BLIND SOURCE SEPARATION IN NEUROSCIENCE

by

Hakan Ak, Anıl Kamber

A report submitted for EE492 senior design project class
in partial fulfillment of the requirements for the degree of
Bachelor of Science
(Department of Electrical and Electronics Engineering)
in Boğaziçi University

June 3^{rd}, 2024

Principal Investigator: Prof. Burak Acar
Co-Principal Investigator: Prof. Alper T. Erdoğan

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to Prof. Alper T. Erdoğan and Prof. Burak Acar for their unwavering guidance and support throughout this project.

ABSTRACT

We assume that the brain operates in terms of subnetworks that cooperate to perform specific tasks. These subnetworks are presumed to be sparse and overlapping. For instance, multiple subnetworks might be active while the brain is performing a task. The aim is to identify these subnetworks by separating the source signals associated with distinct subnetworks from 32-channel EEG that is recorded while the subject is performing a task. Due to the presumption of overlapping subnetworks and the nonnegative nature of EEG signals, known techniques such as Independent Component Analysis (ICA) and Nonnegative Matrix Factorization (NMF) are not feasible for separating the source signals directly from EEG. ICA presumes that the sources are independent, and NMF assumes that the sources are nonnegative [1], [2]. In this project, we brought Polytopic Matrix Factorization (PMF) that provides "*identifiable polytopes*" to extend feature attributes of latent factors to the neuroscience realm to extract latent vector components associated with distinct subnetworks. Due to the absence of ground truths for the sources expected to be extracted from EEG signals, we used the best-fitted dipole as the ground truth and residual variance as the error metric.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS i

ABSTRACT ii

LIST OF FIGURES v

LIST OF TABLES vi

1 BLIND SOURCE SEPARATION 1
- 1.1 Principal Component Analysis . . . 1
- 1.2 Independent Component Analysis . . . 3
  - 1.2.1 FastICA Algorithm . . . 3
  - 1.2.2 Infomax Algorithm . . . 5
- 1.3 Nonnegative Matrix Factorization . . . 8
- 1.4 Polytopic Matrix Factorization . . . 13

2 WORK COMPLETED 16
- 2.1 EEG Data . . . 16
  - 2.1.1 Preprocessing EEG Signals . . . 17
- 2.2 Applying ICA on EEG Data . . . 18
- 2.3 Applying PMF on EEG Data . . . 19
  - 2.3.1 Dipole Fitting . . . 19
  - 2.3.2 Residual Variance . . . 20
  - 2.3.3 Hyperparameter Tuning . . . 20
  - 2.3.4 Determined Case . . . 21
  - 2.3.5 Conversion into ERP Data . . . 21
  - 2.3.6 PMF Results and Analysis . . . 21
- 2.4 Reproducibility Problem for PMF . . . 25

3 CONCLUSIONS 26
- 3.1 Future Works . . . 26
- 3.2 Realistic Constraints . . . 26
  - 3.2.1 Social, Environmental and Economic Impact . . . 26
  - 3.2.2 Cost Analysis . . . 26
  - 3.2.3 Standards . . . 26

APPENDICES 27

BIBLIOGRAPHY **29**

LIST OF FIGURES

1.1 Large variance was interpreted as signal, where low variance to be noise. [3] . . . 2
1.2 An example of PCA, first the mean of the data shown in Fig. 1.2a was subtracted to make it normalized, then the eigenvectors of the covariance matrix was calculated. In Fig. 1.2b, we see the principal components overlayed on the original data, [4]. . . . 3
1.3 b is the normalized version where H is row-stochastic. The dots are $\boldsymbol{x_l}$. . . . 9
1.4 A demonstration of sufficiently-scattered condition of NMF in the source domain. . . . 10
1.5 The intuition behind the NMF identifiability conditions 11
1.6 The visualization of simplex VolMin-based NMF. . . . 12
1.7 Comparison of the sufficiently-scattering conditions of Sparse PMF and NMF. . . . 15
1.8 Comparison between sufficiently-scattered examples of 2-D and 3-D sparse PMF [7]. . . . 15

2.1 International 10-20 System 16
2.2 BODE Plot for High Pass Filter 17
2.3 BODE Plot for Low Pass Filter 17
2.4 First 30 seconds of the final version of the signal used. 18
2.5 Resulting IC components 19
2.6 The brain topographic maps for the 32 sources discovered by PMF. 22
2.7 Filtered version of Fig. 2.6. 22
2.8 Temporal source activities of the filtered sources. 23
2.9 Best-fitted dipole location, time-frequency analysis and brain topographic map for Source 1 23
2.10 Best-fitted dipole location, time-frequency analysis and brain topographic map for Source 19 24
2.11 Best-fitted dipole location, time-frequency analysis and brain topographic map for Source 28 24
2.12 MNI coordinates of best-fitted dipoles (sources) projected onto middle slices of each view. For reproducibility concerns, the data has been divided into four sequences, each containing an equal number of visual events. Each sequence is regarded as an individual subject. Each color represents a different subject. . . . 25

1 P is the convex combination of x_1, x_2, and x_3 28

LIST OF TABLES

LIST OF APPENDICES

Appendix A **28**

Chapter 1

BLIND SOURCE SEPARATION

This project explores Blind Source Separation (BSS) methods to uncover distinct cognitive subnetworks in the human brain. Despite minimal prior knowledge of the signals, BSS techniques promise to extract valuable information from data. The data can take the form of brain images, EEG, financial data, or other complex phenomena. This investigation aims to reveal these subnetworks by breaking down signals from various sources such as fMRI/BOLD, EEG, and MEG

1.1 Principal Component Analysis

Principal Component Analysis (PCA) is a dimensionality reduction method that aims to reduce the dimensions of the data and obtain useful information from it. PCA defines principal components that explain variations in a data set, allowing for the useful differentiation between groups. This data can take the form of tabular data, mixed audio and video signals, or results from a physical experiment.

Let $\mathbf{X}$ be our original $m \times n$ data matrix, where each column represents different experiments, samples, or observations, and each row represents the sensor signals. The question PCA asks is as follows: *Can we find another basis to best re-express our data?* Here, we assume *linearity* by restricting ourselves to find a basis that is a *linear combination* of its basis vectors. Let $\mathbf{Y}$ be an $m \times n$ matrix, which is the new representation of the data. We can relate $\mathbf{X}$ and $\mathbf{Y}$ by a linear transformation $\mathbf{P}$, such that

$$\mathbf{PX} = \mathbf{Y}. \tag{1.1}$$

Here, the transformation $\mathbf{P}$ can also be interpreted as a rotation or change of basis. Now, we should ask: What is a good choice for $\mathbf{P}$? What is the best way to express $\mathbf{X}$ as $\mathbf{Y}$?

We hope to find our basis by trying to maximize the variance. We interpret larger variances as signal and lower variances as noise.

Within our data, we might have sensors that measure nearly the same attribute,

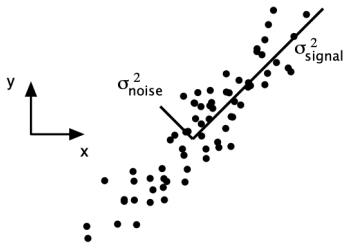


Figure 1.1: Large variance was interpreted as signal, where low variance to be noise. [3]

or they could be *highly correlated*. Recalling that covariance measures the linear relationship between two variables, we can define a covariance matrix as follows:

$$\mathbf{C_X} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathbf{T}, \tag{1.2}$$

which has following properties:

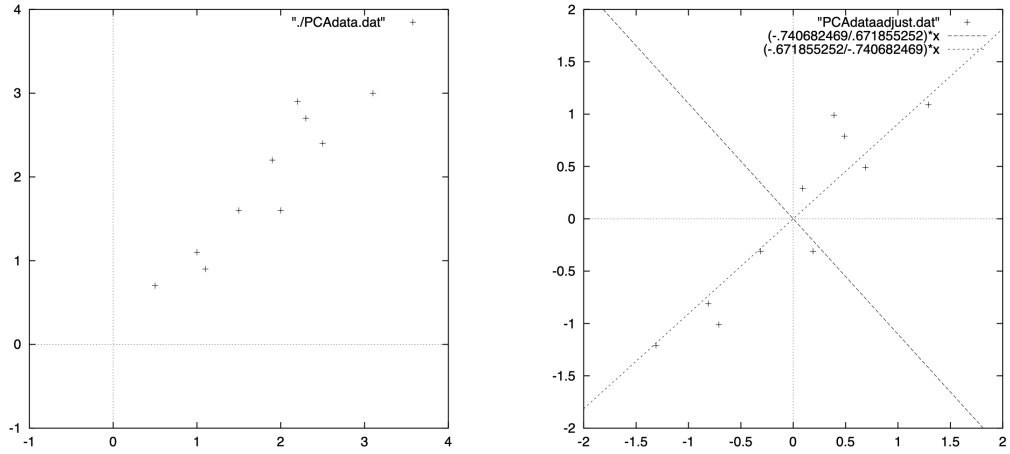
- It is a $m \times m$ symmetric square matrix.
- The diagonal terms are sample variances of specific sensor signals, where large values corresponds to useful information.
- The off-diagonal terms representing the covariance between different sensor signals, where large magnitudes corresponds to redundant information.

Suppose we are able to manipulate $\mathbf{C_X}$ to obtain a manipulated $\mathbf{C_Y}$. For an ideal $\mathbf{C_Y}$, we would want all off-diagonal terms of $\mathbf{C_Y}$ to be zero, meaning $\mathbf{C_Y}$ is a diagonal matrix, and for $\mathbf{Y}$ to be rank ordered according to variance.

So, how do we diagonalize $\mathbf{C_Y}$? PCA assumes that all basis vectors $\mathbf{p_1}, ..., \mathbf{p_m}$ of $\mathbf{P}$ are *orthonormal* and that $\mathbf{P}$ is an *orthonormal matrix*. The covariance matrix can be diagonalized using methods such as eigenvector decomposition or singular value decomposition. The demonstration is omitted here.

PCA has some limitations and fails to solve problems in certain scenarios:

- While PCA tries to decorrelate the data, or remove the second-order dependencies, this might not be sufficient for our data if the first and second-order statistics are not sufficient (e.g., non-Gaussian data). In other words, the important structure in the data might not align with orthogonal axes.
- Our data might not fit well on linear axes. For example, some information might fit perfectly on polar axes.



(a) Original Data (b) Principal components of the data.

Figure 1.2: An example of PCA, first the mean of the data shown in Fig. 1.2a was subtracted to make it normalized, then the eigenvectors of the covariance matrix was calculated. In Fig. 1.2b, we see the principal components overlayed on the original data, [4].

1.2 Independent Component Analysis

Independent Component Analysis (ICA) is a powerful signal processing technique that decomposes a complex mixture of signals into their underlying independent components. It operates on the assumption that the observed signals are a linear combination of unknown source signals, which are statistically independent of each other, [1].

The fundamental concept behind ICA is to identify and separate these sources by exploiting statistical independence, assuming that in real-world scenarios, such as images, sound, or data, the sources are often statistically independent, [1].

The key assumptions in ICA include the statistical independence of the source signals, the non-Gaussianity of the components, and the linearity of their mixtures. By leveraging these assumptions, ICA aims to separate mixed signals into their original, independent components, enabling the exploration and extraction of meaningful information from complex data, [1].

1.2.1 FastICA Algorithm

Below, we will examine an algorithm used to apply ICA, namely FastICA, a *fast fixed-point algorithm.*

FastICA uses negentropy as the measure of independence. Negentropy is defined using differential entropy.

First, we define the differential entropy. Differential entropy, H is defined as:

$$H(\mathbf{y}) = -\int f(\mathbf{y}) log f(\mathbf{y}) d\mathbf{y}. \tag{1.3}$$

Now, we can define negentropy as follows:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}), \tag{1.4}$$

where $\mathbf{y}_{gauss}$ is a Gaussian random variable with the same covariance matrix as $\mathbf{y}$.

Negentropy is well justified by statistical theory and can be seen as the optimal estimator of non-Gaussianity in terms of statistical properties [1]. However, negentropy is difficult to compute directly. Therefore, approximations have been developed to tackle this problem. The approximation used by FastICA is as follows:

$$J(y) \propto [E\{G(y)\} - E\{G(v)\}]^2, \tag{1.5}$$

where $G(v)$ is a non-quadratic function.

With FastICA, we first try to find one computational unit, which is a weight vector $\mathbf{w}$ that we will update using a learning rule. This rule aims to find a direction with a unit vector $\mathbf{w}$ such that the projection $\mathbf{w^T x}$ maximizes the non-Gaussianity.

We will use the described negentropy as the measure of non-Gaussianity. We need $\mathbf{w^T x}$ to have unit variance, so our constraint is that the norm of $\mathbf{w}$ must be *unity*.

If we denote the derivative of the function G in (5) as g, the basic form of FastICA is as follows:

1. Create an initial random vector $\mathbf{w}$
2. $\mathbf{w} \leftarrow E\{\mathbf{x} g(\mathbf{w^T x})\} - E\{g'(\mathbf{w^T x})\}$
3. $\mathbf{w} \leftarrow \mathbf{w}/\|\mathbf{w}\|$
4. Back to step 2 until convergence.

Here, convergence means that consecutive $\mathbf{w}$ vectors point in the same direction (or $\mathbf{w^T w} \approx 1$). Now that we have calculated $\mathbf{w}$ for one component, we want to extend it to finding m independent components. The key point here is that the vectors $\mathbf{w_i}$ corresponding to different ICs are orthogonal since we are working in the whitened space, i.e.,

$$E\{(\mathbf{w_i^T x})(\mathbf{w_j^T x})\} = \mathbf{w_i^T w_j}. \tag{1.6}$$

Thus, to estimate m ICs, we first calculate the vectors $\mathbf{w_1}, ..., \mathbf{w_p}$, then orthogonalize them at each iteration to prevent them from converging to the same maximum [5]. To achieve this, we can use deflationary orthogonalization with the Gram-Schmidt method, which means we will find our ICs one by one. Also, an alternative method is to use symmetric orthogonalization, which aims to find them all at once.

Suppose we have obtained p different $\mathbf{w}$ vectors. For the $(p+1)$-th iteration, we first run our one-unit algorithm to obtain some $\mathbf{w_{p+1}}$. After every iteration step, we subtract all the projections $(\mathbf{w}_{p+1}^T\mathbf{w_j})\mathbf{w_j}$ for $j = 1, ..., p$ (for the previous p vectors) from $\mathbf{w_{p+1}}$.

Now our algorithm becomes as follows:

1. Choose m, the number of ICs. Set $p \leftarrow 1$.

2. Initialize $\mathbf{w_p}$ randomly with unit norm.

3. Update $\mathbf{w_p}$ as follows: $\mathbf{w_p} \leftarrow E\mathbf{x}g(\mathbf{w_p^T x}) - Eg'(\mathbf{w_p^T x})$.

4. Orthogonalize $\mathbf{w_p}$: $\mathbf{w_p} \leftarrow \mathbf{w_p} - \sum_{j=1}^{p-1}(\mathbf{w_p^T w_j})\mathbf{w_j}$.

5. Normalize $\mathbf{w_p}$: $\mathbf{w_p} \leftarrow \frac{\mathbf{w_p}}{\|\mathbf{w_p}\|}$.

6. If $\mathbf{w_p}$ has not converged, go back to step 3.

7. Set $p \leftarrow p + 1$. If $p \leq m$, go back to step 2.

FastICA can also be applied using different approaches. It can be applied by taking the measurement of non-Gaussianity as kurtosis. Additionally, it can be used with the approach of maximum likelihood, [5].

1.2.2 Infomax Algorithm

Before discussing the Infomax Algorithm, we should first define kurtosis.

The fourth moment $E[x^4]$ of the PDF of x is $E[x^4] = \int_{-\infty}^{\infty} p_x(x)x^4, dx$. If x has zero mean, then a normalized version of $E[x^4]$ is known as **kurtosis**, which is defined in terms of the fourth and second central moments as $K = \frac{E[x^4]}{E[x^2]^2} - 3$. Kurtosis provides a measure of how super-Gaussian or "peaky" a PDF is.

We know that signal mixtures tend to have Gaussian PDFs and that source signals have non-Gaussian PDFs. We also know that each source signal can be extracted by orthogonal projection; however, we do not know how to find an unmixing vector. One type of method to address this is called *projection pursuit*.

Projection pursuit seeks one projection at a time such that the extracted signal is as non-Gaussian as possible. The bad news is that the Central Limit Theorem is not true in general; that is, it is not always true that any Gaussian signal is a mixture of non-Gaussian signals. However, in practice, this often holds.

Super-Gaussian signals have PDFs that are more peaky than that of a Gaussian signal, whereas sub-Gaussian signals have PDFs that are less peaky than that of a Gaussian signal. We will assume that our source signals are super-Gaussian. Then we would expect:

- The kurtosis of the extracted signal y will be maximum precisely when $y = s$.
- The kurtosis of the extracted signal y will be maximum when $\mathbf{w}$ is orthogonal to the projected axis $S_1{}'$ or $S_2{}'$.

Since projection pursuit extracts one source signal in each operation, we need to use a reduced set of signal mixtures after each extraction. This is achieved using a method known as *Gram-Schmidt orthogonalization*. To give a geometric intuition, if projection pursuit extracts the M-th source signal using a vector $\mathbf{w_M}$, then this $(M-1)$D space contains the $(M-1)$ transformed axes $(S_1{}', S_2{}', S_3{}', S_4{}', \ldots, S_{M-1}{}')$.

When trying to find mutually independent source signals in higher dimensions rather than just two dimensions, we need a measure to determine how close the extracted signals are to being independent. While we cannot measure independence directly, we can measure *entropy*.

Entropy is a measure of the uniformity of the distribution of a bounded set of values, defined as:

$$H(Z) = -\sum_i p_i \ln p_i. \tag{1.7}$$

Given an invertible multivariate function g of signals $\mathbf{y} = \mathbf{W}\mathbf{x}$ extracted by the unmixing matrix $\mathbf{W}$, find a $\mathbf{W}$ such that the signals $\mathbf{Y} = g(\mathbf{y})$ have maximum entropy. It turns out that the entropy of the signals $\mathbf{Y}$ is maximized if the extracted signals have a CDF that matches the function g. This *cdf-matching* property is a key aspect of ICA. As the derivative of a CDF defines the corresponding PDF, we can assume that source signals have specific PDFs, such as super-Gaussian. Since g is an invertible function, if the signals $\mathbf{Y} = g(\mathbf{y})$ are independent, then the extracted signals $\mathbf{y} = g^{-1}(\mathbf{Y})$ are also independent.

If we now consider the signal $Y = g(y)$, where $y = \mathbf{w}^T\mathbf{x}$ is a single row $\mathbf{w}^T$ of an unmixing matrix $\mathbf{W}$, then the entropy is,

$$H(Y) = -\frac{1}{N}\sum_{t=1}^{N} \ln p_Y(Y^t). \tag{1.8}$$

We will show that if the CDF function of y is g, then Y has maximum entropy.

Assume y has a univariate PDF. We can then extend our solutions to the multivariate case. Let's define a small interval Δy around $y = y^1$. Then, the probability that y is in this small interval is:

$$p(y^1 - \Delta y/2 < y \leq y^1 + \Delta y/2) \approx p_y(y^1)\Delta y. \tag{1.9}$$

Similarly,

$$p(Y^1 - \Delta Y/2 < Y \leq Y^1 + \Delta Y/2) \approx p_Y(Y^1)\Delta Y. \tag{1.10}$$

Given that $Y^1 = g(y^1)$, the probability of observing y in the range of $y^1 \pm \Delta y/2$ equals the probability of observing Y in the range $Y^1 \pm \Delta Y/2$. Then, this is equivalent to:

$$p_Y(Y^1)\Delta Y = p_y(y^1)\Delta y. \tag{1.11}$$

With an algebraic manipulation and making $\Delta y \to 0$, note that CDF functions have non-negative derivatives:

$$p_Y(Y) = \frac{p_y(y)}{|\frac{dY}{dy}|}, \tag{1.12}$$

where $|\frac{dY}{dy}|$ is known as the Jacobian. We can omit the absolute value operator since PDFs are never negative. We can replace g' with the PDF p_s assumed for the source signals, $g' = p_s$. This can be substituted into the entropy equation as follows:

$$H(Y) = -\frac{1}{N}\sum_{t=1}^{N} \ln \frac{p_y(y^t)}{p_s(y^t)}. \tag{1.13}$$

The fraction inside the summation can be represented as a subtraction, and this difference between two PDFs is known as Kullback-Leibler divergence or relative entropy. If $p_y = p_s$, then the random variable $\mathbf{Y}$ has a uniform distribution, which maximizes entropy. The same technique can be applied to multivariate PDFs.

Maximizing the joint entropy of $\mathbf{Y}$ also maximizes the amount of mutual information between $\mathbf{x}$ and $\mathbf{Y}$, [6]. For this reason, the above description is called *infomax*.

1.3 Nonnegative Matrix Factorization

Nonnegative Matrix Factorization (NMF) is a method that aims to factor a data matrix into some low-rank latent factor matrices with a nonnegativity constraint [2]. NMF seeks to find such a factorization model:

$$\mathbf{X} \approx \mathbf{WH}, \quad \mathbf{W} \in \mathbb{R}^{M \times R}, \quad \mathbf{H} \in \mathbb{R}^{R \times N}, \tag{1.14}$$

where $\mathbf{W} \geq \mathbf{0}$, $\mathbf{H} \geq \mathbf{0}$, and $R \leq \min\{M, N\}$, with R denoting the *factorization rank*. The standard NMF problem can be formulated as follows:

$$\min_{\mathbf{W} \in \mathbb{R}^{M \times R}, \mathbf{H} \in \mathbb{R}^{R \times N}} ||\mathbf{X} - \mathbf{WH}||_F \text{such that} \qquad \mathbf{W}, \mathbf{H} \geq \mathbf{0} \tag{1.15}$$

Here, another measure of distance could be used instead of the Frobenius norm.

The problem of finding an exact factorization, that is, finding $\mathbf{W} > \mathbf{0}$ and $\mathbf{H} > \mathbf{0}$ such that $\mathbf{X} = \mathbf{WH}$, is referred to as *exact NMF*. The minimum R that an exact NMF exists is the *nonnegative rank* of $\mathbf{X}$, denoted $\text{rank}_+(\mathbf{X})$. We have $\text{rank}(\mathbf{X}) \leq \text{rank}_+(\mathbf{X}) \leq \min(N, M)$ since the solution $\mathbf{X} = \mathbf{MI} = \mathbf{IM}$ is trivial.

For NMF, identifiability refers to the ability to identify the data-generating factors $\mathbf{W}$ and $\mathbf{H}$ that give rise to $\mathbf{X} = \mathbf{WH}$. A matrix factorization model without constraints on the latent factors is nonunique, since we have

$$\mathbf{X} = \mathbf{WH} = \mathbf{WQ}(\mathbf{Q}^{-1}\mathbf{H}), \tag{1.16}$$

for any invertible $\mathbf{Q} \in \mathbb{R}^{R \times R}$. Therefore, for whichever $(\mathbf{W}, \mathbf{H})$ that fits the model, the solution $(\mathbf{WQ}, \mathbf{Q}^{-1}\mathbf{H})$ fits it equally well. Intuitively, adding constraints such as nonnegativity on the latent factors may help us achieve identifiability, since constraints can reduce the volume of the solution space and remove many possibilities for $\mathbf{Q}$. When looking for identifiability, we often ignore cases where $\mathbf{Q}$ only results in column reordering and scaling of $\mathbf{W}$ and $\mathbf{H}$, which is considered unavoidable but also inconsequential in most applications [2].

Let's change our notation and recall that $\mathbf{X} = \mathbf{WH}^T$, where $\mathbf{W} \in \mathbb{R}^{M \times R}$, $\mathbf{H} \in \mathbb{R}^{N \times R}$, and $\mathbf{W} \geq 0$, $\mathbf{H} \geq 0$, with $R \geq \min\{M, N\}$. Since $\mathbf{H}$ is nonnegative;

$$\vec{x_l} \in \text{cone}\{\mathbf{W}\}, \quad l = 1, 2, 3, \ldots, N \tag{1.17}$$

Suppose that $\mathbf{H}$, in addition to being nonnegative, also satisfies a row sum-to-one assumption, i.e.,

$$\mathbf{H1} = \mathbf{1}, \quad \mathbf{H} \geq 0. \tag{1.18}$$

When $\mathbf{H}$ is row-stochastic, we have

$$\vec{x_l} \in \text{conv}\{\mathbf{W}\}, \quad l = 1, 2, 3, \ldots, N, \tag{1.19}$$

where

$$\text{conv}\{\mathbf{W}\} = \{\mathbf{x} \in \mathbb{R}^M \mid \mathbf{x} = \mathbf{W}\boldsymbol{\theta}, \boldsymbol{\theta} \geq 0, \boldsymbol{\theta}^T\mathbf{1} = 1\} \tag{1.20}$$

denotes the convex hull of the columns of $\mathbf{W}$. When $\text{rank}(\mathbf{W}) = R$, it can be shown that the columns of $\mathbf{W}$ are also the vertices of this convex hull. However, the converse is not true.

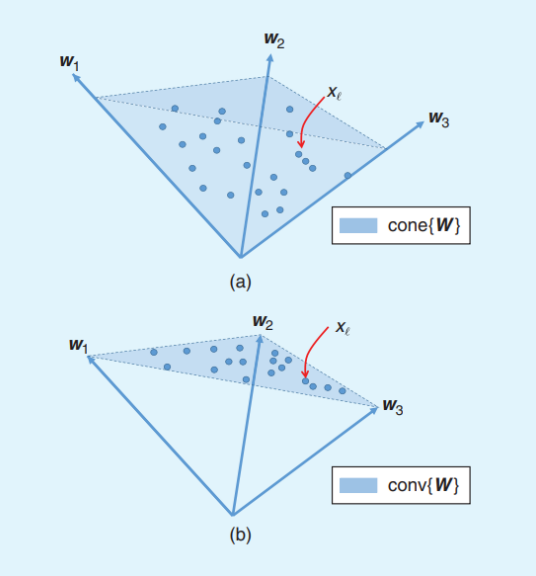


Figure 1.3: b is the normalized version where H is row-stochastic. The dots are $\boldsymbol{x_l}$.

From the geometries, it is clear that NMF is tantamount to recovering a data enclosing conic hull, which can be very ill-posed as many solutions may exist. Any $\hat{\mathbf{W}} = [\hat{\mathbf{w}}_\mathbf{1}, \hat{\mathbf{w}}_\mathbf{2}, ..., \hat{\mathbf{w}}_\mathbf{R}]$ that satisfies $\text{conv}\{\mathbf{X}\} \subseteq \text{conv}\{\mathbf{W}\} \subseteq \mathbb{R}_+^M$ also satisfies the data model $\mathbf{X} = \hat{\mathbf{W}}\hat{\mathbf{H}}^\mathbf{T}$ for some $\hat{\mathbf{H}}$, where both $\mathbf{H}$ and $\hat{\mathbf{H}}$ satisfy the row-stochastic assumption.

Intuitively, if the data points $\mathbf{x_l}$ are well spread in $\mathbb{R}_+^M$ such that $\text{conv}\{\mathbf{X}\} \approx \mathbb{R}_+^M$, then it would be hard to find a $\hat{\mathbf{W}}$ such that $\text{conv}\{\mathbf{X}\} \subseteq \text{conv}\{\mathbf{W}\} \subseteq \mathbb{R}_+^M$ holds - and a unique factorization model is then possibly guaranteed [2]. In the literature, such characterizations are usually described in terms of the geometric properties of $\mathbf{H}$. The scattering of the rows of $\mathbf{H}$ is directly related to the column vectors of $\mathbf{X}$ since they are linked by a full-column-rank matrix $\mathbf{W}$ [2].

Definition: A nonnegative matrix $\mathbf{H} \in \mathbb{R}^{N \times R}$ is said to satisfy the separability condition if

$$\text{cone}\{\mathbf{H}^T\} = \text{cone}\{\mathbf{e_1}, \mathbf{e_2}, ..., \mathbf{e_R}\} = \mathbb{R}_+^R. \tag{1.21}$$

In the literature, $\mathbf{H}$ is said to satisfy the separability condition if every row vector of $\mathbf{H}$ can be represented as

$$\mathbf{H}(l_r, :) = \alpha_r \mathbf{e_r}^T, \tag{1.22}$$

where $\alpha_r > 0$ is a real number and $\mathbf{e_r}$ is the r^{th} coordinate vector in $\mathbb{R}^R$. When $\alpha_r = 1$, $\mathbf{H}$ is row-stochastic.

The separability condition is considered a relatively restrictive condition. It means that the extreme rays of the nonnegative orthant are touched by the rows of $\mathbf{H}$, and thus the conic hull of $\mathbf{H}^T$ equals the *entire nonnegative orthant* [2]. It is easy to see that $\mathbf{x_l} = \alpha_r \mathbf{w_r}$ exists for each $r = 1, \ldots, R$, i.e., the $\mathbf{x_l}$'s are scaled versions of the corners in cone$\{\mathbf{W}\}$. This means that cone$\{\mathbf{X}\}$ = cone$\{\mathbf{W}\}$ under separability. Another condition that is much more relaxed than separability and can effectively model the spread of the nonnegative vectors is called *sufficiently scattered*.

A nonnegative matrix $\mathbf{H} \in \mathbb{R}^{N \times R}$ is said to be sufficiently scattered if the following two conditions are satisfied: Firstly, the rows of $\mathbf{H}$ are spread enough so that

$$C \subseteq \text{cone}\{\mathbf{H}^T\}, \tag{1.23}$$

where C is a second-order cone defined as

$$C = \{\mathbf{x} \in \mathbb{R}^R \mid \mathbf{x}^T \mathbf{1} > \sqrt{R-1}\|\mathbf{x}\|_2\}. \tag{1.24}$$

Secondly, cone$\{\mathbf{H}^T\} \subseteq$ cone$\{\mathbf{Q}\}$ does not hold for any orthonormal $\mathbf{Q}$ except *the permutation matrices*. This second-order cone is tangent to every facet of the nonnegative orthant [2].

Assume $\mathbf{H}$ is both nonnegative and row-stochastic. Under separability, there is an index set $\mathbf{\Lambda} = \{l_1, l_2, l_3, \ldots, l_R\}$ that collects the indices satisfying $\mathbf{H}(l_r, :) = \mathbf{e_r}^T$ for $r = 1, 2, \ldots, R$. The vertices of the convex hull are now touched by some data samples. The problem is now reduced to finding data columns that span a convex hull that encloses all the other data columns.

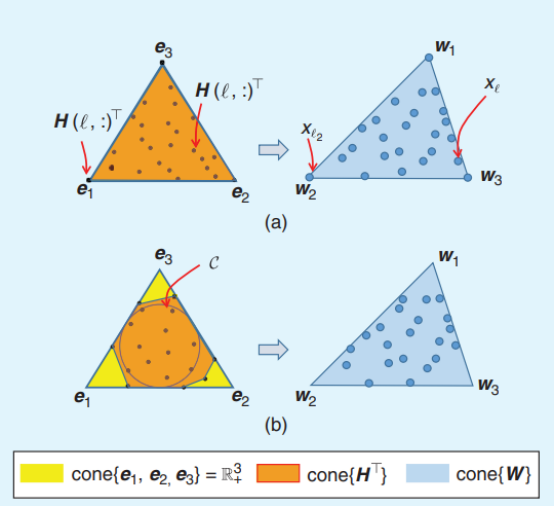


Figure 1.4: A demonstration of sufficiently-scattered condition of NMF in the source domain.

The vertices of a convex hull cannot be represented by any other vectors that reside inside the convex hull using their convex combinations. If noise is absent,

all $\mathbf{x_l}$ are distinct, and $l \subseteq \mathbf{\Lambda}$, then we always have:

$$\min_{\boldsymbol{\theta}\geq 0, \mathbf{1}^T\boldsymbol{\theta}=1} \|\mathbf{x_l} - \mathbf{X}_{-l}\boldsymbol{\theta}\|_q > 0, \tag{1.25}$$

where the notation $\mathbf{X}_{-l}$ means that the column $\mathbf{x_l}$ is removed from $\mathbf{X}$. When $l \nsubseteq \Lambda$;

$$\min_{\boldsymbol{\theta}\geq 0, \mathbf{1}^T\boldsymbol{\theta}=1} \|\mathbf{x_l} - \mathbf{X}_{-l}\|_q = 0, \tag{1.26}$$

The previous approach needs to solve N convex problems, each of which has $N-1$ decision variables, which can be quite inefficient when N is large. To avoid a large number of convex programs, by assuming row-stochasticity of $\mathbf{H}$:

$$\min_{C} \|\mathbf{C}\|_{\text{row-0}}, \tag{1.27}$$

subject to,

$$\mathbf{X} = \mathbf{XC}, \tag{1.28}$$

$$\mathbf{C} \geq 0, \mathbf{1}^T\mathbf{C} = \mathbf{1}^T, \tag{1.29}$$

where $\min_C \|\mathbf{C}\|_{\text{row-0}}$ counts the number of nonzero rows in $\mathbf{C}$. W.L.G let's say that $\mathbf{X} = [\mathbf{W}, \mathbf{X}']$. Then, it is obvious that:

$$\mathbf{C} = [\mathbf{H}^T, 0]^T. \tag{1.30}$$

There are some cases where separability does not hold in practice. If separability is violated, can we still guarantee identifiability of $\mathbf{W}$ and $\mathbf{H}$?

If both $\mathbf{W}$ and $\mathbf{H}$ are sufficiently scattered, then any optimal solution $(\hat{\mathbf{W}}, \hat{\mathbf{H}})$ is identifiable. Specifically, let us assume that there exists $\mathbf{Q}$ such that $\mathbf{X} = \mathbf{WQ}(\mathbf{HQ}^{-\mathbf{T}})^{\mathbf{T}}$ holds. Intuitively, $\mathbf{Q}^{-\mathbf{T}}$ is an operator that can rotate, enlarge, or shrink. If $\mathbf{H}$ is separable or sufficiently scattered and $\mathbf{HQ}^{-\mathbf{T}}$ is constrained to be nonnegative, one can see that $\mathbf{Q}^{-\mathbf{T}}$ cannot rotate or enlarge the area since this will result in negative elements of $\mathbf{HQ}^{-\mathbf{T}}$ [2]. If $\mathbf{W}$ is sufficiently scattered or if there are some rows in $\mathbf{W}$ touching the boundary of the nonnegative orthant, then $\mathbf{WQ}$ will be infeasible since $\mathbf{Q}$ cannot shrink or enlarge the corresponding area determined by cone$\{\mathbf{W}^T\}$. Hence, $\mathbf{Q}$ can only be a permutation and column-scaling matrix [2].

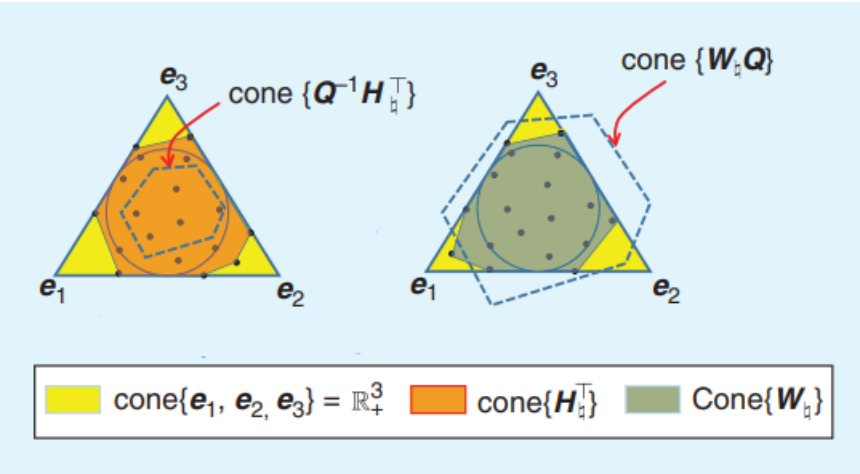


Figure 1.5: The intuition behind the NMF identifiability conditions

It is noted that in the literature of NMF, a necessary condition for the plain NMF identifiability is that both $\mathbf{W}$ and $\mathbf{H}$ contain some zero elements. In some cases, $\mathbf{W}$ does not contain any zero elements. Under such circumstances, can we still guarantee identifiability of $\mathbf{W}$ and $\mathbf{H}$?

If the $\mathbf{x_l}$ are sufficiently spread in conv$\{\mathbf{W}\}$, then finding the minimum-volume enclosing the data points identifies $\mathbf{W}$:

$$\min_{\mathbf{W},\mathbf{H}} \det(\mathbf{W}^T\mathbf{W}), \tag{1.31}$$

subject to $\mathbf{X} = \mathbf{W}\mathbf{H}^T$, where $\mathbf{H}$ is nonnegative and row-stochastic. Actually, $\mathbf{H}$ does not necessarily need to be row-stochastic. If $\mathbf{H}$ is nonnegative and

$$\mathbf{1}^T\mathbf{H} = \rho\mathbf{1}^T, \tag{1.32}$$

it is sufficient.

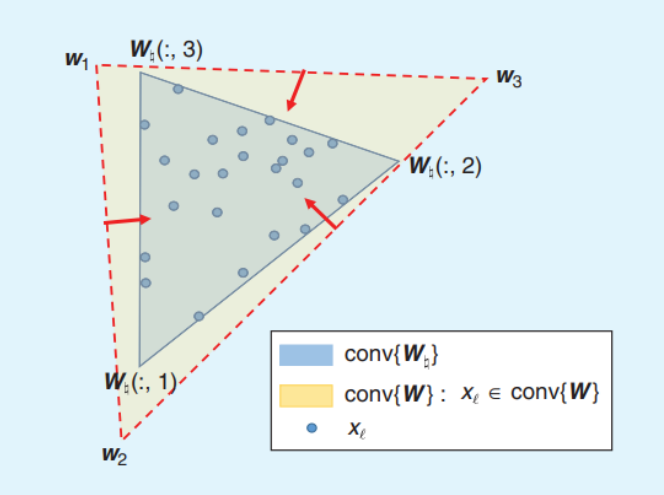


Figure 1.6: The visualization of simplex VolMin-based NMF.

Consider a square full-rank $\mathbf{W}$. It has a unique inverse, $\mathbf{Z} = \mathbf{W}^{-1}$. Then the Simplex VolMin problem can be restated as:

$$\max_{\mathbf{Z}} |\det(\mathbf{Z})|, \tag{1.33}$$

since $\det(\mathbf{W}^T\mathbf{W}) = |\det(\mathbf{Z})|^2$ when $M = R$, and minimizing $|\det(\mathbf{W})|$ is equivalent to maximizing $|\det(\mathbf{Z})|$, subject to

$$\mathbf{1}^T\mathbf{Z}\mathbf{X} = \mathbf{1}^T, \quad \mathbf{Z}\mathbf{X} \geq 0. \tag{1.34}$$

One of the routes is successive convex approximation. The objective can be changed to minimizing $-\log|\det(\mathbf{Z})|$, which does not change the optimal solution, but the log function keeps $-\log|\det(\mathbf{Z})|$ safely inside the differentiable domain, as it penalizes nonsingular $\mathbf{Q}$ very heavily. Then by right-multiplying

$\mathbf{X}^\dagger = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}$, where $\mathbf{X}^\dagger$ implies pseudo-inverse, the resulting constraint becomes $\mathbf{1}^T\mathbf{Z} = \mathbf{a}$.

The cost function can be approximated by the following quadratic function locally around $\mathbf{Z} = \mathbf{Z}^{(t)}$:

$$\hat{f}(\mathbf{Z}; \mathbf{Z}^{(t)}) = f(\mathbf{Z}^{(t)}) + \langle \nabla_{\mathbf{Z}} f(\mathbf{Z}^{(t)}), \mathbf{Z} - \mathbf{Z}^{(t)} \rangle + \frac{\gamma}{2}\|\mathbf{Z} - \mathbf{Z}^{(t)}\|_F^2, \tag{1.35}$$

where $f(\mathbf{Z})$ denotes the objective in equations (1) and (2), and γ is associated with step size. At each iteration:

$$\mathbf{Z}^{(t+1)} \longleftarrow \arg \min_{\mathbf{1}^T\mathbf{Z}=\mathbf{a}, \mathbf{Z}\mathbf{X}\geq 0} \hat{f}(\mathbf{Z}; \mathbf{Z}^{(t)}). \tag{1.36}$$

1.4 Polytopic Matrix Factorization

In unsupervised scenarios, sources are unknown, and no training data is available for their estimation. Structured matrix factorization techniques rely on prior information or assumptions about the sources, such as rank, nonnegativity, sparsity, and antisparsity, to accomplish the desired decomposition [7]. For instance, for NMF, the domain choice is simply the $\mathbb{R}^+$.

The framework that PMF considers is a semi-structured data model, in which the input matrix, $\mathbf{X}$, is modeled as the product of a full column rank matrix and a matrix containing samples from a polytope as its column vectors:

$$\mathbf{X} = \mathbf{W}_g\mathbf{H}_\mathbf{g}, \mathbf{W}_g \in \mathbb{R}^{M\times r}, \mathbf{H}_\mathbf{g} \in \mathbb{R}^{r\times N}, \tag{1.37}$$

$$H_j \in \mathcal{P}, \quad j = 1, \ldots, N. \tag{1.38}$$

where $r \leq min\{M, N\}$ and r denotes the *factorization rank*. The subscript g denotes the ground truth of the factor matrices, and the convex polytope is defined by

$$\mathcal{P} = \{\mathbf{x} \mid \langle \mathbf{a}_i, \mathbf{x} \rangle \leq b_i,\ i = 1, \ldots, f\}, \tag{1.39}$$

where f is the number of faces, and vectors $\mathbf{a}_i$ are the face normals. The choice of the polytope indicates the assumed characteristics of the latent components. PMF aims to factor $\mathbf{X}$ in the form $\mathbf{WH}$ such that

$$\mathbf{W} = \mathbf{W}_g\mathbf{\Pi}^T\mathbf{D}^{-1}, \tag{1.40}$$

$$\mathbf{H} = \mathbf{D}\mathbf{\Pi}\mathbf{H}_\mathbf{g}, \tag{1.41}$$

where $\mathbf{\Pi} \in \mathbb{R}^{r\times r}$ is a permutation matrix that represents the irresolvable ambiguity in obtaining the ordering of the columns of the factor matrices, and $\mathbf{D} \in \mathbb{R}^{r\times r}$ is a full column rank matrix that represents scaling ambiguity [7]. In the literature, it is shown that all polytopes that comply with a specific symmetry restriction qualify

for the PMF framework, and they are referred to as identifiable polytopes [7]. The availability of an infinite number of polytopes provides flexibility in characterizing latent vector components.

For the identification of factor matrices, *determinant-maximization*, which has been utilized for NMF [8], is proposed. Since the determinant of the sample autocorrelation matrix is used as a scattering measure, the problem of extracting sources could be formulated as the following optimization problem:

$$\begin{aligned} \underset{\mathbf{W}\in\mathbb{R}^{M\times r},\mathbf{H}\in\mathbb{R}^{r\times N}}{\text{maximize}} \quad & \det(\mathbf{H}\mathbf{H}^T) \\ \text{subject to} \quad & \mathbf{X} = \mathbf{W}\mathbf{H}, \\ & \mathbf{H}_{:,j} \in \mathcal{P}, \quad j = 1, \ldots, N. \end{aligned} \tag{1.42}$$

Identifiability is a crucial concept in matrix factorization methods, including NMF [2]. The PMF framework offers a new identifiability condition that is applicable to identifiable polytopes. If latent vectors are not sufficiently-scattered and are gathered in a small subregion of the polytope, they will fail to reflect its topology [7]. The Det-Max criterion's ability to recover factor matrices depends on the scattering degree of latent vectors within the polytope. Consequently, it would not succeed in this specific situation. Thus, a sufficient condition for the scattering of latent vectors within the polytope, enabling identifiability for the Det-Max criterion, is proposed in [7]. PMF identifiability results in:

- $\mathcal{P} = \mathcal{B}_\infty$, i.e., the unit ℓ_∞-norm ball, which we refer to as the "antisparse" PMF case,
- $\mathcal{P} = \mathcal{B}_1$, i.e., the unit ℓ_1-norm ball, which we refer to as the "sparse" PMF case,
- $\mathcal{P} = \mathcal{B}_\infty \cap \mathbb{R}^+_n = \{x | 0 \leq x \leq 1\}$, which is referred to as the "antisparse nonnegative" PMF case,
- $\mathcal{P} = \mathcal{B}_1 \cap \mathbb{R}^+_n = \{x | 1^T x \leq 1, x \geq 0\}$, which is referred to as the "sparse nonnegative" PMF case.

The proposed criterion uses the maximum volume inscribed ellipsoid (MVIE) of the polytope, which serves as its best inscribed ellipsoidal approximation [7]. To this new criterion, the samples should be sufficiently-scattered across the polytope such that their convex hull would also include the MVIE of the polytope under a particular tightness constraint.

For comparison with NMF, the sufficient-scattering condition of NMF states that the convex hull of the latent vectors must include the second order cone, as shown in Fig. 1.4. The second order cone, $\left\{x \mid x^T 1 \geq \sqrt{r-1}\|x\|_2, x \in \mathbb{R}^r\right\}$ is used in NMF as a measure of scattering within $\mathbb{R}^+$ [2]. On the other hand, for PMF, the MVIE of the polytope is utilized as a measure of scattering within the polytope. As shown in Fig. 1b, the sphere is the MVIE of the ℓ_1 norm ball in

3-D. For each polytope choice, the MVIE must be determined.

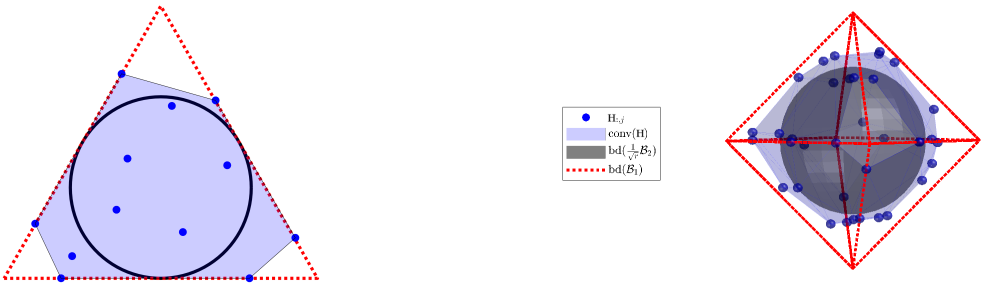


Figure 1.7: Comparison of the sufficiently-scattering conditions of Sparse PMF and NMF.

For EEG, $\mathcal{P}$ is chosen to be the ℓ_1 norm ball, since the relationship between the sparsity and ℓ_1 norm constraints is well-examined in [9], [10].

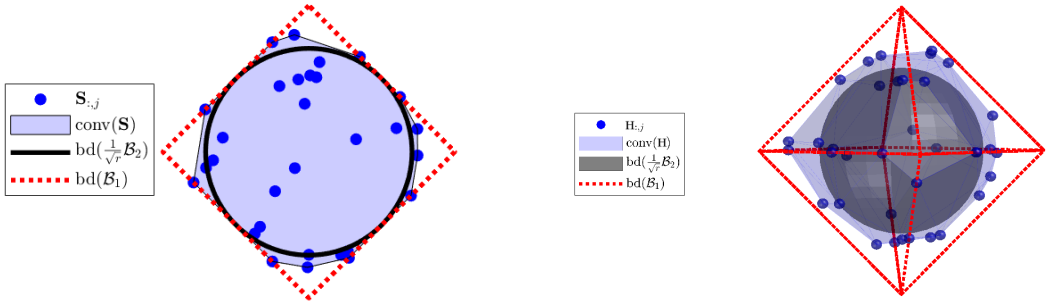


Figure 1.8: Comparison between sufficiently-scattered examples of 2-D and 3-D sparse PMF [7].

Chapter 2

WORK COMPLETED

2.1 EEG Data

The dataset used for the applications of ICA and PMF is EEG data from a selective visual attention experiment [11]. This EEG data was measured during an experiment conducted at the Swartz Center for Computational Neuroscience (SCCN). In the experiment, a stimulus briefly appears in one of five squares arrayed horizontally above a central fixation cross. In each experimental block, one (target) square is differently colored from the rest. Whenever a stimulus appears in the target box, the subject is asked to respond quickly with a right thumb button press. If the stimulus is a circular disk, the subject is instructed to ignore it.

The EEG was sampled at 128Hz and consists of 32 channels. The total length of the data is 238,305 seconds.

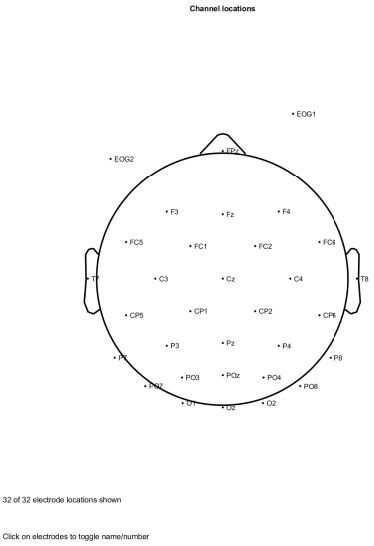


Figure 2.1: International 10-20 System

2.1.1 Preprocessing EEG Signals

For these steps, the EEGLAB environment on MATLAB was employed [12]. Firstly, a Hamming-windowed Sinc FIR Low Pass Filter (LPF) with a cutoff frequency of 1 Hz was used. Then, a Hamming-windowed Sinc FIR High Pass Filter (HPF) with a cutoff frequency of 45 Hz was applied.

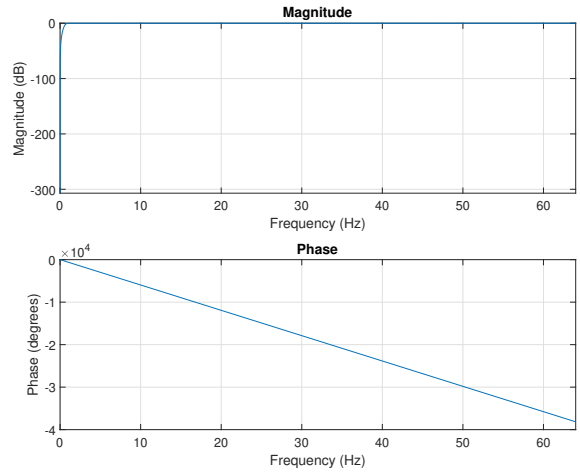


Figure 2.2: BODE Plot for High Pass Filter

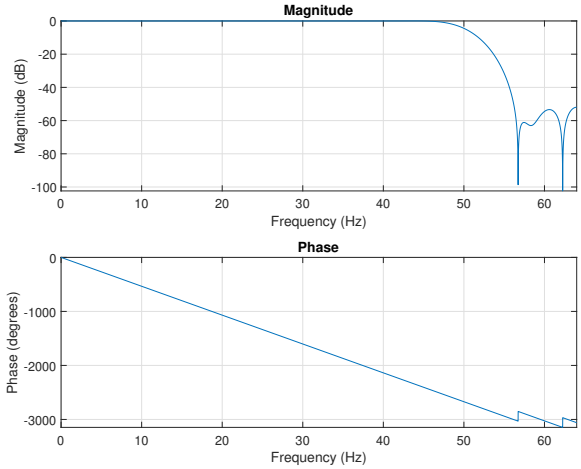


Figure 2.3: BODE Plot for Low Pass Filter

After the filtering steps, to cope with the differences between the electrode voltages, the channels were referenced to the average of all channels. Then, the data was upsampled to a 1024 Hz sampling frequency.

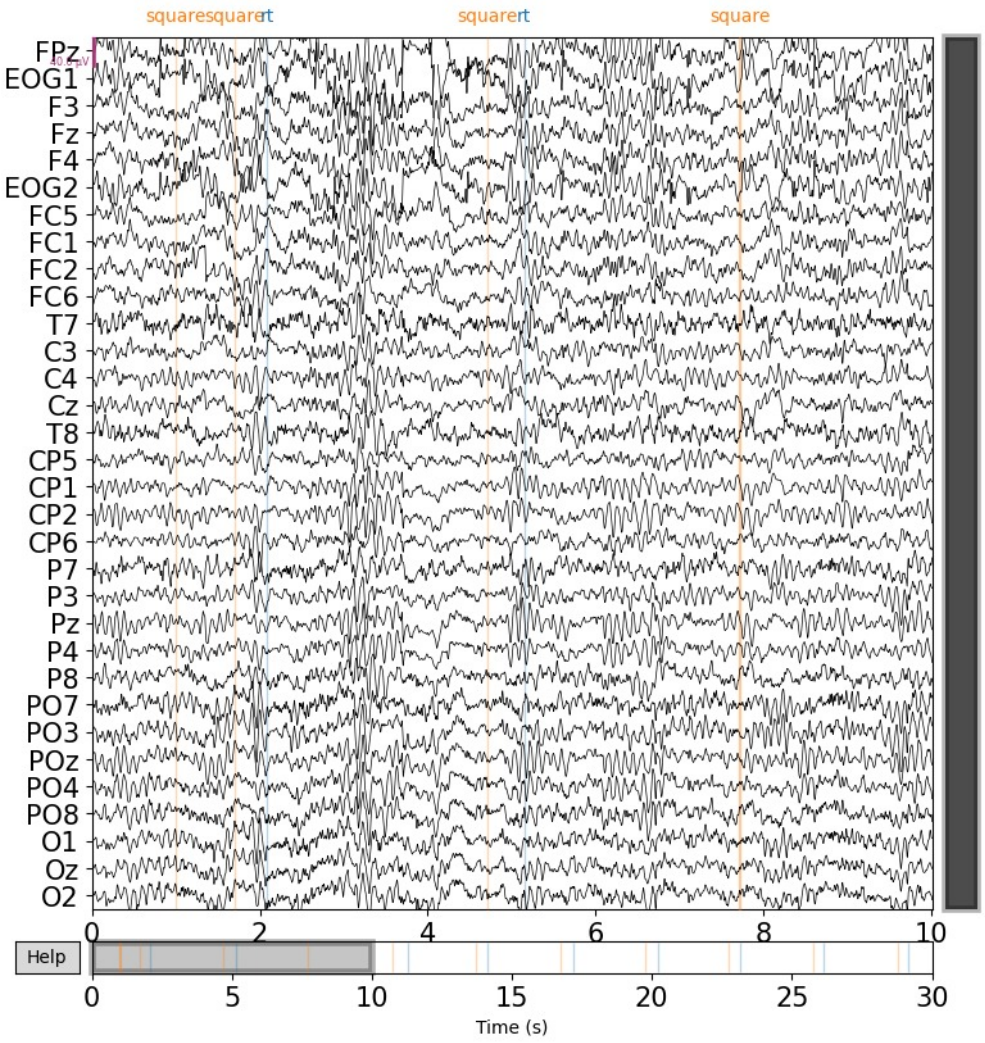


Figure 2.4: First 30 seconds of the final version of the signal used.

After these processing steps, the data was exported from MATLAB and imported into Python. Subsequently, it was imported and used with MNE-Python for the remaining processing, [13].

2.2 Applying ICA on EEG Data

We employed ICA on our processed data using the *infomax* method in MNE-Python. Then, the first 20 ICA components were printed. After some examination, the results were satisfactory. Although examining the identified components requires expertise, some components, such as eye artifacts, were identifiable.

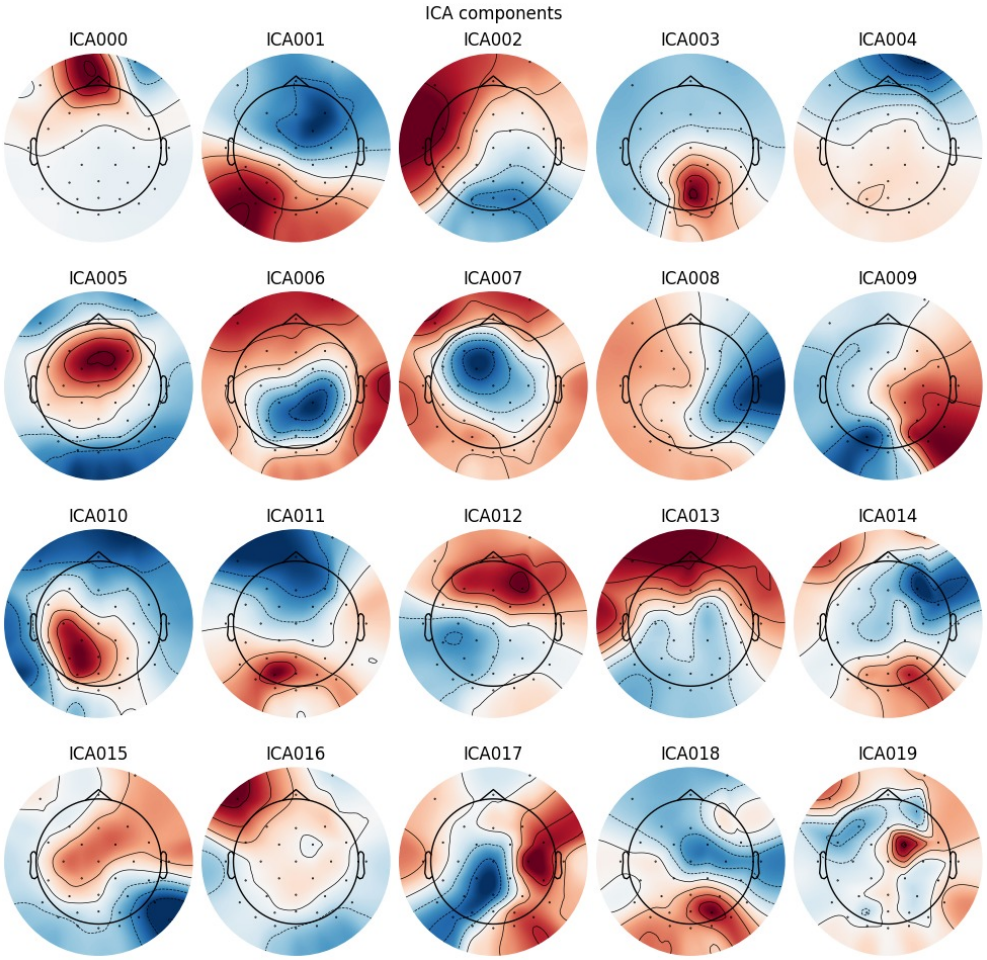


Figure 2.5: Resulting IC components

2.3 Applying PMF on EEG Data

Before proceeding to the results with the PMF algorithm, we shall discuss the concepts involved in the process.

2.3.1 Dipole Fitting

In the context of electromagnetic theory, a dipole refers to a pair of positive and negative electrical charges separated by a small distance. In neuroscience, a dipole is an idealization of synchronous neuronal activity; thus, brain activity is considered a combination of the activities of many dipoles. Dipole fitting hypothesizes dipoles within the brain and adjusts their positions, orientations, and magnitudes to best match the EEG data. The goal is to work backward to infer the characteristics of the dipoles that could have produced the measured EEG data. MNE-Python, a library for neurophysiological data analysis, provides tools for dipole fitting. We extract latent vector components from the EEG data by implementing sparse PMF and then use dipole fitting to map these latent vector components onto scalp maps.

Although the EEGLAB Toolbox allows for the easy implementation of dipole fitting on ICA components, the MNE Toolbox does not provide a straightforward way to fit dipoles to any component matrix (ICA or PMF) calculated for the whole signal.

As a workaround, the components were wrapped around an evoked object. For the BEM model used in dipole fitting, the "fsaverage" model from the FreeSurfer package was used, as surface data for the dataset was unavailable.

The actual comparison made with dipole fitting involves comparing the component map calculated by the algorithm with the component map that would be created by the simulated dipole.

2.3.2 Residual Variance

For a quantitative comparison, the EEGLAB's residal variance computation was used. The computation is as follows:

$$RV = \frac{\text{Var(PMF Component} - \text{Simulated Component)}}{\text{Var(PMF Component)}} \tag{2.1}$$

2.3.3 Hyperparameter Tuning

Parameter tuning performed on the factorization algorithm can be inspected in two aspects.

First, a grid search was performed for the learning rate and number of epochs separately, taking into consideration the number of sources with $RV < -0.25$. Selections were made accordingly.

Secondly, the variances of the dataset were adjusted to match the variances of a sample signal known to decompose well with its parameters.

This adjustment was allowed since changing the variances is a linear operation. The procedure is as follows:

If our generative model is as such:

$$\mathbf{Y} = \mathbf{HS} \tag{2.2}$$

where $\mathbf{Y}$ is our dataset and $\mathbf{H}$ are the component maps.

$\mathbf{Y}$ was multiplied by a diagonal matrix $\mathbf{D}$ to match each row to a target variance:

$$\mathbf{Z} = \mathbf{DY} = \mathbf{DHS} \tag{2.3}$$

Then the algorithm was applied to $\mathbf{Z}$. The obtained component map is actually an estimate of $\mathbf{DH}$. By multiplying it by a $\mathbf{D}^{-1}$ matrix from the left, we obtain the desired $\mathbf{H}$ matrix.

2.3.4 Determined Case

The PMF algorithm assumes the system to be either determined (number of sources = number of mixtures) or overdetermined (number of sources $\leq$ number of mixtures) [7]. Thus, we set the number of sources (r) in our algorithm to the maximum allowed, which is equal to the number of mixtures, 32. Although we tested and found that setting a higher number of sources yielded meaningful results, the algorithm has not been mathematically shown to work in the underdetermined case, so we did not proceed with that approach.

2.3.5 Conversion into ERP Data

In neuroscience literature, the process of dividing and obtaining the window for each trial of an experiment separately from the data is called epoching, and each trial is then called an *epoch*. The entire data sequence itself is called *continuous* data. Averaging epochs to get a single, shorter data set is called evoking, and the averaged data is referred to as *evoked* data. This is most commonly used for Event-Related Potential (ERP) analysis.

From our observations, we found that applying the PMF algorithm to evoked data provided significantly better source separation compared to applying PMF to continuous data.

Therefore, we epoched the data within a window of [-0.01, 2.5] seconds centered at each "square" event. Then, we obtained the evoked object by averaging the windows for all trials.

2.3.6 PMF Results and Analysis

For the decomposition with the Polytopic Matrix Factorization method, the Sparse PMF algorithm [7] was utilized. The 32 sources calculated by the algorithm are presented in Fig. 2.6, along with their residual variances.

We observed that after applying PMF to the evoked data, we obtained sources with drastically low residual variances.

For a detailed examination, the identified linear maps were filtered based on the condition $RV < 0.15$. Although this is an aggressive threshold for general purposes, we aimed to identify the most significant sources. A filtered version of Fig. 2.6 is presented in Fig. 2.7.

Upon examining these linear maps, we found that they correspond to meaningful dipoles previously studied with ICA. Additionally, some sources appeared to be in similar locations with low residual variances, which is less likely to be

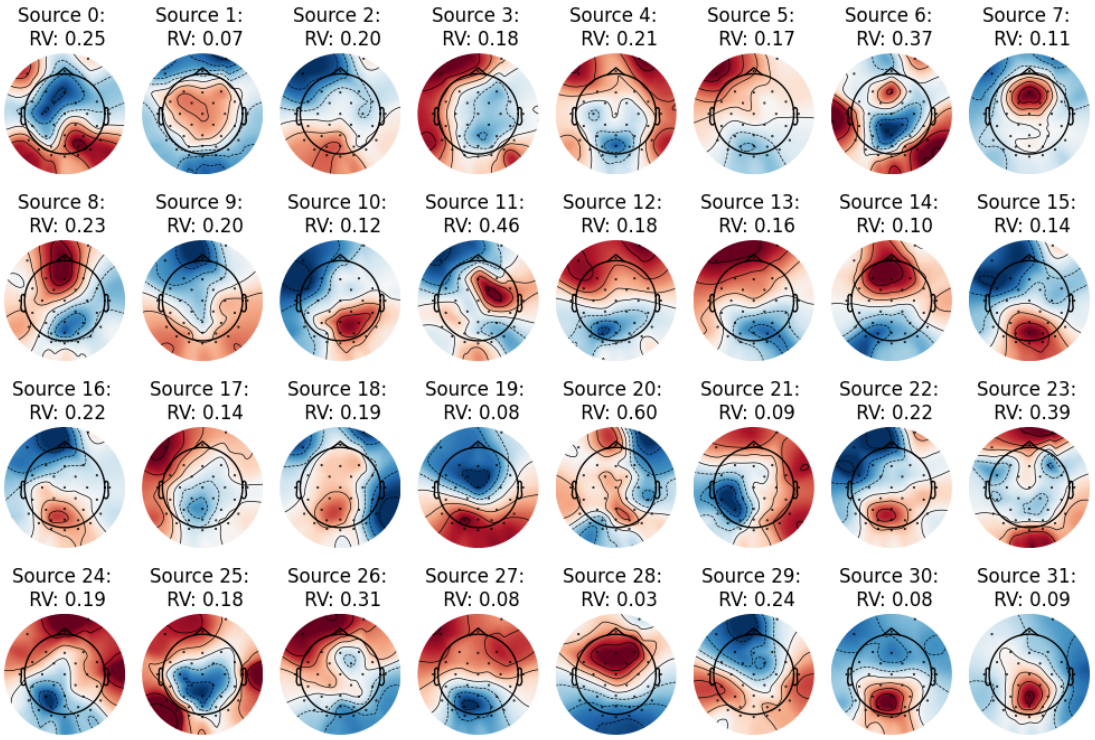


Figure 2.6: The brain topographic maps for the 32 sources discovered by PMF.

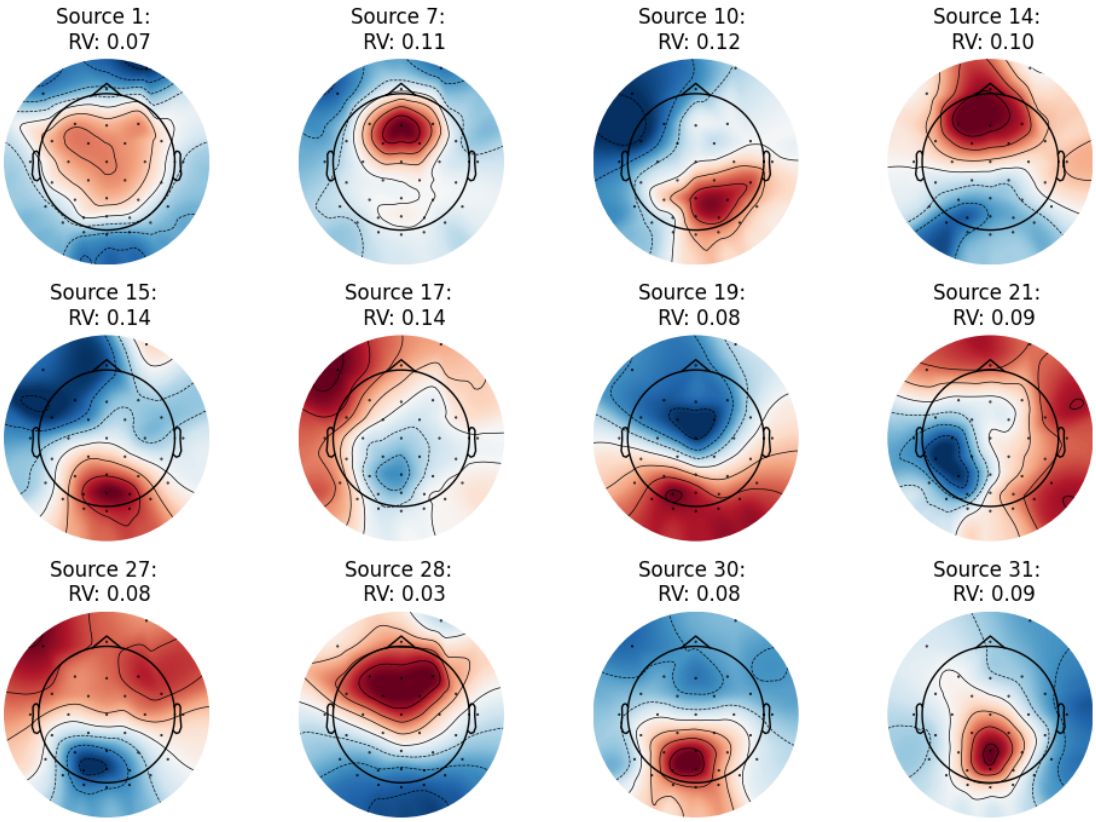


Figure 2.7: Filtered version of Fig. 2.6.

observed with ICA. This could be explained by the fact that the ICA algorithm

constrains the sources to be independent, while PMF does not. Thus, we can find sources that are close or similar to each other.

Furthermore, we shall examine the temporal source activities. The source signals are presented in Fig. 2.8.

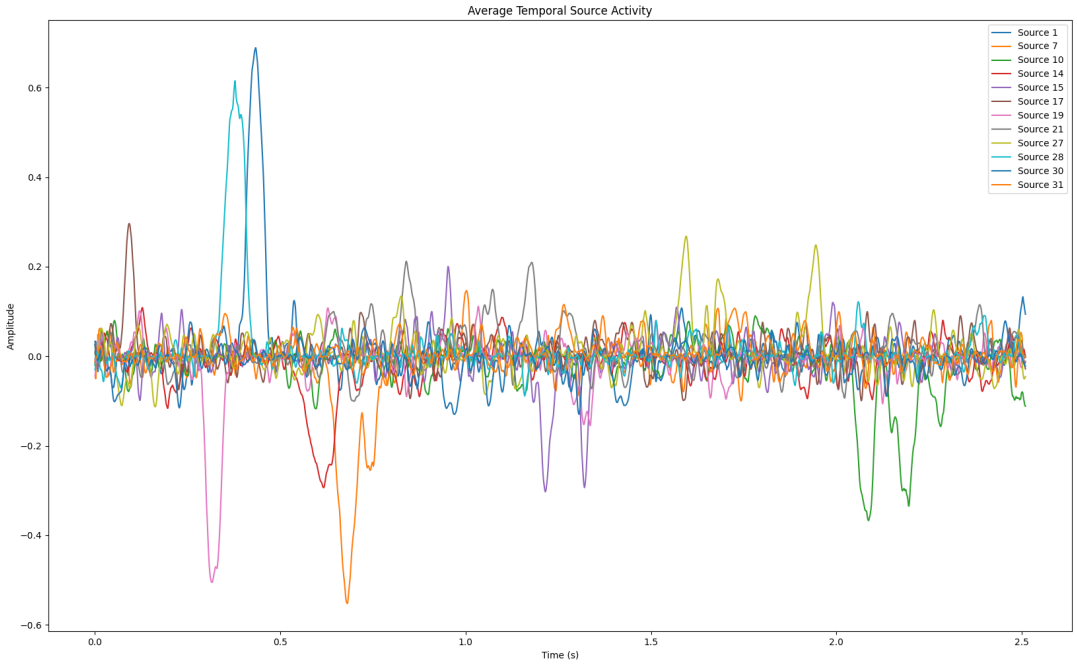


Figure 2.8: Temporal source activities of the filtered sources.

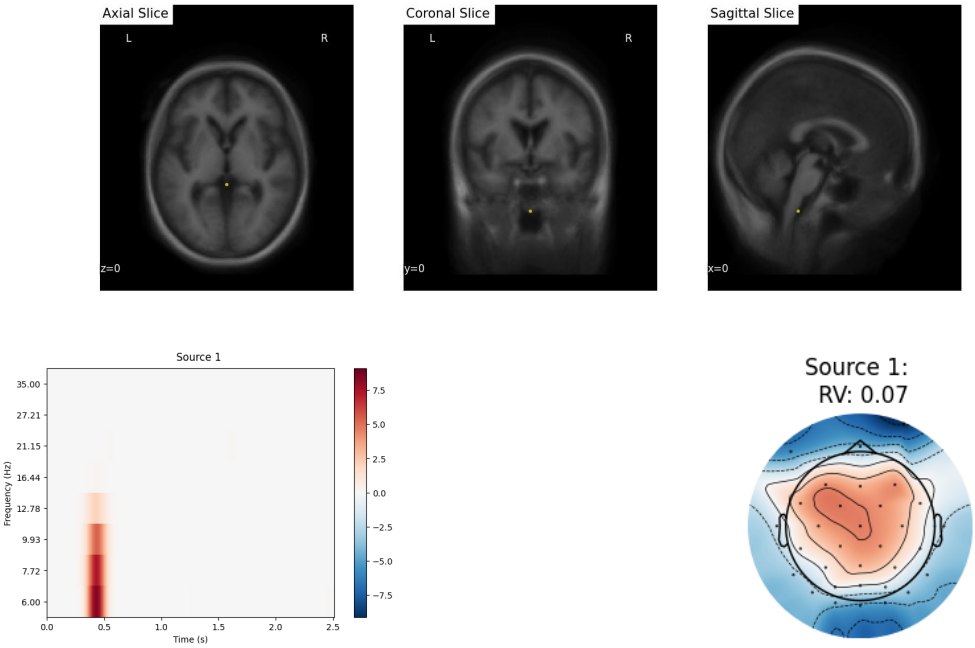


Figure 2.9: Best-fitted dipole location, time-frequency analysis and brain topographic map for Source 1

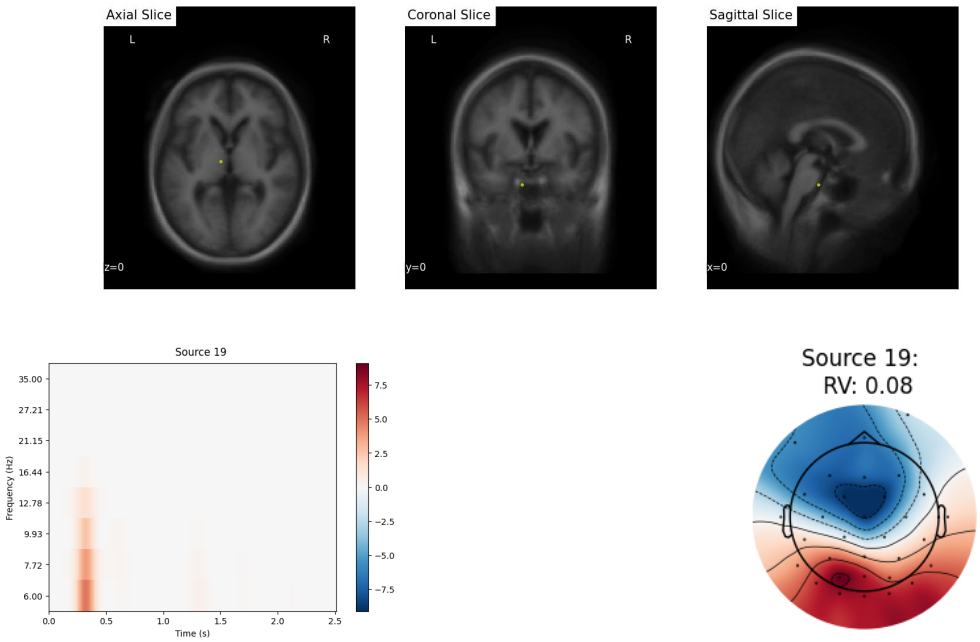


Figure 2.10: Best-fitted dipole location, time-frequency analysis and brain topographic map for Source 19

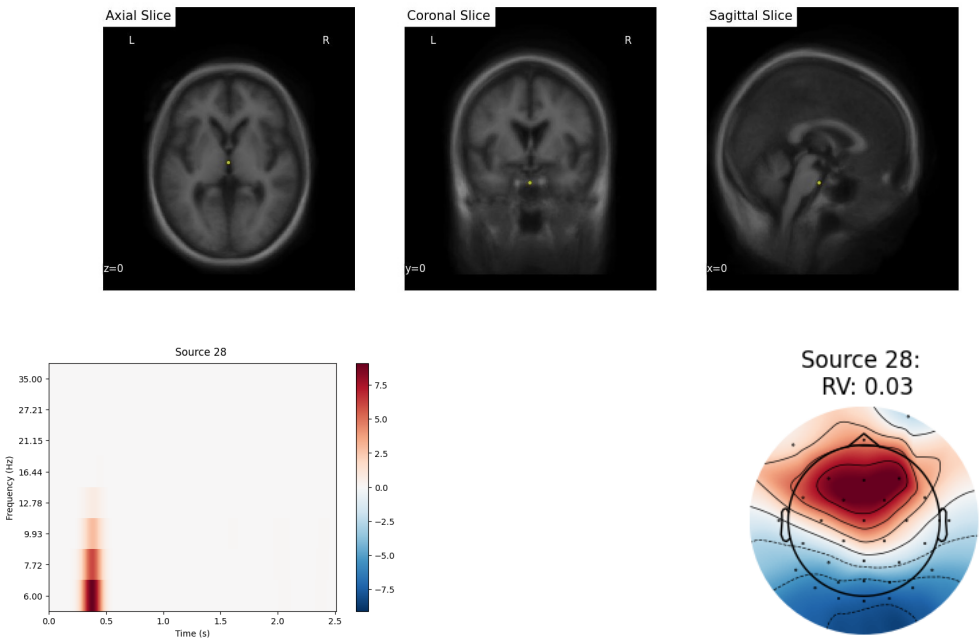


Figure 2.11: Best-fitted dipole location, time-frequency analysis and brain topographic map for Source 28

Examining Fig, 2.9, 2.10,2.11, we can conclude that PMF extracted some meaningful sources from the EEG data. Since the discovered sources has low residual variances, which means its dipolar and they have a response at around 0.4 seconds.

2.4 Reproducibility Problem for PMF

We also considered the problem of reproducing the source signals with different patients. As an example, we divided the whole data sequence into three parts and treated each part as data from different patients.

Following this, we applied the PMF algorithm to each of them and calculated the positions of the best-fitted dipoles. We displayed these positions on a brain plot with axial, sagittal, and coronal views to see if there were similarities between the simulated patients. Although the results appeared similar, we believe it was not sufficient to draw a definitive conclusion. The resulting plot can be seen in Fig. 2.12. In the plot, the sources were filtered based on RV to provide a clearer view.

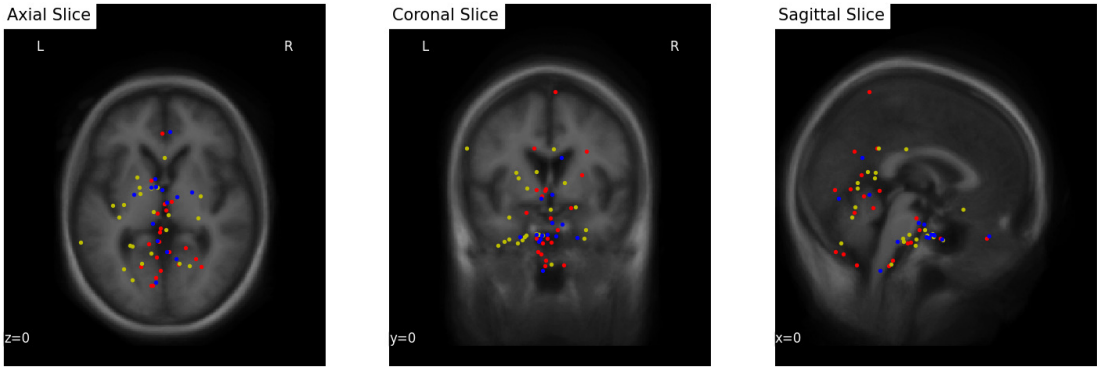


Figure 2.12: MNI coordinates of best-fitted dipoles (sources) projected onto middle slices of each view. For reproducibility concerns, the data has been divided into four sequences, each containing an equal number of visual events. Each sequence is regarded as an individual subject. Each color represents a different subject.

Chapter 3

CONCLUSIONS

3.1 Future Works

A processing pipeline utilizing PMF could be developed to differentiate between two groups by analyzing EEG sources during resting state or following a specific stimulus.

Further research could investigate the feasibility of employing the PMF as a feature extractor and embedding it within a deep learning framework.

3.2 Realistic Constraints

3.2.1 Social, Environmental and Economic Impact

This project can help build/discover subnetworks in the brain. Those subnetworks can potentially serve as diagnostic biomarkers. Medical doctors who are making the decision depending on the results can decide with a higher confidence. Also, the project could help reducing the extra tests needed, therefore, it could reduce the cost of medical service. Also, with using the PMF method on EEG data, the separation of

3.2.2 Cost Analysis

If the calculation of the salary would be based on the total of 7 credit hours, it would add up to 364 hours of total commitment. A PhD student could work on a similar project. If a PhD student's salary in the U.S. were taken as a basis, 19$/hr, the total salary to be paid would be around 7k$. If a fresh EE graduate in Turkey would be considered, it would correspond to a value of 3k$.

3.2.3 Standards

The project adheres to relevant engineering standards, particularly IEEE, IET, EU, and Turkish standards, and follows the engineering code of conduct.

APPENDICES

Appendix A

Given a finite number of points $\vec{x_1}, \vec{x_2}, ..., \vec{x_n}$ in a real vector space, a convex combination of these points is defined as:

$$\alpha_1\vec{x_1} + \alpha_2\vec{x_2} + ... + \alpha_n\vec{x_n}, \;\; n \in \mathbb{N} \setminus \{0\} \tag{1}$$

where $\alpha_i \in \mathbb{R}$, $\forall i \;\; \alpha_i \geq 0$ and;

$$\alpha_1 + \alpha_2 + ... + \alpha_n = 1, \;\; n \in \mathbb{N} \setminus \{0\}. \tag{2}$$

The geometric interpretation of (1) and (2) is that every convex combination of two points lies on the line segment between the points.[14]

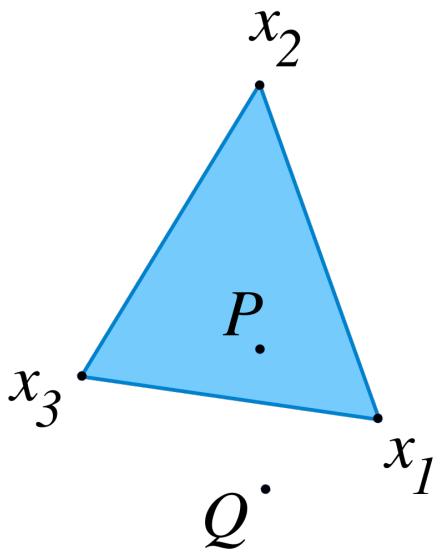


Figure 1: P is the convex combination of x_1, x_2, and x_3

BIBLIOGRAPHY

Bibliography

[1] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[2] X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, "Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications," *IEEE Signal Processing Magazine*, vol. 36, no. 2, pp. 59–80, 2019. DOI: 10.1109/MSP.2018.2877582.

[3] J. Shlens, "A tutorial on principal component analysis," *CoRR*, vol. abs/1404.1100, 2014. arXiv: 1404.1100. [Online]. Available: http://arxiv.org/abs/1404.1100.

[4] L. I. Smith, "A tutorial on principal components analysis," 2002. [Online]. Available: https://api.semanticscholar.org/CorpusID:60161425.

[5] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. Wiley-Interscience, 2001.

[6] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[7] G. Tatli and A. T. Erdogan, "Polytopic matrix factorization: Determinant maximization based criterion and identifiability," *IEEE Transactions on Signal Processing*, vol. 69, pp. 5431–5447, 2021, ISSN: 1941-0476. DOI: 10.1109/tsp.2021.3112918. [Online]. Available: http://dx.doi.org/10.1109/TSP.2021.3112918.

[8] R. Schachtner, G. Pöppel, and E. Lang, "Towards unique solutions of non-negative matrix factorization problems by a determinant criterion," *Digital Signal Processing*, vol. 21, no. 4, pp. 528–534, 2011.

[9] D. L. Donoho, "For most large underdetermined systems of equations, the minimal ℓ_1-norm near-solution approximates the sparsest near-solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 7, pp. 907–934, Jul. 2006.

[10] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[11] S. Makeig, M. Westerfield, T.-P. Jung, *et al.*, "Functionally independent components of the late positive event-related potential during visual spatial attention," *Journal of Neuroscience*, vol. 19, no. 7, pp. 2665–2680, 1999.

[12] A. Delorme and S. Makeig, "Eeglab: An open source toolbox for analysis of single-trial eeg dynamics including independent component analysis," *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.

[13] A. Gramfort, M. Luessi, E. Larson, *et al.*, "MEG and EEG data analysis with MNE-Python," *Frontiers in Neuroscience*, vol. 7, no. 267, pp. 1–13, 2013. DOI: 10.3389/fnins.2013.00267.

[14] R. T. Rockafellar, *Convex analysis* (Princeton Mathematical Series). Princeton University Press, Princeton, NJ, 1970, vol. No. 28, pp. xviii+451.