Signal Decomposition Methods with Application to Energy Disaggregation

Alireza Rahimpour

Supervisor: Dr. Hairong Qi

University of Tennessee
Department of Electrical Engineering and Computer Science

June 23, 2015
Signal decomposition

Separation of a set of source signals from a set of mixed signals.

\[ V = WH \]  

(1)

- \( V \in R^{m \times n} \): Observation matrix
- \( W \in R^{m \times r} \): Basis matrix, dictionary or source matrix
- \( H \in R^{r \times n} \): Activation matrix

- \( m \): dimension of the observation
- \( r \): number of sources
- \( n \): number of observations
One Example

Cocktail party

What is source separation?
What is source separation?

Magic?

\[ V = WH \]

In fact, without some **prior knowledge**, it is not possible to uniquely estimate the original source signals.

Need to impose **constraints** on the problem.
Three main families of signal decomposition methods

1. Non-negative Matrix Factorization (NMF):
   Both sources and mixtures are non-negative.

2. Sparse Component Analysis (SCA):
   Most of the entries of activation matrix are zero.

3. Independent Component Analysis (ICA):
   Sources are statistically independent and non-Gaussian.
Supervised vs. Unsupervised methods

\[ V = WH \]

\( V \): Observation matrix
\( W \): Basis matrix or dictionary
\( H \): Activation matrix

**Unsupervised:**

Only observation matrix \( V \) is available and \( W \) and \( H \) are both unknown: Blind source separation.

**Supervised:**

Observation matrix \( V \) and basis matrix \( W \) are known and goal is to estimate the activation matrix \( H \).
History of Signal decomposition methods

- Blind source separation (BSS) problem formulated around 1982, by Hans, et al. for a biomedical problem: first papers in the mid of the 80’s.
- Great interest from the community, mainly in France and later in Europe and in Japan, and then in the USA.
  - In June 2009, 22000 scientific papers are recorded by Google Scholar (Comon and Jutten, 2010)
  - People with different backgrounds: signal processing, statistics and machine learning.
- Until the end of the 90’s, BSS = ICA
- First NMF methods in the mid of the 90’s but famous contribution in 1999.
- First SCA approaches around 2000 but massive interest since.
History of Signal decomposition methods

- Blind source separation (BSS) problem formulated around 1982, by Hans, et al. for a biomedical problem: first papers in the mid of the 80’s.
- Great interest from the community, mainly in France and later in Europe and in Japan, and then in the USA.
  - In June 2009, 22000 scientific papers are recorded by Google Scholar (Comon and Jutten, 2010)
  - People with different backgrounds: signal processing, statistics and machine learning.
- Until the end of the 90’s, BSS=ICA
- First NMF methods in the mid of the 90’s but famous contribution in 1999.
- First SCA approaches around 2000 but massive interest since.
Non-negative Matrix Factorization (NMF)
Non-negative Matrix Factorization (NMF)

A matrix factorization where everything is non-negative.

\[ V \in R_{+}^{m \times n}, \quad W \in R_{+}^{m \times r}, \quad H \in R_{+}^{r \times n} \]

\[
\begin{bmatrix}
  V
\end{bmatrix} \approx \begin{bmatrix}
  W
\end{bmatrix} \begin{bmatrix}
  H
\end{bmatrix}
\]
Why Non-negativity?

Data is often non-negative by nature:

- Pixel intensities
- Amplitude spectra
- Energy consumption
- Occurrence counts
- User scores
- Stock market values
- ..., etc.
Why Non-negativity? cont.

NMF allows **only additive** but not subtractive combinations during the factorization:

- Result in **part-based representation** of the data.
- Such nature can discover the hidden components that have specific structures and physical meanings.
Part-based representation of NMF

\[ V = W \cdot H = [W_1, W_2] \cdot [H_1; H_2] \]

\( V \) is approximated as sum of matrix “layers”
History of research on NMF:

NMF is more than 30-year old!

- **First work:**
  - Positive Matrix Factorization-PMF (Paatero and Tapper, 1994)

- Popularized by *Lee and Seung* (1999) for *learning the parts of objects* *(Nature journal).*

- Since then, widely used in various research areas for diverse applications.
History of research on NMF:

NMF is more than 30-year old!

- First work:
  - Positive Matrix Factorization-PMF (Paatero and Tapper, 1994)

- Popularized by Lee and Seung (1999) for learning the parts of objects (Nature journal).

- Since then, widely used in various research areas for diverse applications.
Solving NMF

How do we solve for $W$ and $H$, given a known $V$?

Answer: Frame as optimization problem:

\[
\begin{align*}
\text{Minimize } & D(V || WH) \\
\text{subject to } & W \geq 0, H \geq 0
\end{align*}
\]

- **Two measures of divergence** proposed by (Lee and Seung, 1999) are:
  - Euclidean (EU): \( D_{EU}(V || WH) = \sum_{ij} (V_{ij} - (WH)_{ij})^2 \)
  - Kullback-Leibler (KL):
    \[
    D_{KL}(V || WH) = \sum_{ij} \left( V_{ij} \frac{\log V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij} \right)
    \]
How do we solve for W and H, given a known V?

Answer: Frame as optimization problem:

\[
\begin{align*}
\text{Minimize } & D(V \| WH) \\
\text{subject to } & W \geq 0, H \geq 0
\end{align*}
\]

- **Two measure of divergence** proposed by (Lee and Seung, 1999) are:
  - Euclidean (EU): \( D_{EU}(V \| WH) = \sum_{ij} (V_{ij} - (WH)_{ij})^2 \)
  - Kullback-Leibler (KL):
    \[
    D_{KL}(V \| WH) = \sum_{ij} (V_{ij} \frac{\log V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij})
    \]
And many other divergence measures

- Itakura-Saito (IS) divergence (Fvotte et al., 2009)
- $\beta$- divergence (Eguchi and Kano., 2001)
- $\alpha$-divergence (Cichocki et al., 2006, 2008)
- Bregman divergence (generalizes $\beta$-divergence) (Bregman, 1967; I. S. Dhillon and S. Sra, 2005)
- Rnyis divergence (Devarajan and Ebrahimi, 2005)
- $\gamma$-divergence (Fujisawa and Eguchi, 2008)
- etc.

NMF divergence choice depends on the data and on the application.
There are several ways in which the $W$ and $H$ may be found:

- Lee and Seung’s multiplicative update rule (Lee and Seung, 1999) has been a popular method.

- Some successful algorithms are based on Alternating Non-negative Least Squares (ANLS) (Berry et al., 2006):
  
  - In each step of algorithm, first $H$ is fixed and $W$ found by a non-negative least squares solver, then $W$ is fixed and $H$ is found analogously.

- And many other...
Initialization

A good initialization of parameters (W and H) is important due to the existence of local minima.

- Random initializations:
  - initialize (nonnegative) parameters randomly several times;
  - keep the solution with the lowest final cost.

- Structural data-driven initializations:
  - initialize W by clustering of data points V (Kim and Choi, 2007);
  - initialize W by singular value decomposition (SVD) of data points V (Boutsidis and Gallopoulos, 2008);
  - etc ...
A good initialization of parameters (W and H) is important due to the existence of local minima.

- Random initializations:
  - initialize (nonnegative) parameters randomly several times;
  - keep the solution with the lowest final cost.

- Structural data-driven initializations:
  - initialize W by clustering of data points V (Kim and Choi, 2007);
  - initialize W by singular value decomposition (SVD) of data points V (Boutsidis and Gallopoulos, 2008);
  - etc ...
Adding constraints to NMF

Problems:
- NMF solution is not unique.
- Hence NMF is not guaranteed to extract latent components as desired within a particular application.

Possible solution:
Given the application, impose some knowledge-based constraints on:
W, H, or on both W and H.

- Appropriate constraints may lead to more suitable latent components.
Adding constraints to NMF

Problems:
- NMF solution is not unique.
- Hence NMF is not guaranteed to extract latent components as desired within a particular application.

Possible solution:
Given the application, impose some knowledge-based constraints on:
W, H, or on both W and H.

- Appropriate constrains may lead to more suitable latent components.
Adding constraints to NMF

Different types of constraints have been considered in previous works:

- **Sparsity constraints**: either on $W$ or $H$ (e.g., Hoyer, 2004; Eggert and Korner, 2004);

Enforcing only few non-zero entries in $h_n$:

![Unconstrained H vs Sparse H](image)
Adding constraints to NMF, Cont.

- Group sparsity-inducing penalties in NMF.
  (A. Lefvre et al., 2011; Sun and Mazumder, 2013; El Badawy et al., 2014)

Enforcing only few non-zero pre-defined groups (blocks) in $H$:
And many more possible constraints

- Smoothness on activation matrix (Virtanen, 2007; Jia and Qian, 2009; Essid and Fevotte, 2013);
- Cross-modal correspondence constraints (Seichepine et al., 2013; Liu et al., 2013);
- Minimum volume NMF (Miao and Qi, 2007);
- Orthogonal NMF (Choi, 2008);
- etc.
Supervised NMF

Assumption:

One has access to example signals at the training step

1. NMF uses these training data to train a set of basis vectors for each source \((i = 1, \ldots, k)\).

\[ V_i^{\text{train}} = W_i^{\text{train}} H_i^{\text{train}} \]

2. The basis matrix (trained bases) \(W_i^{\text{train}}\) is used as a representative model for the training data for each source \(i\).

3. In the testing or separation stage, the trained basis matrices for all sources are concatenated in one bases matrix:

\[ W = [W_1, \ldots, W_k] \]
Supervised NMF

Assumption:
One has access to example signals at the training step

1. NMF uses these training data to train a set of basis vectors for each source \((i = 1, \ldots, k)\).

\[ V_{i}^{\text{train}} = W_{i}^{\text{train}} H_{i}^{\text{train}} \]

2. The basis matrix (trained bases) \(W_{i}^{\text{train}}\) is used as a representative model for the training data for each source \(i\).

3. In the testing or separation stage, the trained basis matrices for all sources are concatenated in one bases matrix:

\[ W = [W_1, \ldots, W_k] \]
Supervised NMF, Cont.

- In the testing stage, for a new test observation $V$ the bases matrix $W$ is kept fixed and only the weights matrix $H$ is updated: 
  $$\hat{H} = \arg\min_{H \geq 0} ||V_{test} - WH||_2$$

- The estimated signal for each source is therefore found by multiplying each source basis in the bases matrix with its corresponding weights in the weights matrix:
  $$\hat{V}_i = W_i \hat{H}_i$$
In the testing stage, for a new test observation $V$ the bases matrix $W$ is kept fixed and only the weights matrix $H$ is updated: 
$$\hat{H} = \arg\min_{H \geq 0} ||V_{test} - WH||_2$$

The estimated signal for each source is therefore found by multiplying each source basis in the bases matrix with its corresponding weights in the weights matrix:
$$\hat{V}_i = W_i \hat{H}_i$$
Applications

- Computer vision and Image analysis:
  - Feature representation, clustering (Lee et al., 99)
  - Action recognition (Masurelle et al., 2014)
  - Denoising and inpainting (Mairal et al., 2010)
  - Face recognition (Soukup and Bajla, 2008)
  - Tagging (Kalayeh et al., 2014)
Other applications

- Acoustic signal processing:
  - Speech recognition (Cichocki et al., 04)
  - Signal enhancement/denoising: (Sun and Mazumder, 2013)
  - Compression (Ozerov et al., 2011b; Nikunen et al., 2011)

- Medical Imaging: CT image reconstruction (Damon et al., 2013)

- Text mining:
  - Document clustering (Xu et al., 03; Shahnaz et al., 06)
  - Topic detection and trend tracking (Berry et al., 05)

- Social network: (Chi et al., 07; Wang et al., 08)

- Recommendation system:
  - The Netflix Prize: predict user ratings for films (Zhang et al., 06)
Other applications

- Acoustic signal processing:
  - Speech recognition (Cichocki et al., 04)
  - Signal enhancement/denoising: (Sun and Mazumder, 2013)
  - Compression (Ozerov et al., 2011b; Nikunen et al., 2011)

- Medical Imaging: CT image reconstruction (Damon et al., 2013)

- Text mining:
  - Document clustering (Xu et al., 03; Shahnaz et al., 06)
  - Topic detection and trend tracking (Berry et al., 05)

- Social network: (Chi et al., 07; Wang et al., 08)

- Recommendation system:
  - The Netflix Prize: predict user ratings for films (Zhang et al., 06)
Sparse Component Analysis (SCA)
The Sparse Decomposition Problem:

Let \( \mathbf{x} \in \mathbb{R}^m \) be a signal.

Let \( \mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_p] \in \mathbb{R}^{m \times p} \) be a set of normalized "basis vectors". We call it dictionary.

\( \mathbf{D} \) is "adapted" to \( \mathbf{x} \) if it can represent it with a few basis vectors—that is, there exists a sparse vector \( \mathbf{\alpha} \in \mathbb{R}^p \) such that \( \mathbf{x} \approx \mathbf{D} \mathbf{\alpha} \). We call \( \mathbf{\alpha} \) the sparse code.
Sparse Decomposition problem

\[
\min_{\alpha \in \mathbb{R}^p} \frac{1}{2} \| x - D\alpha \|^2_2 + \lambda \psi(\alpha)
\]

\(\psi\) induces sparsity in \(\alpha\). It can be

- the \(l_0\) “pseudo-norm”. \(\| \alpha \|_0 \triangleq \#\{i \text{ s.t. } \alpha[i] \neq 0\}\) (NP-hard)
- the \(l_1\) norm. \(\| \alpha \|_1 \triangleq \sum_{i=1}^{p} |\alpha[i]|\) (convex)
  
  \[
  \text{L}_p \text{ norm: } \| \alpha \|_p = \left(\sum_i |\alpha_i|^p\right)^{1/p}
  \]

1. Slide from (Marial, Sapiro 2010)
Sparse Decomposition problem

\[ \min_{\alpha \in \mathbb{R}^p} \frac{1}{2} \left\| x - D\alpha \right\|_2^2 + \lambda \psi(\alpha) \]

\psi\) induces sparsity in \(\alpha\). It can be

- the \(\ell_0\) “pseudo-norm”. \(\|\alpha\|_0 \triangleq \# \{ i \text{ s.t. } \alpha[i] \neq 0 \}\) (NP-hard)
- the \(\ell_1\) norm. \(\|\alpha\|_1 \triangleq \sum_{i=1}^{p} |\alpha[i]|\) (convex)
- ...

\[ L_p \text{ norm: } \|\alpha\|_p = \left( \sum_i |\alpha_i|^p \right)^{1/p} \]

\(^1\) Slide from (Marial, Sapiro 2010)
In general the $l_p$ norm is a popular measure of sparsity in the mathematical analysis community. For measuring sparsity, $0 < p < 1$ is of the most interest but, unfortunately, it leads to a non-convex optimization problem.

- We usually choose $p = 1$ for obtaining the sparse solution.
Approaches

- **Relaxation methods:**
  Smooth the $L_0$, e.g., basic pursuit (BP, also referred to as LASSO), FOCUSS. (Chen, Donoho & Saunders: 1995)

- **Greedy algorithm:**
  Build the solution one non-zero element at a time, e.g., matching pursuit (MP), OMP (Mallat & Zhang: 2003)
How to choose the Dictionary D?

- Choose D from a known set of transforms (e.g., wavelets, DCT, etc.)
  But, this is not optimal.

- Training (Learn from examples):
  Several dictionary learning approaches:
How to choose the Dictionary D?

- Choose D from a known set of transforms (e.g., wavelets, DCT, etc.)
  But, this is not optimal.

- Training (Learn from examples):
  Several dictionary learning approaches:
  - K-SVD: (M. Aharon, M. Elad, 2006)
    - Alternates between sparse coding of the examples based on the current dictionary and an update process for the dictionary atoms so as to better fit the data.
How to choose the Dictionary D?

- Choose D from a known set of transforms (e.g., wavelets, DCT, etc.)
  But, this is not optimal.

- Training (Learn from examples):
  Several dictionary learning approaches:

  - K-SVD: (M. Aharon, M. Elad, 2006)
    - Alternates between sparse coding of the examples based on the current dictionary and an update process for the dictionary atoms so as to better fit the data.
Sparsity

Many natural signals are sparse or compressible in the sense that they have concise representation when expressed in the proper basis.

Transform sparsity of MR images.

The largest coefficients are preserved while all others set to zero.

- Transform-based compression: JPEG, JPEG-2000, and MPEG.
Application: Compressed sensing

Replace samples with few linear projections: \( y = \phi x \)

\[
\begin{align*}
M \times 1 & \quad \Phi & \quad N \times 1 \\
\text{measurements} & & \text{sparse signal} \\
& \quad M \times N & \quad K \\
\end{align*}
\]

\( K < M \ll N \)

\( \phi = \) Measurement or sensing basis
\( Y = \) Our measurement
\( X = \) Original signal
Why Compressed sensing?

Reducing the number of measurements

- The number of sensors may be limited.
- The measurements may be extremely expensive: neutron scattering imaging.
- The sensing process may be slow: MRI.
Independent Component Analysis (ICA)
Independent Component Analysis

This method is the base of blind source separation.

\[ X = AS \]

- All we observe is \( X \) and we must estimate both \( A \) and \( S \) using it.

Assumption:

1. The sources being considered are statistically independent.
2. The independent components have non-Gaussian distribution.
ICA preprocessing

Preprocessing

Preprocessing steps in order to simplify and reduce the complexity of the problem:
- Centering
- Whitening
- Dimensionality reduction

Whitening and dimension reduction can be achieved with:
- Principal Component Analysis (PCA)
- Eigenvalue Decomposition (EVD)
ICA finds the independent components by:

Maximizing the statistical independence of the estimated components.

Many ways to define independence, and this choice governs the form of the ICA algorithm.

Minimization of mutual information (Comon, 1994)
- Kullback-Leibler Divergence.
- Maximum entropy.

Maximization of non-Gaussianity (Bell and Sejnowski, 1995)
Non-Gaussianity measures:
- Kurtosis: \( \text{kurt}(x) = 0 \) if \( x \) Gaussian
- Negentropy: \( (= 0 \) when Gaussian)
ICA finds the independent components by:

Maximizing the statistical independence of the estimated components.

Many ways to define independence, and this choice governs the form of the ICA algorithm.

Minimization of mutual information (Comon, 1994)
- Kullback-Leibler Divergence.
- Maximum entropy.

Maximization of non-Gaussianity (Bell and Sejnowski, 1995)
Non-Gaussianity measures:
- Kurtosis: \( \text{kurt}(x) = 0 \) if \( x \) Gaussian
- Negentropy: \( (= 0 \) when Gaussian)
Application

- Audio signal processing: (Cichocki & Amari 2002);
- Image processing: (Cichocki & Amari 2002);
- Data mining: (Lee et al. 2009);
- Biomedical signal processing: (Azzerboni et al. 2004);
- And many more.
Application: Signal decomposition for Load disaggregation
Signal decomposition for Load disaggregation

Load disaggregation
The task of taking a whole-home energy signal from a single monitoring point and separating it into its component appliances.

Also called: Non-Intrusive Load Monitoring (NILM)
Non-Intrusive: without use of additional measuring equipment inside the consumer’s house.
### Why does it matter?

#### Customers
- Understand their bill and plan their monthly budget.
- Identify/repair/replace energy hogs.
- Be able to make a financial decision for when to use an appliance.

#### Utilities
- Identify/verify appliances that could participate in Demand Response.
- Improve relationships with customers.
- Understand customer behavior to improve capacity planning.
How much energy can we really save?

ACEEE analyzed results from 36 different studies between 1995-2010 and found energy savings varied based on the data given to customers:

![Energy Savings Chart]

- **3.8%** Estimated Feedback (Web-based energy audits with info on ongoing basis)
- **6.8%** Enhanced Billing (Household-specific info, advice)
- **8.4%** Daily/Weekly Feedback (Household-specific info, advise on daily or weekly basis)
- **9.2%** Real-Time Feedback (Real-time premise level info)
- **12.0%** Real-Time Plus Feedback (Real-time info down to the appliance level)
Goal?

Having device level energy information can cause users to conserve significant amounts of energy.

But, current electricity meters only report whole-home data. So, Developing algorithmic methods for disaggregation presents a key point in energy conservation.
Goal?

Having device level energy information can cause users to conserve significant amounts of energy.

![Breakdown of Energy Consumption By Appliance](image)

But, current electricity meters only report whole-home data. So, Developing algorithmic methods for disaggregation presents a key point in energy conservation.
Commercial Load monitoring device:

Plugwise:

https://www.plugwise.com/home-stretch-start#
Related works

Academic work on energy disaggregation began with the work of George W. Hart, at MIT in the early 1980s.

Methods based on Classification:
These approaches look for some features in power signals, and cluster devices according to these features.
1-signal analysis and feature extraction.
2-model learning and classification.

Methods based on decomposition:
Sparse coding (Kolter, Andrew Y. Ng 2010, MIT, Stanford)
In general, algorithms for energy disaggregation have received growing interest in recent years. (Kolter, Batra, and Ng 2010; Kim et al. 2011; Ziefman and Roth 2011)

**Another challenge:**
Existing energy disaggregation approaches virtually all use high-frequency sampling (e.g. per second or faster)

But, **smart meters** (communication-enabled power meters): limited to recording whole home energy usage at low frequencies.

-Currently installed in more than 32 million homes (Institute for Electric Efficiency 2012)
In general, algorithms for energy disaggregation have received growing interest in recent years. (Kolter, Batra, and Ng 2010; Kim et al. 2011; Ziefman and Roth 2011)

Another challenge:
Existing energy disaggregation approaches virtually all use high-frequency sampling (e.g. per second or faster)

But, **smart meters** (communication-enabled power meters): limited to recording whole home energy usage at low frequencies.

- Currently installed in more than 32 million homes (Institute for Electric Efficiency 2012)
NMF for Load disaggregation

\[ \tilde{X} \in \mathbb{R}^{m \times d} : \text{Aggregated energy consumption for all devices in a given house.} \]

\[ D_i \in \mathbb{R}^{m \times n} : \text{The energy consumption of the } i\text{th appliance.} \]

Each column of \( D_i \): One day of power consumption.

d: Number of testing days (unforeseen)
n: Number of training days
m: Number of samples in each column (i.e., each day)
k= number of all the devices at home.
NMF for Load disaggregation

\( \tilde{X} \in R^{m \times d} \): Aggregated energy consumption for all devices in a given house.

\( D_i \in R^{m \times n} \): The energy consumption of the \( ith \) appliance. Each column of \( D_i \): One day of power consumption.

d: Number of testing days (unforeseen)
n: Number of training days
m: Number of samples in each column (i.e., each day)
\( k = \) number of all the devices at home.

**Dictionary \( D \):**

Concatenation of all the basis vectors for all the devices.

\( D \in R^{m \times T} \) (\( D = [D_1, D_2, \ldots, D_k] \) and \( T = n \times k \)).
**NMF for Load disaggregation**

\( \tilde{X} \in R^{m \times d} \): Aggregated energy consumption for all devices in a given house.

\( D_i \in R^{m \times n} \): The energy consumption of the \( i \)th appliance. Each column of \( D_i \): One day of power consumption.

d: Number of testing days (unforeseen)
n: Number of training days
m: Number of samples in each column (i.e., each day)
k= number of all the devices at home.

**Dictionary \( D \):**

Concatenation of all the basis vectors for all the devices.

\( D \in R^{m \times T} \) (\( D = [D_1, D_2, \ldots D_k] \) and \( T = n \times k \)).
Introduction
Non-negative Matrix Factorization (NMF)
Sparse Component Analysis (SCA)
Independent Component Analysis (ICA)

Application: Signal decomposition for Load disaggregation

Conclusion

What is Load disaggregation?
Related works
Problem formulation

\[ \tilde{X} = DA \]

\[ \hat{A}_{1:k} = \arg\min_{A_{1:k} \geq 0} \left\| \tilde{X} - [D_1 \ldots D_k] \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix} \right\|_2^2 \]

\( A_i (i = 1, \ldots, k) \): the activation matrix for the \( i \)th device’s base matrix \( (D_i) \).

After calculating the activation coefficient for each device \( (\hat{A}_i) \), the estimated signal for the \( i \)th device would be:

\[ \hat{X}_i = D_i \hat{A}_i \]

\( \hat{X}_i \in R^{m \times d} \), \( D_i \in R^{m \times n} \) and \( \hat{A}_i \in R^{n \times d} \)
$\tilde{X} = DA$

$\hat{A}_{1:k} = \arg\min_{A_{1:k} \geq 0} \| \tilde{X} - [D_1 \ldots D_k] \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_k \end{bmatrix} \|_2^2$

$A_i (i = 1, \ldots, k):$ the activation matrix for the $i$th device’s base matrix ($D_i$).

After calculating the activation coefficient for each device ($\hat{A}_i$), the estimated signal for the $i$th device would be:

$\hat{X}_i = D_i \hat{A}_i$

$\hat{X}_i \in \mathbb{R}^{m \times d}, D_i \in \mathbb{R}^{m \times n}$ and $\hat{A}_i \in \mathbb{R}^{n \times d}$
Training stage:
Decomposition of the aggregated signal to each device’s power consumption at the same days as the training set (n).

Testing stage:
Estimating the power consumption of each device by having access to the aggregated signal in a totally different random days (d) than the days in training set (n).

Using 80% of the data as training set (as basis in dictionary) and 20% of the data for the testing set (aggregated signal).
NMF for LD, Cont.

Training stage:
Decomposition of the aggregated signal to each device’s power consumption at the same days as the training set (n).

Testing stage:
Estimating the power consumption of each device by having access to the aggregated signal in a totally different random days (d) than the days in training set (n).

Using 80% of the data as training set (as basis in dictionary) and 20% of the data for the testing set (aggregated signal).

Constrained NMF
What additional constraints can we define for this problem?
NMF for LD, Cont.

Training stage:
Decomposition of the aggregated signal to each device’s power consumption at the same days as the training set (n).

Testing stage:
Estimating the power consumption of each device by having access to the aggregated signal in a totally different random days (d) than the days in training set (n).

Using 80% of the data as training set (as basis in dictionary) and 20% of the data for the testing set (aggregated signal).

Constrained NMF
What additional constraints can we define for this problem?
Conclusion
Summary:

- We reviewed different works on signal decomposition based on NMF, ICA, SCA.
- Choosing the right method highly depends on the structure of data and intended application.
- Adding prior knowledge to the problem as a constraint.
- NMF is effective and has been gaining increasing popularity in the last few years: significant number of publications on different topics.
- Application of NMF method on Energy disaggregation.
- Future:
  - How to handle big data? e.g., Web-scale data mining.
  - Online signal decomposition: how to update the factorization when new data comes in without recomputing from scratch.
Available softwares for different signal decomposition algorithms:

- **SPAMS PACKAGE:**
  http://spams-devel.gforge.inria.fr/downloads.html
  SPArse Modeling Software is an optimization toolbox for solving various sparse estimation problems. Dictionary learning and matrix factorization (NMF, sparse PCA, ...)

- **libNMF:** A Library for Nonnegative Matrix Factorization.
  http://www.univie.ac.at/rlcta/software/

- **Beta-NTF:** Nonnegative Matrix and Tensor Factorization in Python.
  https://code.google.com/p/beta-ntf/

- **Sklearn.decomposition.NMF:** Non-Negative matrix factorization.

- **The NMF:DTU Toolbox-MATLAB.**

- **The FastICA package for MATLAB.**
  http://research.ics.aalto.fi/ica/fastica/

- **Sparsity-based Blind Source Separation (BSS) method:**
  http://www.ast.obspm.fr/article715.html

- Many softwares available in ICA Central, ICALab.
References


Thanks for your attention!

Questions/Comments?