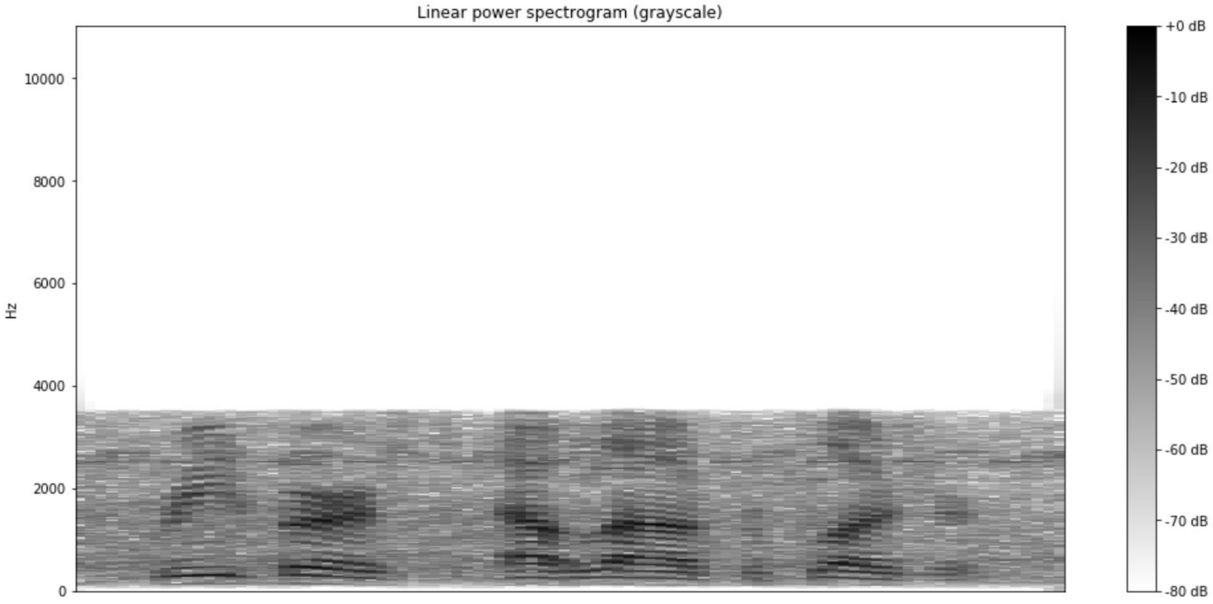


## Speech Audio:

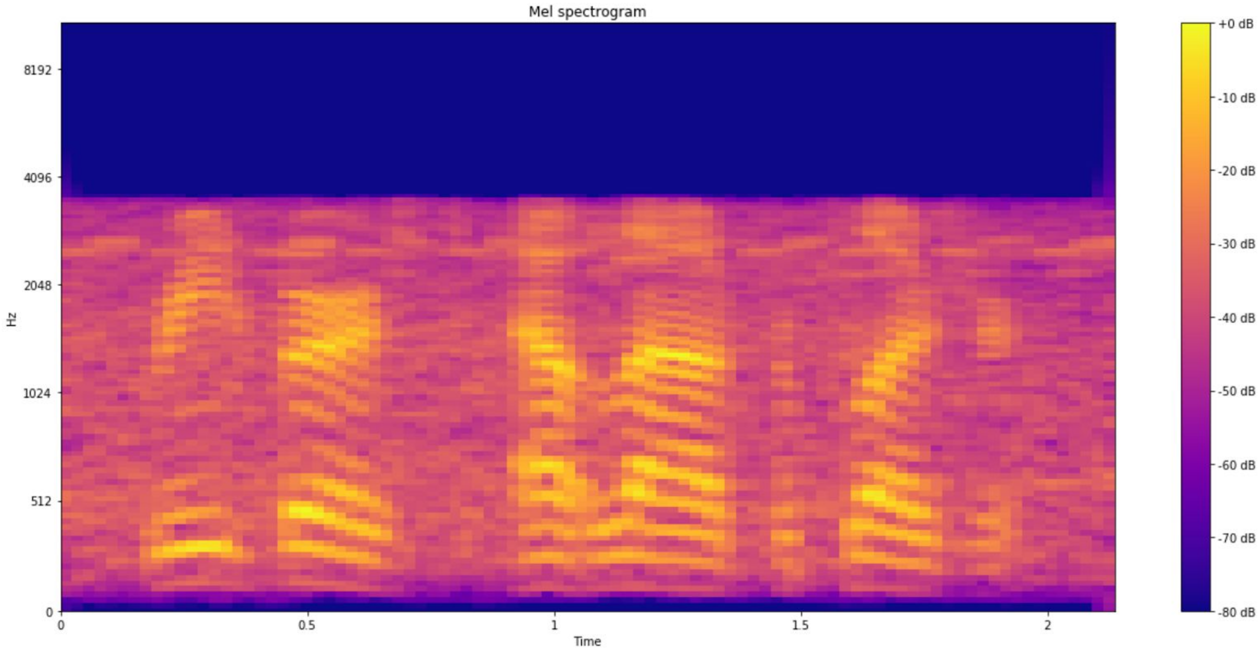
### 1. Wave Plots:



### 2. Linear Power Spectrogram:

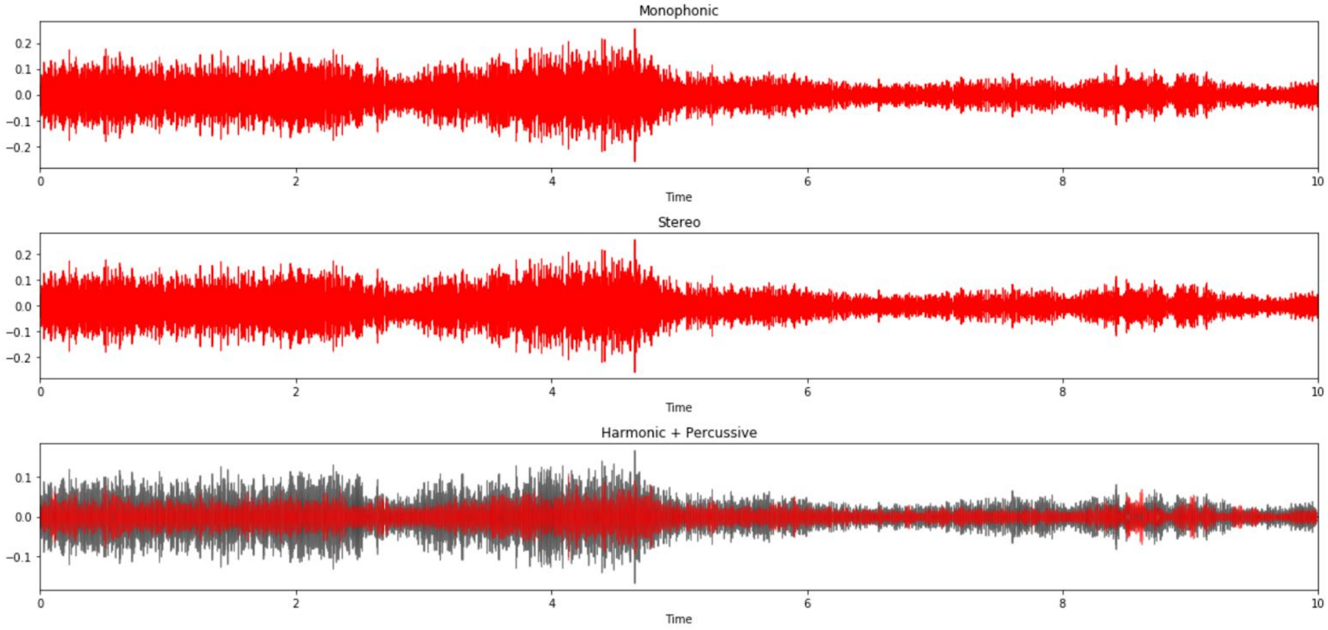


### 3. Mel Spectrogram:

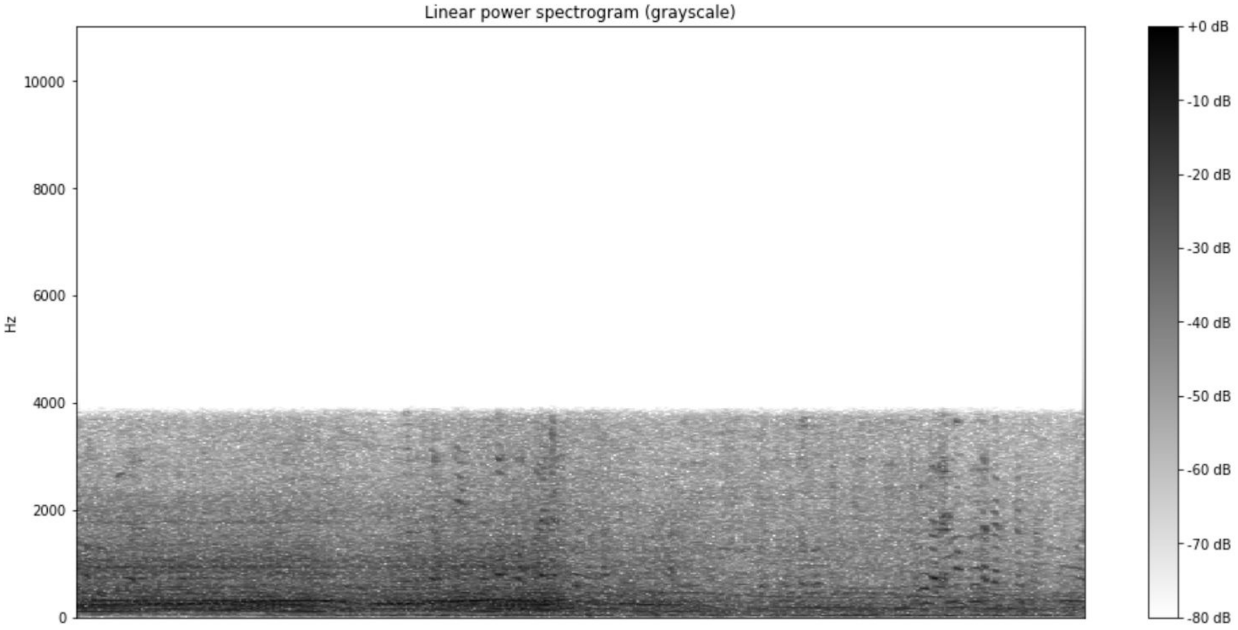


## Noise Audio:

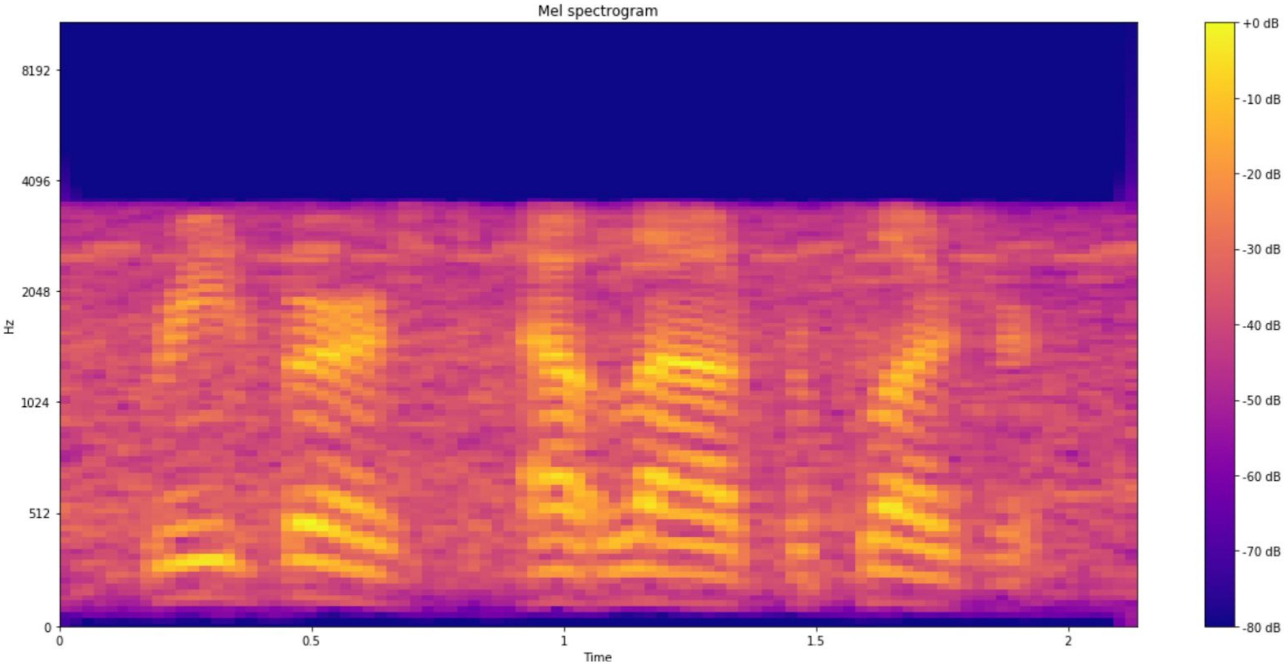
1. **Wave Plots:**



## 2. Linear Power Spectrogram



## 3. Mel Spectrogram



## The steps of the training stage are :

1. Having background noise $s_{noise}$ and clean speech $s_{clean}$ we obtain their spectrograms $S_{noise}$ and $S_{clean}$. This is done by means of the Short-Term Fourier Transform that simply slides window across the signal and does usual FFT inside it.

2. Working with complex spectrograms is not convenient, thus we want to obtain a real matrix from it. So we took the absolute values of each cell which corresponds to the magnitude matrix of the initial spectrogram. In that way we obtain $V_{clean} = |S_{clean}|$, $V_{noise} = |S_{noise}|$.

3. After that we apply NMF to represent the magnitude matrices V as a product of two nonnegative matrices W and H. The interpretation of these matrices is the following: W contains frequency “building blocks” or patterns of a signal, while H contains time-activation information about it - when and how strong each pattern should be applied to form the initial signal in the best way. The intuitive toy example with the piano notes sequence can also be found in the code .

4. Through the NMF we learn dictionaries for both clean speech and background noise. The important thing to mention here is the hidden dimension (rank) of the NMF. Rank of decomposition for the clean speech is chosen approximately equal to the number of phonemes (distinct "building blocks" of speech) in the English language which is about 40.

## The denoising stage then is the following :

1. Take the spectrogram of the noisy signal.

2. Project it on the already learned and joined vocabularies of clean and noise patterns - $W_{joined}$ = ($W_{clean}$ $W_{noise}$). One can see the scheme in the figure given below. It is done by means of the subroutine of ANLS algorithm where one matrix (W in this case) is fixed. Then we take only the “clean” component of this decomposition and our denoised magnitudes are $V_{reconstructed}$ = $W_{clean}$ $H_{joined..}$

3. The important step here is how to reconstruct the phase information from the magnitudes. This problem is quite complicated and there are few approaches invented for it. In general case it is impossible to reconstruct the signal precisely. But we choose window parameters in such a way that they overlaps massively and thus implicitly store information about a phase. So reconstruction is possible and can be made in a few different ways which are described in details in our IPython Notebook.

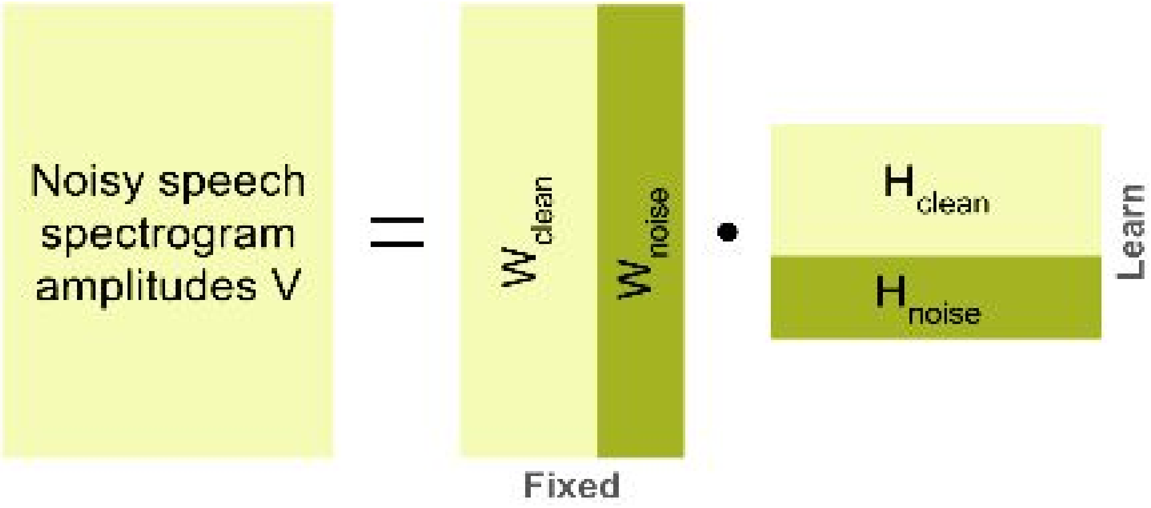
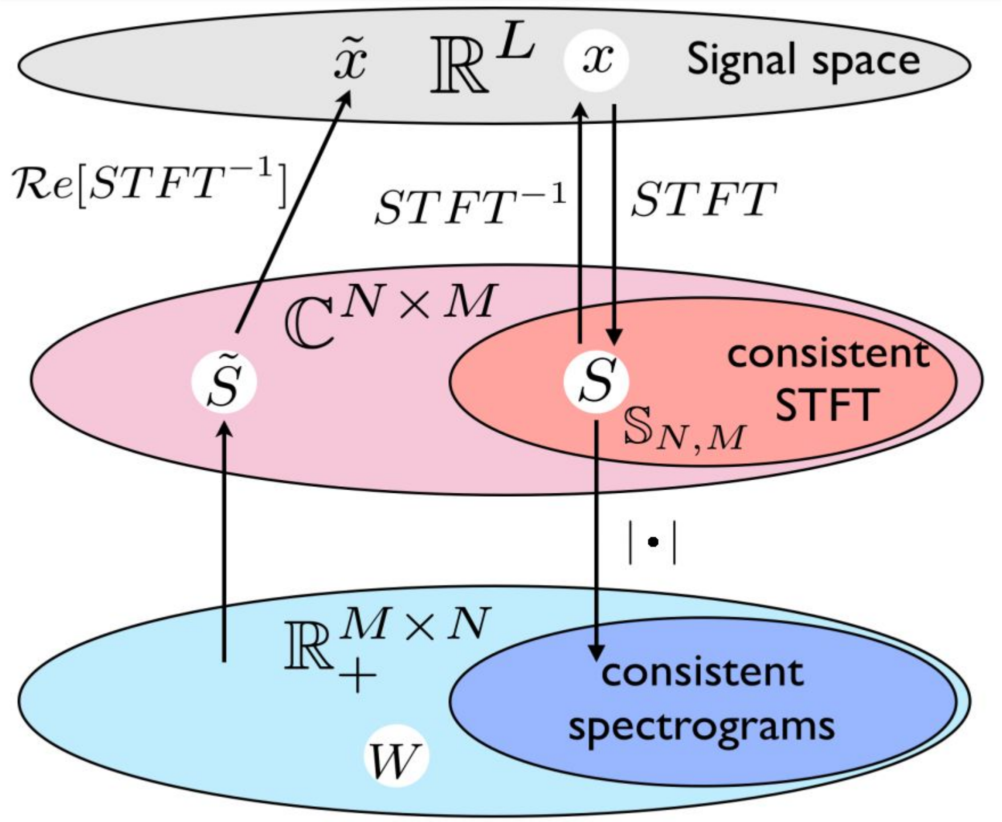


Figure : Denoising stage

4. The final step is to reconstruct the signal itself from the spectrogram. It is done by means of usual Inverse Short-Term Fourier Transformation.

## Signal Reconstruction



We reconstructed the signal using 4 techniques and compared the results obtained from them. The four methods used are :

- Naive method with zero phase :

$$\hat{X} = STFT^{-1}(\hat{S})$$

- Noisy Signal Phases :

$$\hat{X} = STFT^{-1}(\hat{S} \times \exp{(i\angle STFT(X_{noisy}))})$$

- Griffin & Lim iterative method :

$$\hat{X}_n = STFT^{-1}(\hat{S} \times \exp(i\angle STFT(\hat{X}_{n-1})))$$

- VAD (Voice Activity Detector) Method :

# Results

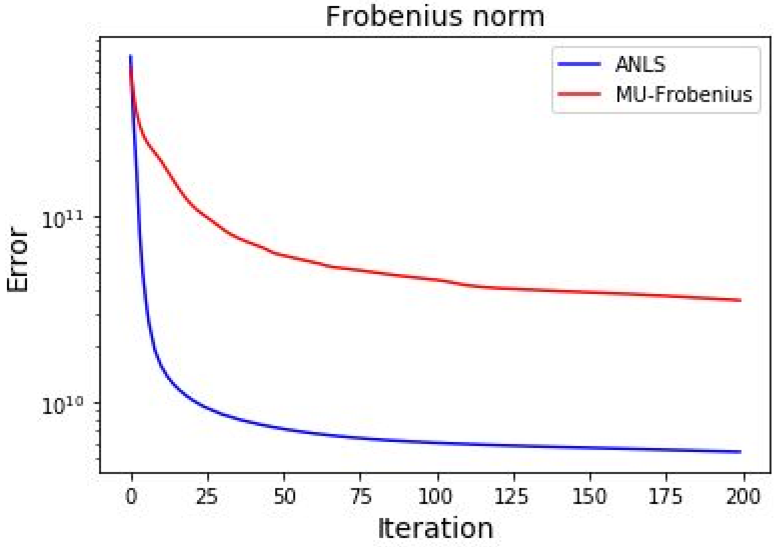
As a result of our work we implement the proposed workflow for the speech denoising. One of the unexpected difficulties was the part with the signal reconstruction from the magnitudes because we don’t think about it at the beginning of the project. But when we faced this problem we brainstormed the possible solutions, investigate the papers on this topic and come up with a few solutions

## NMF

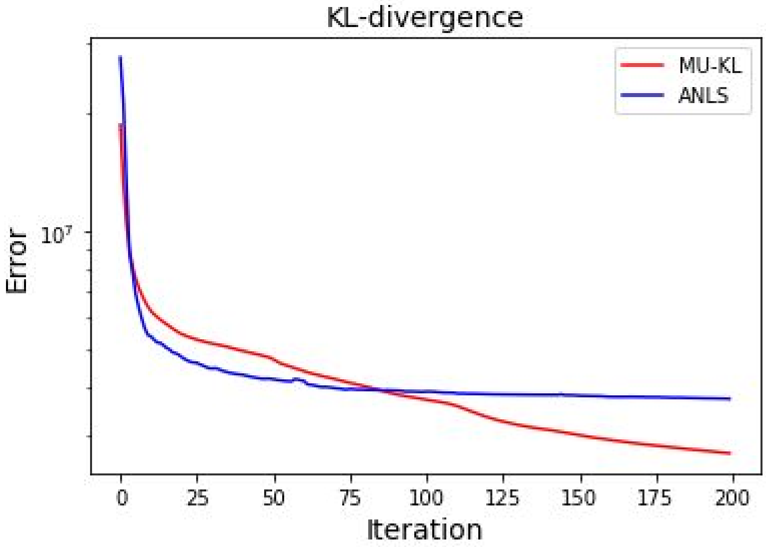
In the figure 3 one can observe the convergence of the two tested optimization methods.

In the figure ‘3a’ there are two algorithms compared in terms of Frobenius norm of difference of a matrix $\|V - W.H\|_F$ : ANLS with Frobenius norm, MU with Frobenius norm.

While in the figure 3b we compare the same ANLS algorithms (ANLS with Frobenius norm) but MU algorithm now minimizes the KL divergence.



3(a) Frobenius



3(b) KL divergence

Figure 3: Comparison of NMF optimization algorithms in different norms

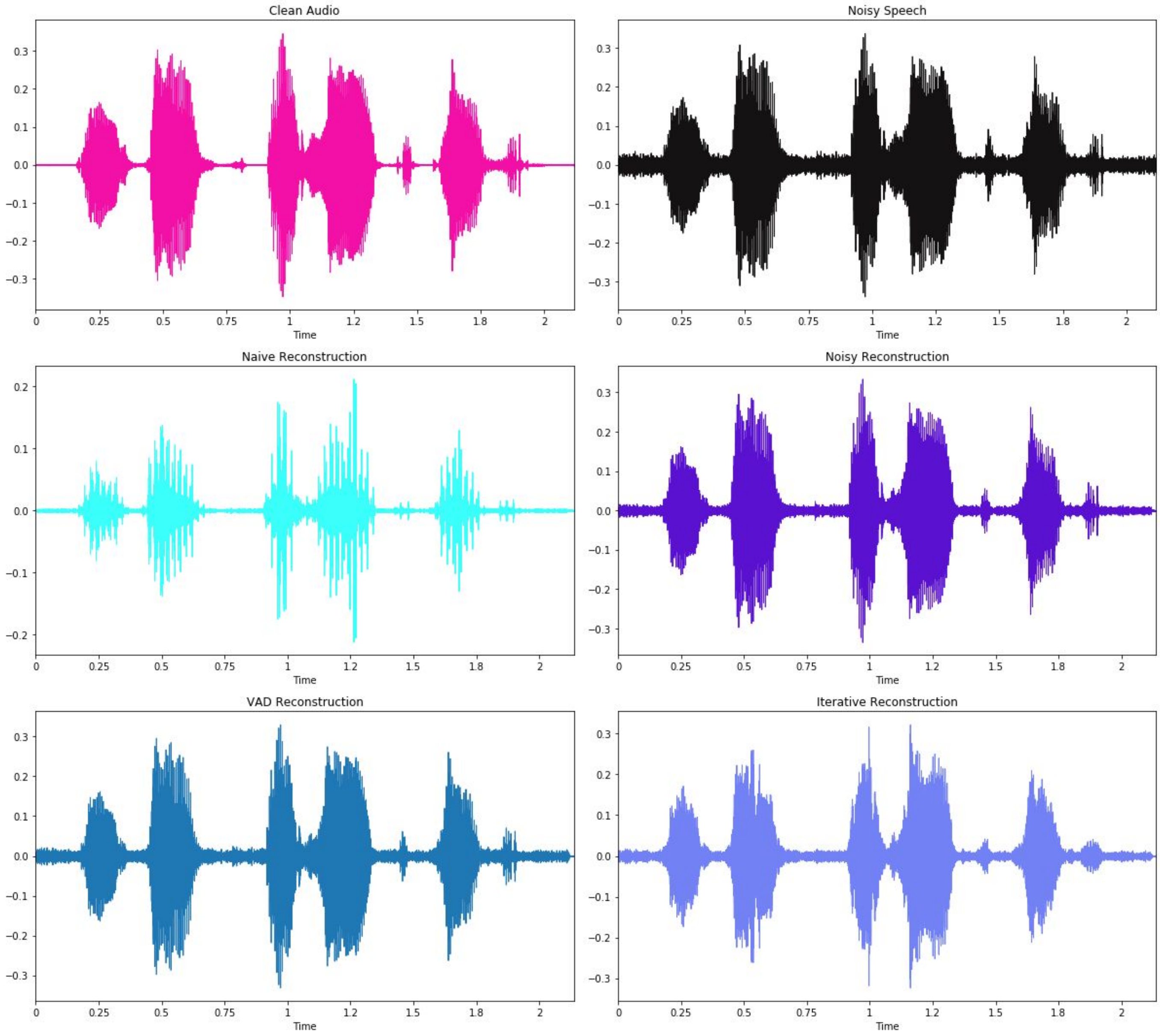
In terms of both KL divergence and Frobenius norm our implementation of ANLS shows good result. So, we decided to use ANLS in final version of the pipeline.

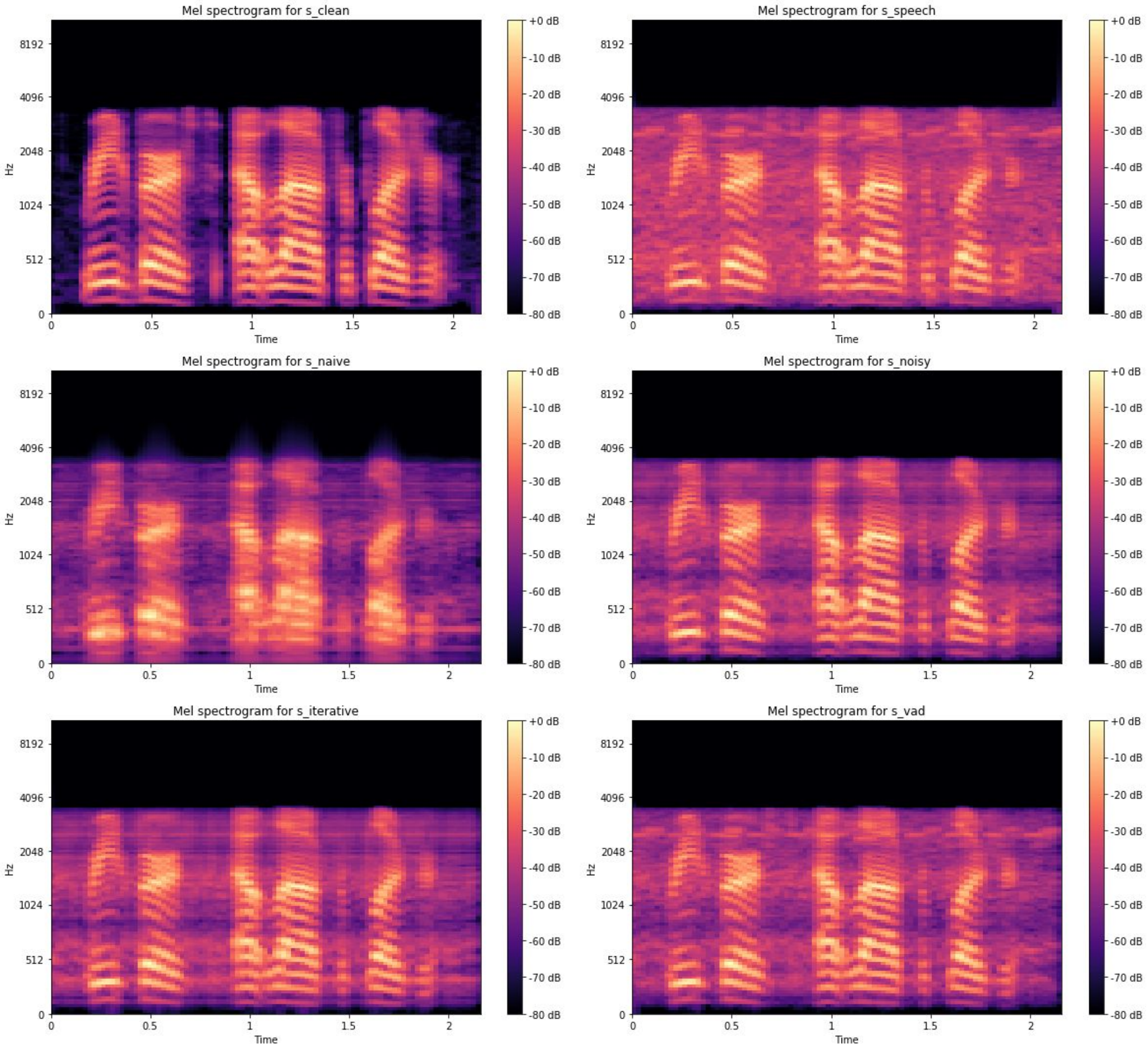
As for Quasi-Newton method, we implemented it, but didn't apply it to the magnitudes matrix V explicitly, because it needs a lot of resources (actually we ran out of memory with the spectrograms). Demonstration of the algorithm on smaller matrices can be found in separate notebook.

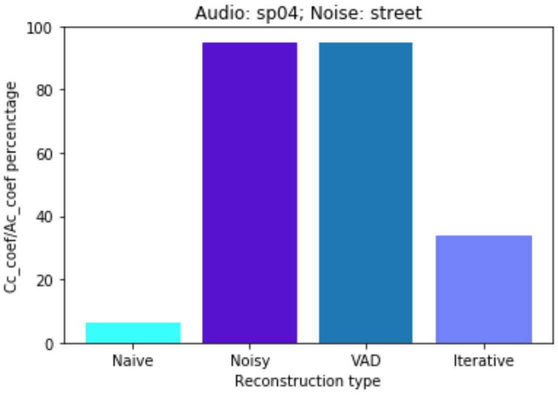
So as it was mentioned before, methods of the second order are not the best way to deal with the NMF, it is much better to use any gradient methods with multistart, moreover problem is multiextremal.

## Signal Reconstruction:

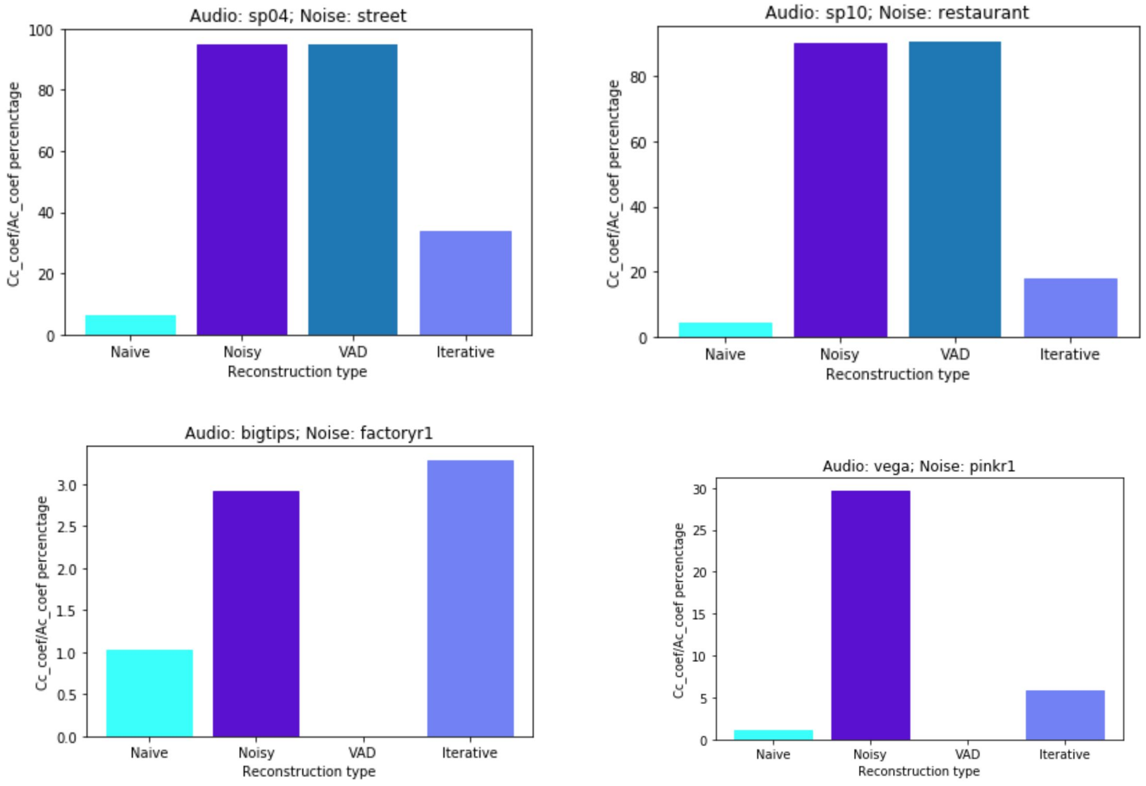
For same example:




Mel spectrogram for s_clean
Mel spectrogram for s_speech
Mel spectrogram for s_naive
Mel spectrogram for s_noisy
Mel spectrogram for s_iterative
Mel spectrogram for s_vad
Hz
8192
4096
2048
1024
512
0
Time
0
0.5
1
1.5
2
+0 dB
-10 dB
-20 dB
-30 dB
-40 dB
-50 dB
-60 dB
-70 dB
-80 dB


Audio: sp04; Noise: street
Cc_coef/Ac_coef percenctage
100
80
60
40
20
0
Naive
Noisy
VAD
Iterative
Reconstruction type

## Comparison of reconstruction techniques on different speech samples:



Audio: sp04; Noise: street

Audio: sp10; Noise: restaurant

Audio: bigtips; Noise: factoryr1

Audio: vega; Noise: pinkr1

# Future Scope

[10,12]During the work process and development of the initial idea we also found out that researchers try to use Convolutive NMF to deal with the denoising problem.

This idea sounds perspective from our point of view because of the two reasons:

(i) speech itself has a time-continual structure and applying convolutions here looks more than reasonable,

(ii) recently many deep learning approaches for speech recognition and generation (e.g. amazing Google Wavenet) rely on convolutions in their architectures and is able show state-of-the-art performance in virtue of that.

In [15] Speech enhancement experiments were done to examine the performance of the trained denoising DAE. Noise reduction, speech distortion, and perceptual evaluation of speech quality (PESQ) criteria are used in the performance evaluations. Experimental results show that adding depth of the DAE consistently increase the performance when a large training data set is given. In addition, compared with a minimum mean square error based speech enhancement algorithm, our proposed denoising DAE provided superior performance on the three objective evaluations.

So the next step of our research may be to embed CNMF into our framework.

Alternatively we can also work on new autoencoder pipelines as suggested in [16]. Where they investigated the use of convolutional autoencoders for audio denoising and found that large filter sizes and Tanh activations consistently achieved higher performance than small filter sizes and linear rectifiers across varying autoencoder architectures. Authors also presented empirical evidence in the form of qualitative comparisons as well as quantitative reconstruction error between the output of the autoencoder and spectrograms of clean audio tracks to demonstrate that convolutional autoencoders are a potentially promising approach for speech denoising.

# References

[1] Cauchi B., Goetze S., Doclo S. (2012). Reduction of Non-stationary Noise for a Robotic Living Assistant using Sparse Non-negative Matrix Factorization.

[2] Griffin D. W., Lim J.S. (1984). Signal Estimation from Modified Short-Time Fourier Transform. IEEE Transactions On Acoustics, Speech, And Signal Processing, Vol. ASSP-32, No. 2, April 1984

[3] Hu Y., Loizou P. C. (2008). Evaluation of Objective Quality Measures for Speech Enhancement IEEE Transactions On Audio, Speech, And Language Processing, Vol. 16, No. 1, January 2008

[4] Lyubimov N., Kotov M. (2013). Non-negative Matrix Factorization with Linear Constraints for Single-Channel Speech Enhancement.

[5] Nimfa, a Python library for nonnegative matrix factorization. http://nimfa.biolab.si/

[6] Smaragdis P. (2005) From Learning Music to Learning to Separate. Forum Acusticum Budapest 2005: 4th European Congress on Acoustics. (pp. 1545-1549)

[7] Sturmel N., Daudet L. (2011). Signal Reconstruction from STFT Magnitude: A State-of-the-Art. Proc. of the 14th Int. Conference on Digital Audio Effects (DAFx-11), Paris, France, September 19-23, 2011

[8] Wilson K. W., Raj B., Smaragdis P., Divakaran A. (2008). Speech denoising using nonnegative matrix factorization with priors.

[9] Zdunek R., Cichocki A. (2006). Non-Negative Matrix Factorization with Quasi-Newton Optimization. Eighth International Conference on Artificial Intelligence and Soft Computing, ICAISC, pages 870–879

[10] Vaz C., Dimitriadis D., Thomas S., Narayanan S. (2016) CNMF-based Acoustic Features for Noise-Robust ASR. Proceedings of IEEE International Conference on Audio, Speech and Signal Processing (ICASSP), Shanghai, China, 2016.

[11] Riabenko E. A. (2014). Loss function choice in problem of non-negative matrix factorization.

[12] O'Grady P.D., Pearlmutter B.A. (2006) Convolutive non-negative matrix factorisation with a sparseness constraint. In Proc. of the 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing, pages 427–432.

[13] Ephraim Y., Malah D. (1984) Speech enhancement using a minimum mean square error short-time spectral amplitude estimator. IEEE Transactions on Acoustic, Speech and Signal Processing, 32(6):1109–1121, 1984

[14] Boll S. (1979) Suppression of acoustic noise in speech using spectral subtraction. IEEE Transactions on Acoustics Speech and Signal Processing, 27(2):113–120, 1979

[15] Speech Enhancement Based on Deep Denoising Autoencoder; Xugang Lu, Yu Tsao, Shigeki Matsuda, Chiori Hori; INTERSPEECH2013

[16] Denoising Convolutional Autoencoders for Noisy Speech Recognition; Mike Kayser, Victor Zhong, Stanford University; CS231n Project Report

[17] Lee and Seung; Learning the parts of objects by non-negative matrix factorization, 1999.