A Sparse Dictionary Learning Framework to Discover Discriminative Source Activations in EEG Brain Mapping

Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Presented by :
Divy Sitlani
Krishna Dwypayan Kota
Harpreet Singh Arora
Aditya Neel

Introduction

EEG(Electroencephalography) is one of the most important non-invasive human brain imaging tools.

The aim is to discover essential activated brain sources associated with different brain status.

Can be formulated and solved as a sparse overcomplete dictionary learning problem.

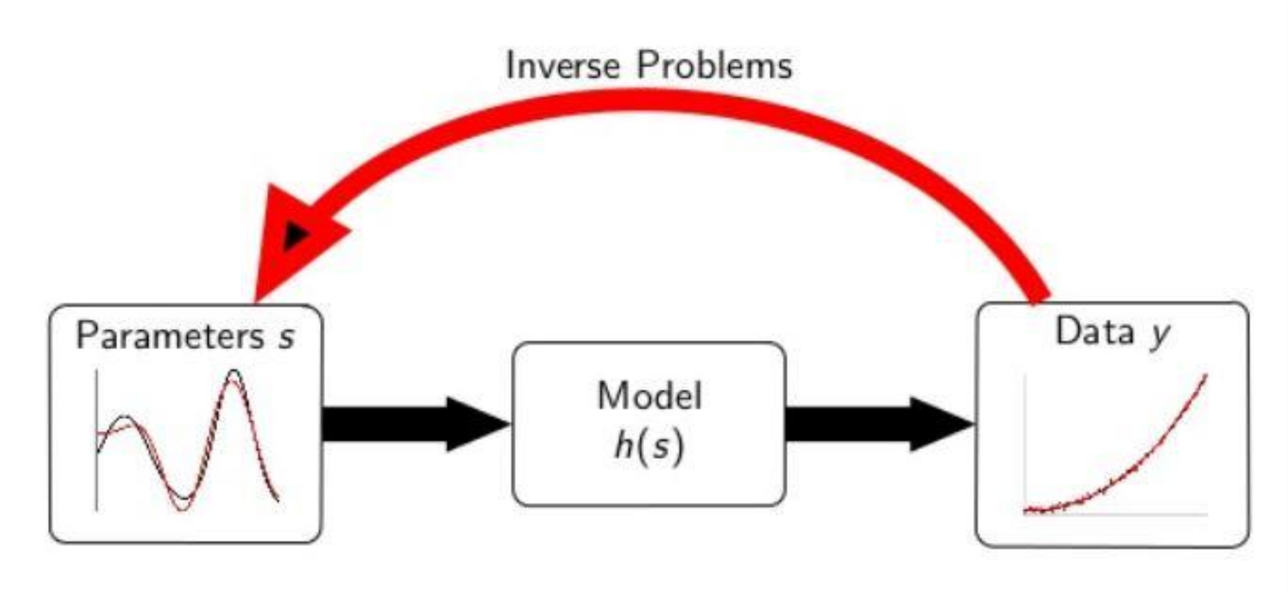
Problem Statement

Calculate the discriminative sources to facilitate the understanding of brain mechanism under different cognitive tasks or different neurological disorders.

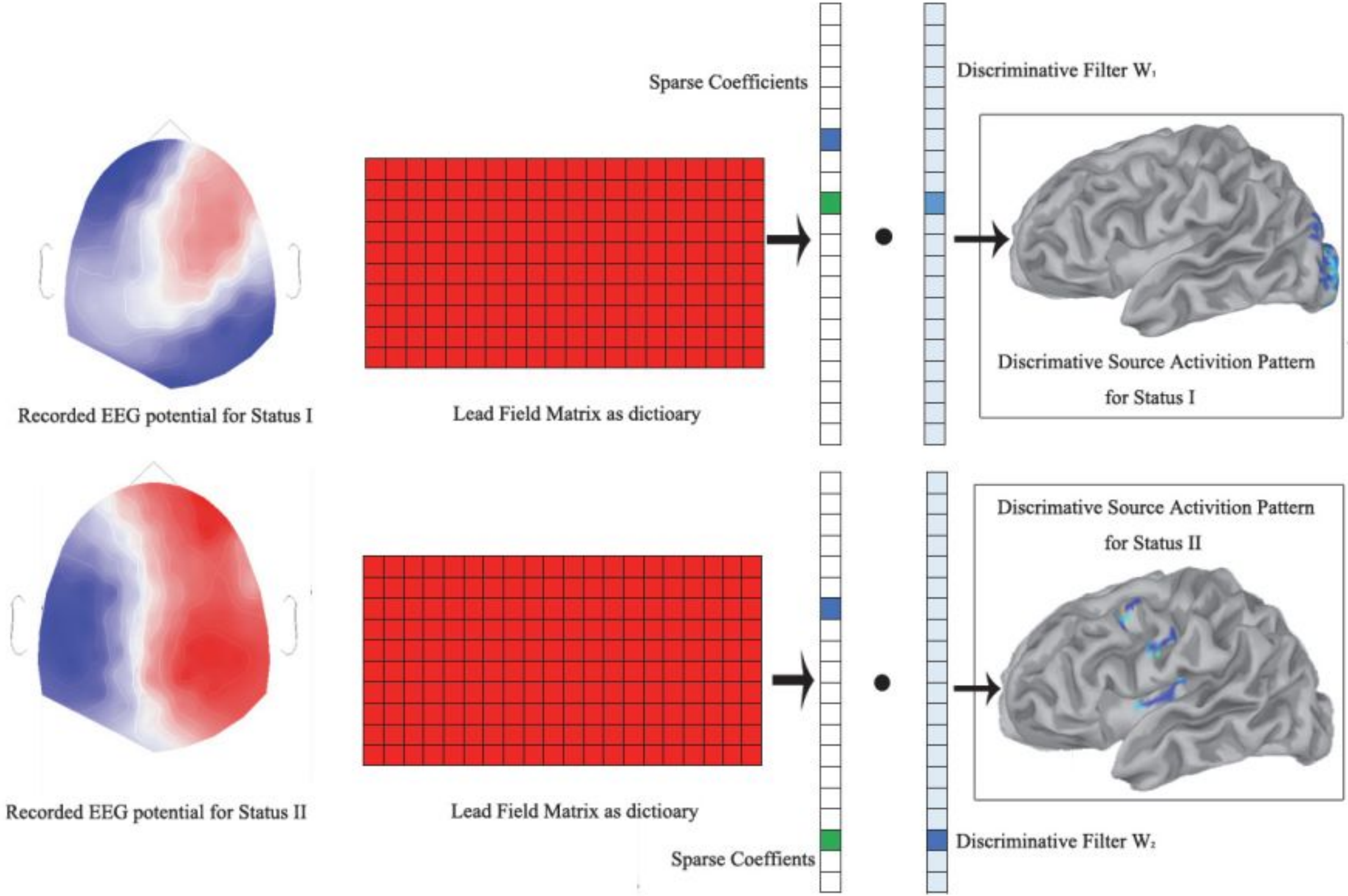
Incorporate a simple linear classifier which can be interpreted as discriminative filters for different brain patterns.

Finding a solution to the Inverse Problem.

Inverse Problem



It starts with the result and calculates the causes.


Sparse Coefficients
Discriminative Filter W1
Discrimative Source Activition Pattern for Status I
Recorded EEG potential for Status I
Lead Field Matrix as dictioary
Discrimative Source Activition Pattern for Status II
Recorded EEG potential for Status II
Lead Field Matrix as dictioary
Discriminative Filter W2
Sparse Coeffients

Discriminative K-SVD

K-SVD focuses on only the representational power of the dictionary (or the efficiency of sparse coding under the dictionary) without considering its capability for discrimination.

In DK-SVD, by directly incorporating the labels in the dictionary-learning stage (as opposed to relying on iteratively updating the dictionary using feedback from the classification stage), we can efficiently obtain a dictionary that retains the representational power while making the dictionary discriminative (i.e., supporting sparse-coding-based classification).

DK-SVD Algorithm

Algorithm 1 Revised DK-SVD algorithm

INPUT: Lead field matrix L, preprocessed EEG signal matrix X, relative controlling scalar β, label matrix H
OUTPUT: classification matrix W, EEG source matrix S
Initialization: Using K-SVD initialization described in Ref.(Aharon, Elad, and Bruckstein 2006)

set $m = 1$
while not converged **do**
  Solve the following sparse coding problem using matching pursuit algorithm for $i = 1, 2, \ldots, N$:
  $\min_{s_i} \|x_i - Ls_i\|_2^2 \quad s.t. \quad \|s_i\|_0 \leqslant T$
  while i is not equal to N_d **do**
    (1) Compute the representation error without atom l_i, $E_i = (X - \sum_{j \neq i}(l_j * s_j))$
    (2) Extract the nonzero entries of s_i and truncate the E_i to E_i^P accordingly.
    (3) SVD decomposition for E_i^P as $E_i^P = U\Lambda V$
    (4) Update l_i and s_i^T:
      $l_i(N_c + 1 : end) \longleftarrow U(:, 1)(N_c + 1 : end)$,
      $\tilde{s}_R^i \longleftarrow \Sigma(1, 1)V(1, :)$.
    (5) Update index $i \longleftarrow i + 1$;
  end while
  $m \longleftarrow m + 1$
end while

Sparse Dictionary Learning

- A Representation Learning Method
  - Automatically discover representations from raw data for feature detection.
  - Machine can both learn and perform specific tasks using learnt Features.
- Aim: To find a sparse representation of Input Data.
- Key Principles:
  - Dictionary has to be Inferred from input data.
  - Represent data using as few components as possible.
- Benefits:
  - Significantly improve sparsity.

The electromagnetic field measured by EEG can be described as the following linear model:

$$X = LS + \in$$

where

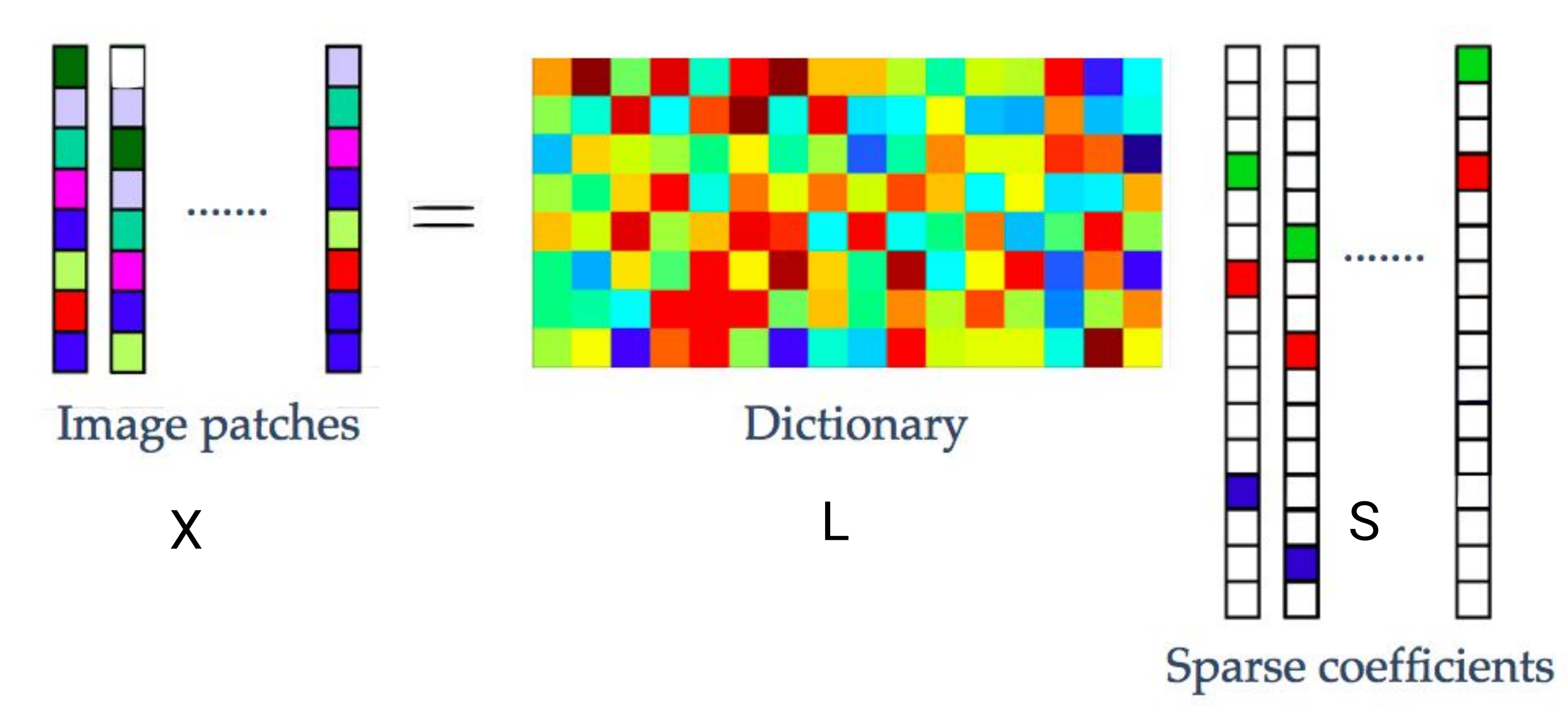
$X \in R\ (N_c\ X\ N_t)$ is the EEG data

$L \in R\ (N_c\ X\ N_d)$ is the lead field matrix(dictionary)

$S \in R\ (N_d\ X\ N_t)$ represents the corresponding driving potentials in N_d source locations

$\in$ is the noise

Pictorial representation of Dictionary Learning



Sparse Coding

Sparse coding is a class of unsupervised methods for learning sets of over-complete bases to represent data efficiently.

The aim of sparse coding is to find a set of basis vectors **ϕ** $_i$ such that we can represent an input vector **x** as a linear combination of these basis vectors:

$$\mathbf{x} = \sum_{i=1}^{k} a_i \phi_i$$

Sparse Coding

- To find S, solve the cost function(Matching Pursuit Algorithm) -

$$\arg\min_{S} \underbrace{\| X - LS \|^2_F}_{\text{Quadratic error}} + \underbrace{\lambda\, \theta(S)}_{\text{Regularisation term}}$$

- To restrict the total number of activated sources less than T(Sparse Coding Problem),

$$\arg\min_{S} \| X - LS \|^2_F \qquad \text{s.t. } \| s_0 \| <= T$$

For EEG recording at time ‘i’, the aim is to represent the signal with minimum error, by trying to find the linear representation from activation patterns in the dictionary L.

Matching Pursuit Algorithm

Matching pursuit(MP) is a sparse approximation algorithm which involves finding the “best matching” projections of multidimensional data onto the span of an over-complete dictionary L.

MP is used to solve the sparse coding problem,

$$\arg\min_{S} || X - LS ||^2_F \qquad \text{s.t.} \qquad || s_0|| <= T$$

Matching Pursuit

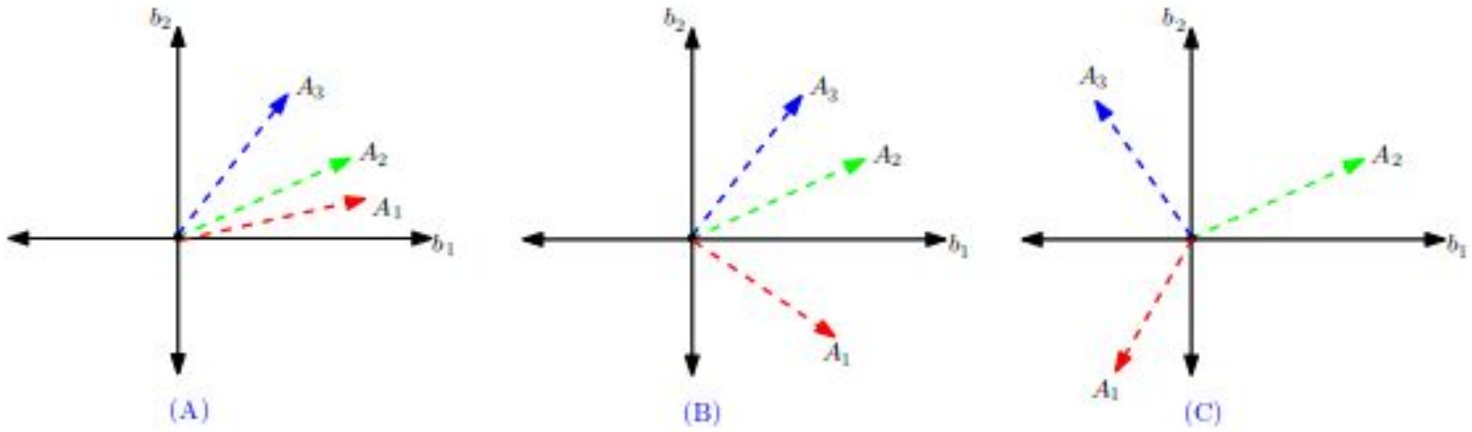
Basic Idea: approximately represent a signal as a weighted sum of finitely many functions (called atoms) taken from *D*.

Normally, not every atom in *D* will be used in this sum. Instead, matching pursuit chooses the atoms one at a time in order to maximally (**greedily**) reduce the approximation error.

Process

Finding the atom that has the biggest inner product with the signal, subtracting from the signal an approximation that uses only that one atom, and repeating the process until the signal is satisfactorily decomposed

Base Coherency indicate how close one base to the others.



Example

$x = \begin{pmatrix} -1.2 \\ 1 \\ 0 \end{pmatrix}$ $A = \begin{pmatrix} -0.707 & 0.8 & 0 \\ 0.707 & 0.6 & -1 \end{pmatrix}$ Therefore $y = A \cdot x$ gives: $y = \begin{pmatrix} +1.65 \\ -0.25 \end{pmatrix}$

Now, Given that : $y = \begin{pmatrix} +1.65 \\ -0.25 \end{pmatrix}$ and $A = \begin{pmatrix} -0.707 & 0.8 & 0 \\ 0.707 & 0.6 & -1 \end{pmatrix}$

How to find original $\mathbf{x}$?

$$b_1 = \begin{pmatrix} -0.707 \\ 0.707 \end{pmatrix} \quad b_2 = \begin{pmatrix} 0.8 \\ 0.6 \end{pmatrix} \quad b_3 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$A \cdot x = \begin{pmatrix} -0.707 & 0.8 & 0 \\ 0.707 & 0.6 & -1 \end{pmatrix} \cdot \begin{pmatrix} -1.2 \\ 1 \\ 0 \end{pmatrix} \qquad = -1.2 \cdot \begin{pmatrix} -0.707 \\ 0.707 \end{pmatrix} + 1 \cdot \begin{pmatrix} +0.8 \\ 0.6 \end{pmatrix} + 0 \cdot \begin{pmatrix} 0 \\ -1 \end{pmatrix} = y = \begin{pmatrix} +1.65 \\ -0.25 \end{pmatrix}$$

1. Find atom with has the biggest inner product with y. $p_i = \max_j < b_j , y >$
2. Calculate the residue. $r_i = p_i - p_i . < p_i , y >$
3. Find atom with has the biggest inner product with r_i
4. Repeat step 2 and 3 until residue achieve a certain threshold.

The reconstructed x is $\begin{pmatrix} -1.34 \\ 1 \\ -0.099 \end{pmatrix}$

Updating Equations

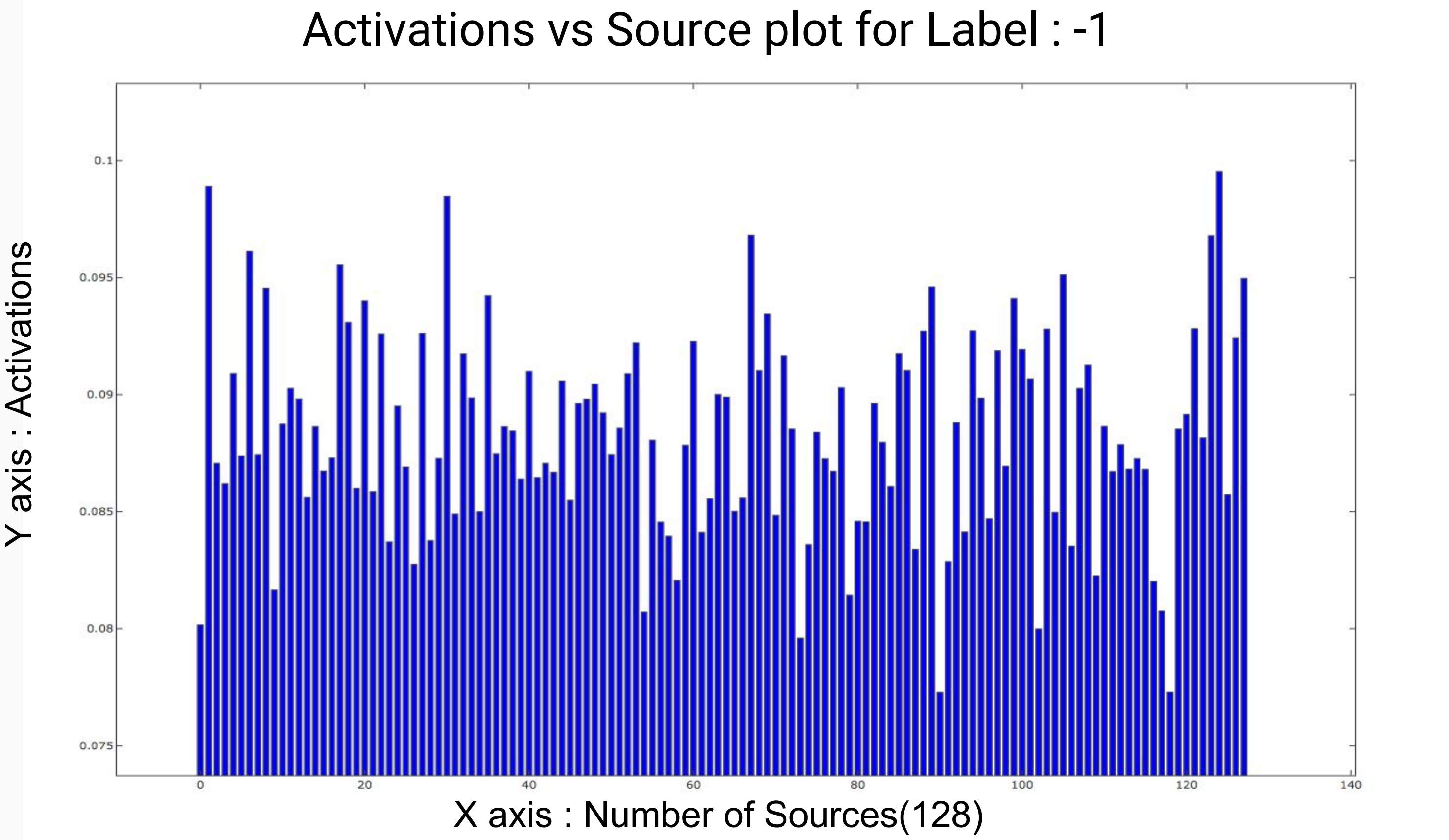
In every iteration:

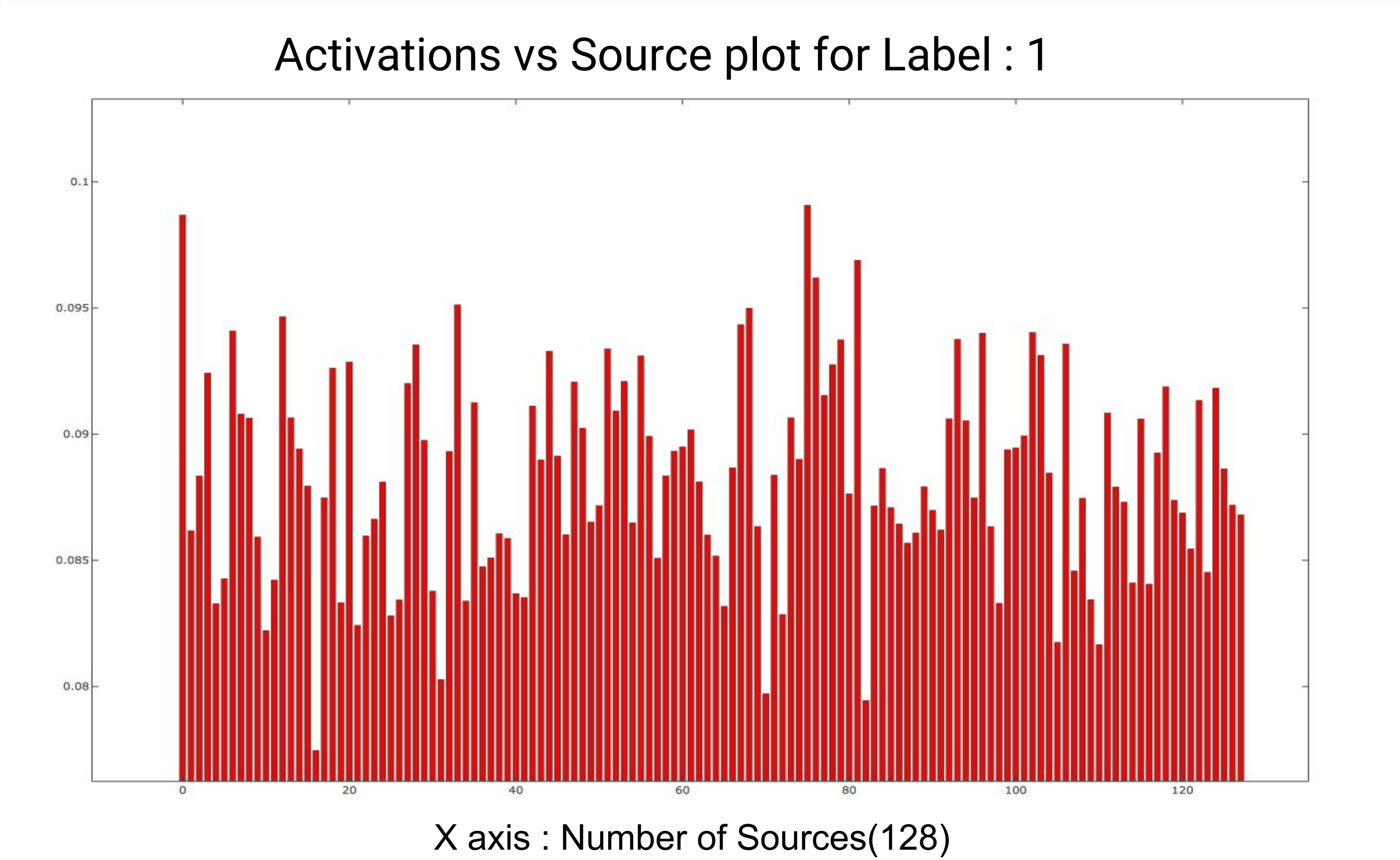
1. Compute the representation error (E_i) without atom l_i
2. Extract the nonzero entries of s_i and truncate the E_i to E_{iP} accordingly.
3. SVD decomposition for E_{iP} as $E_{iP} = U \Lambda V$
4. Update l_i and s_i^T :

   4.1 $l_i(N_c + 1 : end) \leftarrow U(:, 1)(N_c + 1 : end)$,

   4.2 $\tilde{s}^i_R \leftarrow \Sigma(1, 1)V(1, :)$.

Results


Activations vs Source plot for Label : -1
Y axis : Activations
X axis : Number of Sources(128)
0.1
0.095
0.09
0.085
0.08
0.075
0
20
40
60
80
100
120
140


Activations vs Source plot for Label : 1
Y axis : Activations
0.1
0.095
0.09
0.085
0.08
0
20
40
60
80
100
120
X axis : Number of Sources(128)

Applications

DK-SVD can be found widely in use in applications such as

- Image processing - image completion, denoising, inpainting
- Audio processing
- Document analysis.

Thank You!

GitHub link : https://github.com/krishnadwypayan/SMAI-Project