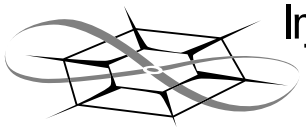
The University of Kansas



Information and Telecommunication Technology Center

Technical Report

Artifact Extraction from EEG Data Using Independent Component Analysis

Shadab Mozaffar
David W. Petr

ITTC-FY2003-TR-03050-02

December 2002

ARTIFACT EXTRACTION FROM EEG DATA USING INDEPENDENT COMPONENT ANALYSIS

ABSTRACT

Independent Component Analysis (ICA) has emerged as a novel and promising new tool for performing artifact corrections on EEG data. In this project, we used ICA to perform artifact correction on three types of artifacts namely, frontal (eye), Occipital (rear-head), and muscle. The EEG analyzing functions of the EEG toolbox available from the Salk Institude (www.salk.edu) were used for the ICA decomposition. We were able to successfully remove the eye and head artifacts from the EEG Data. The muscle artifacts could not be significantly reduced or removed due to the dispersion of the muscle artifact over the scalp.

Acknowledgements

I gratefully acknowledge the generous support and advice given to me by my advisor Dr. David Petr. I would also like to thank Mark G. Frei of FlintHill Scientific for his helpful notes on how to read and interpret EEG data. My special thanks also goes out all my friends in the Remote Sensing Lab for helping me get through the nights with a generous supply of unlimited coffee.

Contents

1. Introduction

1.1 What is EEG?

EEG stands for Electroencephalogram. It senses electrical impulses within the brain through electrodes placed on the scalp and records them on paper using an electroencephalograph. It is a recording of brain activity, which is the result of the activity of billions of neurons in the brain. EEG can help diagnose conditions such as seizure disorders, strokes, brain tumors, head trauma, and other physiological problems. The pattern of EEG activity changes with the level of a person's arousal. A relaxed person has many slow EEG waves whereas an excited person has many fast waves. A standardized system of electrode placement is the international 10-20 system.

A common problem with EEG data is contamination from muscle activity on the scalp. It is desirable to remove such artifacts to get a better picture of the internal workings of the brain.

In this project we will focus on removing such muscle artifacts from recorded EEG data using **Independent Component Analysis**. **Figure 1.1** shows a typical EEG recording.

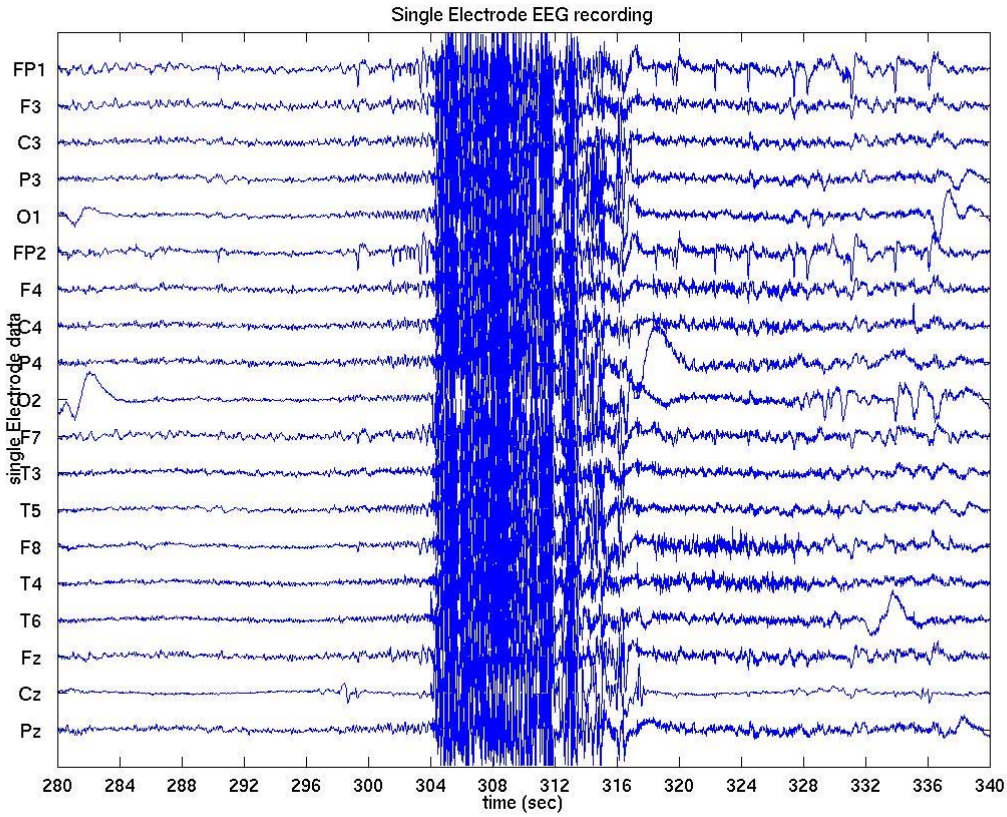


Figure 1.1 Typical EEG recording using an electroencephalograph

1.2 Standard EEG Electrode placement – the international 10-20 system

In order to perform consistent EEG recordings, the International 10-20 Electrode Placement System was developed to describe the locations of electrodes on the skull. Under this system, the EEG electrodes are placed on the scalp at 10 and 20 percent of a measured distance. For example, if a circumference measurement around the skull was approximately 55 cm, a base length of 10% or 5.5 cm and 20% or 11.0 cm would be used to determine electrode locations around the skull. The skull may be different from patient to patient but the percentage relationships remain the same.

Figure 1.2 shows a typical 10–20 electrode placement looking down on the skull. Each site has a letter and a number or another letter to identify the hemisphere location. The letters Fp, F, T, C, P, and O stand for Front polar, Frontal, Temporal, Central, Parietal and Occipital respectively. Even numbers (2, 4, 6, 8) refer to the right hemisphere whereas odd numbers (1, 3, 5, 7) refer to the left hemisphere. The z refers to an electrode placed on the midline. The smaller the number, the closer the position is to the midline.

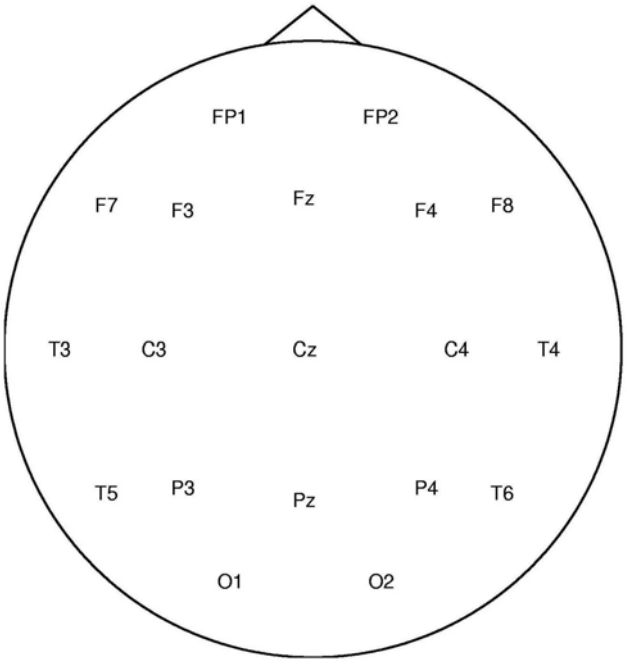


Figure 1.2 Typical electrode placements under the International 10 –20 system

1.3 Types of artifacts

Severe contamination of EEG activity by artifacts such as eye movements, blinks, head movements, muscle, and line noise create a problem for proper EEG interpretation and analysis. The three types of muscle artifacts studied in this project are:

1) **Eye artifacts** – they project mainly to the frontal side
2) **Rear head artifacts** – they project mainly to the occipital side
3) **Muscle artifacts** – dispersed throughout the brain

1.4 Problem Statement

We will attempt to remove artifacts from EEG data from two data sets – data set I and data set II – provided by **Flint Hills Scientific** located in Lawrence Kansas.

It seems desirable at first to place electrodes on scalp locations where muscle activity is localized (such as the frontal side for eye artifact) and then just subtract that from the EEG recording. However, this attempt leads to considerable loss in collected information and is a poor approach. A new and often preferable approach is to use **Independent Component Analysis** (also called ICA).

Independent Component Analysis separates a set of data into its *statistical independent* components. These components can then be studied and those identified as artifacts can be removed.

Chapter 2 deals with the introduction of Independent Component Analysis and how it can be applied to EEG data.

Chapter 3 gives the results obtained from using Independent Component Analysis on EEG data.

Chapter 4 ends with the conclusion. An **APPENDIX** is also included that gives a brief introduction to information theory and a flowchart of the algorithm used for artifact correction.

2. Method

2.1 Blind Signal Separation

Suppose you were to eavesdrop on a party where a lot of people were speaking simultaneously. What would you hear? It would probably be a mixture of all the conversations and not very audible. Granted that those speakers closer to the listener would dominate over those far away but on a whole it would not be very easy to separate all the speakers or even identify the number of speakers if there were a lot of people in the room.

This gives rise to an interesting question – can a person eavesdropping on the party separate all the speakers from the mixture of voices he or she observes without knowing anything about how many or where the people are in the room?

The scenario presented above is commonly referred to as the 'cocktail-party problem'. Such a problem of trying to separate a set of mixed signals without knowing anything about the number of original signals or how they are mixed together is called **Blind Signal Separation** or **Blind Source Separation**. The observer is, in a sense, *blind* to the number of original signals or to how they are mixed together. If there are *n*-speakers in the room and the voices are recorded by *m*-microphones, then we say that this is an *n*-by-*m* system.

For example: Let $n = m = 2$ with $s_1(t)$, $s_2(t)$ being the two original source signals and $x_1(t)$, $x_2(t)$ being the two recorded signals. Now if the recorded signals are linearly related to the source signals then we can say that:

$$x_1(t) = h_{11}s_1(t) + h_{12}s_2(t)$$

$$x_2(t) = h_{21}s_1(t) + h_{22}s_2(t)$$

Where h_{11}, h_{12}, h_{21}, h_{22} are the unknown (or blind) mixing coefficients that produce $x_1(t)$ and $x_2(t)$.

For our work with EEG signals we will model the system as an *n*-by-*m* system where *n*=*m*. At first, it seems questionable to assume that EEG data recorded from *m*-electrodes is made up of *exactly* *n*-statistically independent components since we ultimately cannot know the exact number of independent components embedded in the EEG data. This assumption is further discussed in **section 2.4**. For a large number of sources it is better to represent the equations in matrix form as:

$$\mathbf{X} = \mathbf{HS} \qquad -(2.1)$$

Where,

$$\mathbf{X} = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix}; \quad \mathbf{S} = \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_n(t) \end{bmatrix}; \quad \mathbf{H} = \begin{bmatrix} h_{11} & h_{12} \cdots h_{1n} \\ h_{21} & h_{22} \cdots h_{2n} \\ \vdots & \\ h_{n1} & h_{n2} \cdots h_{nn} \end{bmatrix}_{n=m} \qquad -(2.2)$$

If we knew the mixing parameters h_{ij}, $j = 1, \ldots, n$, and if the mixing was truly linear, this problem could be solved simply by inverting the mixing matrix, **H**. But the entire point of Blind Signal Separation is that we know *neither* h_{ij} or $s_n(t)$ which makes it a lot more complex.

The matrix representation in eq 2.1 suits our purpose since in EEG it is also not known (rather impossible to know) how the signals were mixed within the brain before they are picked up by the electrodes. The electrode readings are a mixture of useless muscle artifacts from within the scalp and useful EEG recordings from within the brain. An algorithm that separates the mixed electrode readings (consisting of muscle artifacts and useful EEG) where only the electrode potentials are known is needed.

One way to solve this problem would be to use the underlying statistical properties of $s_i(t)$ to approximate both h_{ij} and $s_i(t)$. **Independent Component Analysis (ICA)** attempts to do just that.

The remainder of this chapter deals with the introduction of ICA, its assumptions and ambiguities, and its application to EEG data. We conclude with the derivation of the Bell-Sejnowski information maximization (called infomax) algorithm for ICA and its Matlab implementation.

2.2 What is ICA anyway?

Independent Component Analysis, as the name implies, can be defined as the method of decomposing a set of multivariate data into its underlying statistically independent components. Hyvarinen and Oja [1] rigorously define ICA using the statistical "latent variables" model.

Under this model, we observe n random variables $x_1, x_2, \ldots, x_n$ etc which are linear combinations of n random latent variables $s_1, s_2, \ldots, s_n$ as:

$$x_i = a_{i1}s_1 + a_{i2}s_2 + \ldots + a_{in}s_n \quad \text{for all } i = 1, \ldots, n \qquad -(2.3)$$

where a_{ij}, $j = 1, \ldots, n$ are some real coefficients. By definition, the sources s_i are *statistically independent*. The "latent variables" are the sources, s_i, which are also called the independent components. They are called "latent" because they cannot directly be observed. Both the independent components, s_i, and the mixing coefficients, a_{ij}, are not known and must be determined (or estimated) using only the observed data x_i.

The multivariate data may be obtained from a number of sources such as:

(1) Audio signals such as the cocktail party problem introduced earlier, where ICA is used to separate individual speaker recordings from several microphones.
(2) Biomedical data like Electroencephalogram (EEG) and Magneto encephalogram (MEG) where the goal is to remove interfering muscle artifacts such as eye blinks and head movements.
(3) Images from satellites where ICA is used to extract certain features [3].
(4) Telecommunications, especially CDMA technology [3].

The ICA latent variables model is better represented in matrix form. If $\mathbf{S} = [s_1, s_2, s_3, \ldots, s_n]^T$ represents the original multivariate data that is transformed through some transformation matrix $\mathbf{H}$ producing $\mathbf{X}$ such that:

$$\mathbf{X} = \mathbf{HS} \qquad -(2.4)$$

Then ICA tries to identify an unmixing matrix **W** such that:

$$\mathbf{W} \approx \mathbf{H}^{-1} \qquad -(2.5)$$

so that the resulting matrix **Y** is:

$$\mathbf{Y} = \mathbf{WX} = \mathbf{W(HS)} = \mathbf{S'} \approx \mathbf{S} \quad (\text{since } \mathbf{W} \approx \mathbf{H}^{-1}) \qquad -(2.6)$$

As stated earlier, the only thing ICA demands is that the original signals s_1, s_2, … s_n, be at any time instant t *statistically independent* and the mixing of the sources be linear.

2.3 Assumptions for the ICA model

The following assumptions ensure that the ICA model estimates the independent components meaningfully. Actually the first assumption is the only true requirement which ICA demands. The other assumptions ensure that the estimated independent components are unique.

(1) The latent variables (or independent components) are statistically independent and the mixing is linear.

(2) There is no more than one gaussian signal among the latent variables and the latent variables have cumulative density function not much different from a *logistic sigmoid* (discussed further in **section 2.9.3**).

(3) The number of observed signals, *m*, is greater than or equal to the number of latent variables, *n* (i.e. $m \geq n$). If $n > m$, we come to a special category of Independent Component Analysis called ICA with over-complete bases [2]. In such a case the mixed signals do not have enough information to separate the independent components. There have been attempts to solve this particular problem but no rigorous proofs exist as of yet [2]. If $m > n$ then there is redundancy in the mixed signals. The ICA model works ideally when $n = m$.

(4) The mixing matrix is of full column rank, which means that the rows of the mixing matrix are linearly independent. If the mixing matrix is not of full rank then the mixed signals will be linear multiples of one another.

(5) The propagation delay of the mixing medium is negligible.

2.4 The ICA model applied to EEG Data

In case of EEG signals we have *m*-scalp electrodes picking up correlated brain signals where we would like to know what effectively independent brain sources produced these signals. The ICA model appears well suited for this scenario because it satisfies most of the model assumptions of **section 2.3**. We start with assuming that EEG data *can* be modeled as a collection of statistically independent brain signals. Assumption (5) is valid since volume conduction in the brain is effectively instantaneous [7] and assumption (2) is plausible [7]. In this project, we will attempt to separate the *m*-observed EEG signals into *n*-statistically independent components (thus satisfying assumption (3) and (4)). However, it is questionable to assume whether EEG data recorded from *m*-electrodes is made up of *exactly n*-statistically independent components since we ultimately cannot know the exact number of independent components embedded in the EEG data. Nonetheless, this assumption is usually enough to identify and separate artifacts that are concentrated in certain areas of the brain such as eye, temporal, and occipital artifacts [7]. The ICA model tends to have a more difficult time in separating artifacts that are more spaced out over the scalp such as muscle artifacts.

2.5 Ambiguities in the ICA solution

Since assumption (1) of the ICA model is the only requirement that is always strictly enforced, and because ICA is blind to the mixing matrix **H** and the source matrix **S**, the solution will always have the following ambiguities associated with it. For a more thorough discussion of the ambiguities in the ICA solution, the reader can refer to the book on ICA by Hyvarinen and Oja [1]

Ambiguities in the ICA solution:

(1) It is not possible to measure the energies of the independent components. This is because since both **H** and **S** are unknown, a constant multiplying **S** can be cancelled by the same constant dividing **H** and vice versa:

$$\mathbf{X} = (k)\mathbf{H}\left(\frac{1}{k}\right)\mathbf{S} = \left(\frac{1}{a}H\right)\left(\frac{a}{1}S\right) = \left(\frac{b}{1}H\right)\left(\frac{1}{b}S\right) = \cdots \qquad -(2.7)$$

Thus any combination of constants multiplying one matrix and dividing the other can make up the solution for **H** and **S**. To counteract for this, we fix the magnitudes of the independent components by assuming that all of them have $E\{s_i^2\}=1$. However, setting the magnitudes to 1 still does not help us when the constant = -1. We therefore conclude that multiplying an independent component by -1 will have no impact on the validity of the solution. Hence, we can 'flip' any number of independent components and our solution will still be valid.

(2) The order of the independent components may not be the same as the order of the original sources. For example, if $\mathbf{S} = [s_1, s_2, s_3]^T$, the final solution may be any permutation of $\mathbf{Y} = [s_1, s_2, s_3]^T$. The reasoning is the same as the previous ambiguity. Since both **H** and **S** are unknown, we can insert a permutation matrix, **P** and $\mathbf{P}^{-1}$, in the solution without changing it:

$$\mathbf{Y} = (\mathbf{W}\mathbf{P}^{-1})(\mathbf{P}\mathbf{X}) \qquad -(2.8)$$

where **PX** is the observed signals in another order and $\mathbf{W}\mathbf{P}^{-1}$ is a new unmixing matrix estimated by ICA.

2.6 Statistical Independence, Uncorrelatedness, and Whitening

We now very briefly look at some concepts from probability theory that are relevant to ICA.

2.6.1 Statistical Independence

The ICA algorithms work on the assumption that the original signals are statistically independent.

Definition: If s_1, s_2, s_3, …, s_n are n-random variables and their joint probability density function is equal to the product of their marginal probabilities, then s_1, s_2, s_3, …, s_n are defined as being statistically independent:

$$f(s_1, s_2, s_3, \ldots, s_n) = f(s_1) f(s_2) f(s_3) \ldots f(s_n) \qquad -(2.9)$$

2.6.2 Uncorrelatedness

Uncorrelatedness is defined as,

$$E\{s_1 s_2 s_3 \ldots s_n\} = E\{s_1\} E\{s_2\} E\{s_3\} \ldots E\{s_n\} \qquad -(2.10)$$

where $E\{.\}$ is the expectation operator.

Uncorrelatedness is weaker than statistical independence. Independence implies uncorrelatedness, but uncorrelatedness does not always imply independence. However, there is a class of random variables where uncorrelatedness always implies independence. This is when s_1, s_2, s_3, …, s_n are gaussian random variables. We will deal with gaussian independent components in more detail in **section 2.8**.

2.6.3 Whitening

An important preprocessing step before sending the data through the ICA algorithm is whitening. Whitening is weaker than statistical independence but slightly stronger than uncorrelatedness. Whiteness of a zero-mean random vector, e.g. **x**, means that its components are uncorrelated and their variance equals unity. That is, the covariance matrix of **x** equals the identity matrix **I**:

$$E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{I} \qquad -(2.11)$$

For our mixed data, **X**, whitening means that we linearly transform it by multiplying with a matrix (say **V)** such that the resulting matrix, **Z**, is white:

$$\mathbf{Z} = \mathbf{VX} = \mathbf{V(HS)} = \mathbf{H'S} \qquad -(2.12)$$

An important result of whitening is that the new mixing matrix, **H'**, is orthogonal (i.e. its inverse is equal to its transpose). Whitening alone does not ensure statistical independence of **X** but, as we will see in the next section, it plays an important step in the separation process. It is sometimes also called sphering.

2.6.4 Transformation of Probability Density Function

If **X** is transformed into **Y** by some transformation matrix, then the density of **Y** can be written in terms of the original variable **X** as:

$$p(\mathbf{Y}) = \frac{p(\mathbf{X})}{\left|J_y(\mathbf{X})\right|} \qquad -(2.13)$$

where $J_y(\mathbf{X})$ is the Jacobian of **Y** with respect to **X**. If **X** and **Y** are scalar-valued functions x and y, then the above relationship simplifies to:

$$p(y) = \frac{p(x)}{\left|\frac{\partial y}{\partial x}\right|} \qquad -(2.14)$$

2.7 Illustration of ICA with probability density functions

A graphical representation of how multivariate probability density functions change when latent variables are mixed together gives a better understanding of how ICA algorithms tackle the blind signal separation problem. This section is an abridged version of the similar topic dealt with in more detail in the Hyvarinen and Oja book [1]. The reader can refer to **section 7.5** (pp 155) in the book for a more thorough discussion on the topic.

Let us consider a source matrix consisting of two statistically independent and uniform random variables, $\mathbf{S} = [s_1, s_2]^T$. **Figure 2.1** shows their joint probability density function by plotting data points from their distribution. Note that the joint probability is uniform on a square (since it is just the product of their marginal densities).

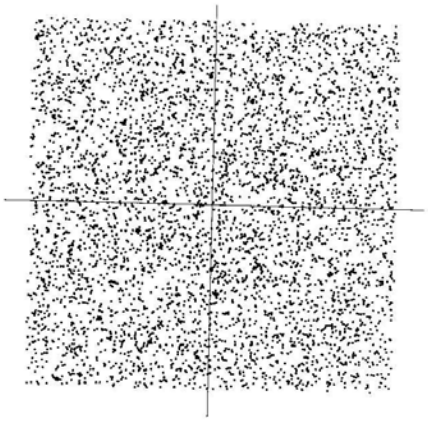


Figure 2.1 Joint distribution of two uniform random variables s_1 and s_2. Horizontal axis: s_1, Vertical axis: s_2

It is clear that s_1 and s_2 are statistically independent since knowing a value of s_1 at any point does not in any way help in guessing the value of s_2. Now lets mix these independent components with any 2-by-2 real valued mixing matrix, **H** $\left(eg \quad \mathbf{H} = \begin{bmatrix} 5 & 10 \\ 10 & 2 \end{bmatrix} \right)$. **Figure 2.2** shows the resulting density function. We see that mixing the independent components somewhat skews the density function.

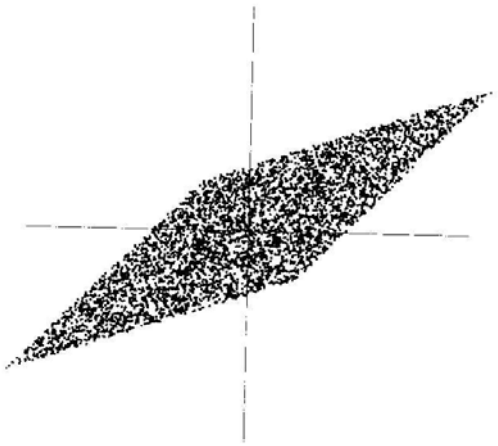


Figure 2.2 Joint distribution of the mixed random variables x_1 and x_2. Horizontal axis: x_1, Vertical axis: x_2

These two mixed variables, x_1 and x_2, are no longer statistically independent (since either variable completely determines the value if the other at its maximum or minimum value). We now whiten (**section 2.6.3**) the mixed variables with the result illustrated in **Figure 2.3**.

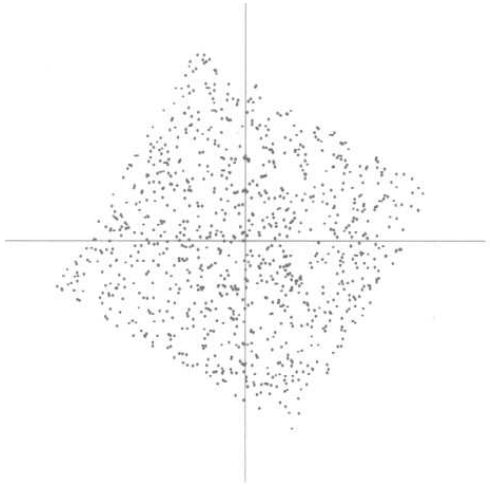


Figure 2.3 Joint distribution of the whitened mixtures of the independent components.

The whitened distribution looks just like our initial distribution (**Figure 2.1**) only rotated about the origin. It now simply becomes a matter of determining a single angle that can rotate the distribution back to its statistical independent form. Hence we see that whitening removes the 'skewness' from the observed data and simplifies the separation process considerably, which is why it is used as a useful preprocessing step in the ICA

algorithm. It can be shown that whitening simplifies the separation process by as much as 50% [1].

2.8 Gaussian latent variables will not separate

We saw in the previous section that whitening removes the 'skewness' from the data and leaves the ICA algorithm with only to determine the rotation required to bring the data back to its statistically independent from. But this process does not work very well when the original independent components have a gaussian probability distribution.

Figure 2.4 shows the joint probability distribution of two statistically independent gaussian random variables s_1 and s_2.

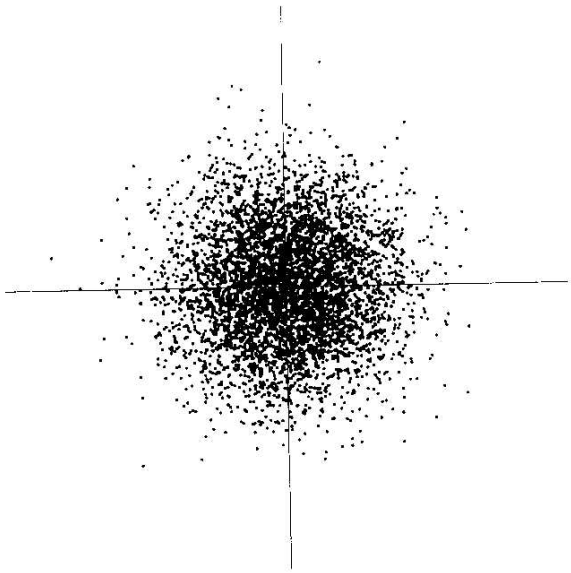


Figure 2.4 Joint distribution of two independent gaussian variables s_1 and s_2

If the data is whitened (which is always the case), then we can narrow the possibilities of the mixing matrix to orthogonal matrices (**section 2.6.3**). The resulting distribution when two independent gaussian random variables s_1 and s_2 are mixed with an orthogonal mixing matrix is shown in **Figure 2.5**. Note that both the original and mixed probability distributions are identical which shows that multiplying a gaussian distribution with an orthogonal matrix has no effect on the probability distribution. Since

the distribution is symmetrical, there is no way to 'rotate' the data back to its independent components. Hence, ICA cannot separate the mixed components when the latent variables are gaussian.

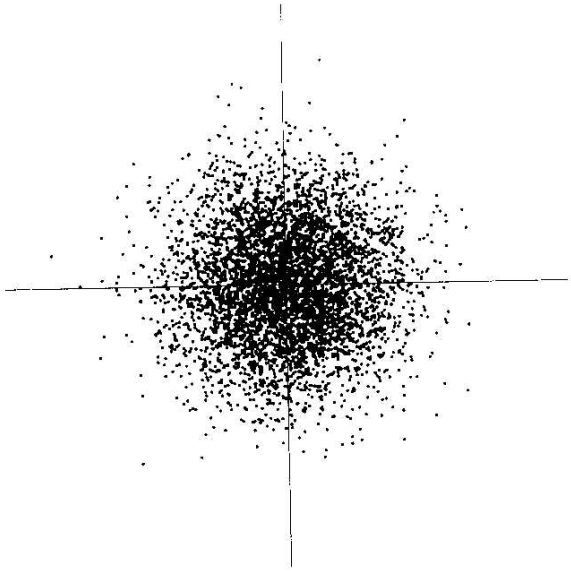


Figure 2.5 Joint distribution of two independent gaussian variables s_1 and s_2 after being mixed with an orthogonal mixing matrix

But what if some of the independent components are gaussian and some are non-gaussian? In this case, ICA can separate all the non-gaussian components but the gaussian components cannot be separated from one another [**1**]. Which is why our second assumption (**section 2.3**) demanded that there be no more than one gaussian random variable in **S**.

2.9 The ICA Algorithm

Before a satisfactory derivation of the ICA algorithm is attempted it is important for the reader to recall some important principles and relationships from information theory (refer to **APPENDIX–A** for an introduction to information theory if necessary).

2.9.1 Entropy

The entropy H(x) of x, if x is a continuous random variable with probability density function (pdf) $p(x)$ is defined as:

$$\mathrm{H}(x) \equiv -\int p(x) \log p(x) = -E\{\log [p(\mathrm{x})]\} \qquad -(2.15)$$

where E{.} is the expectation of x.

This definition also holds for multivariate data. Hence the entropy of $\mathbf{X} = [\mathrm{x}_1, \mathrm{x}_2, \mathrm{x}_3, \ldots \mathrm{x}_n]^T$ is:

$$\mathrm{H}(\mathbf{X}) = -E\{\log [p(\mathbf{X})]\} \qquad -(2.16)$$

Entropy can be thought of as a measure of randomness in a variable.

2.9.2 Mutual Information

If entropy is a measure of randomness in a variable, mutual information can be thought of as a measure of sameness in a variable. If $\mathbf{X} = [\mathrm{x}_1, \mathrm{x}_2, \mathrm{x}_3, \ldots \mathrm{x}_n]^T$ is a set of multivariate data then its joint entropy, H(**X**), and mutual information, I(**X**), are related by:

$$\mathrm{I}(\mathbf{X}) = \sum_{i=1}^{n} H(x_i) - \mathrm{H}(\mathbf{X}) \qquad -(2.17)$$

This means that the mutual information I(**X**) is equal to the difference of the sum of all the marginal entropies $\sum_{i=1}^{n} H(x_i)$ and the joint entropy H(**X**). This should make sense in an intuitive way.

Thus, if we *maximize* the joint entropy H(**X**) it will *minimize* the mutual information I(**X**). And if the mutual information of a multivariate data is minimized to zero, then all the individual elements will become statistically independent.

This is the reasoning used by Bell & Sejnowski [4] to derive their ICA algorithm, which is the algorithm we used for ICA in this project. Thus, the aim for obtaining statistical independence (and therefore performing ICA) then becomes to maximize the joint entropy H(**Y**) of **Y**.

2.9.3 The Bell Sejnowski infomax algorithm

We now present a brief derivation of the Bell Sejnowski Information Maximization algorithm [4]. We consider a simple case of a one-input one-output system to derive the ICA algorithm. The general multi-input multi-output system is similarly derived with *n*-dimensional matrices of vector-valued random variables in place of the scalar valued functions.

Consider a scalar-valued function x with a gaussian pdf $f_x(x)$ that passes through a transformation function $y = g(x)$ to produce the output with pdf $f_y(y)$ (**Figure 2.6**). This is analogous to our matrix operation:

$$\mathbf{Y} = \mathbf{WX} \qquad -(2.18)$$

For our work with EEG data we will take the transformation function y to be the logistic sigmoid function [7] defined as:

$$y = g(x) = \frac{1}{1+e^{-u}}, \quad u = wx+w_0 \qquad -(2.19)$$

where w = slope of y (also called the weight)

w_0 = bias weight to align the high density parts of the input with y (see **Figure 2.6**)

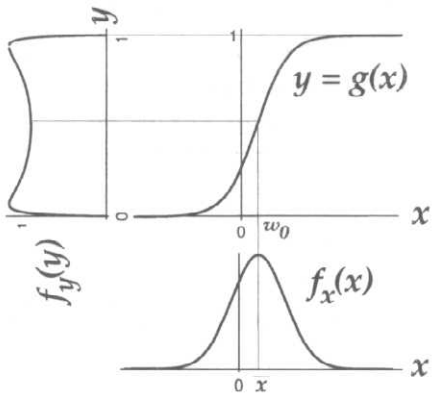


Figure 2.6 Transformation of the pdf, $f_x(x)$, of x when x is mixed with a sigmoid mixing function [4]

As discussed in **section 2.9.2** an increase in the joint entropy of the output, *H(y)*, means a decrease in its mutual information. The entropy of the output is maximized when we align the high density parts of pdf of x with the high sloping parts of the function *g(x)* (hence the need for the biasing weight w_0). The function *g(x)* is monotonically increasing (i.e. has a unique inverse) and thus the pdf of the output $f_y(y)$ can be written as a function of the pdf of the input $f_x(x)$ as:

$$f_y(y) = \frac{f_x(x)}{\left|\frac{\partial y}{\partial x}\right|} \qquad -(2.20)$$

The entropy of the output is given by,

$$H(y) = -E\{\ln f_y(y)\} = -\int_{-\infty}^{\infty} f_y(y) \ln f_y(y) dy \qquad -(2.21)$$

Substituting (2.20) into (2.21) gives,

$$H(y) = E\left(\ln\left|\frac{\partial y}{\partial x}\right|\right) - E\{\ln f_x(x)\} \qquad -(2.22)$$

We now would like to maximize $H(y)$ of eq. 2.22 for statistical independence. Looking at the right hand side we see that the function x is fixed and the only variable we can change is y. Or more preciously, the slope, w, of y. Hence we take the partial of $H(y)$ with respect to w. The second term in eq 2.22 does not depend on w and therefore can be ignored. The change in slope, Δw, necessary for maximum change in entropy is then:

$$\Delta w \propto \frac{\partial H(y)}{\partial w} = \frac{\partial}{\partial w} E\left(\ln\left|\frac{\partial y}{\partial x}\right| \right) \qquad -(2.23)$$

We now come to an important step. We would like to compute the derivative, but we cannot compute the expectation. Hence, we make the stochastic gradient approximation: $E\left(\ln\left|\frac{\partial y}{\partial x}\right| \right) \approx \ln\left|\frac{\partial y}{\partial x}\right|$ to get rid of the expectation [4]. The equation then simplifies to:

$$\Delta w \propto \frac{\partial H(y)}{\partial w} = \frac{\partial}{\partial w}\left(\ln\left|\frac{\partial y}{\partial x}\right| \right) = \left(\frac{\partial y}{\partial x}\right)^{-1} \frac{\partial}{\partial w}\left(\frac{\partial y}{\partial x}\right) \qquad -(2.24)$$

The above equation is the general form of the weight change rule for any transformation function y. For the logistic sigmoid function (eq 2.19), the terms in eq 2.24 are evaluated as:

$$\frac{\partial y}{\partial x} = wy(1-y) \qquad -(2.25)$$

$$\frac{\partial}{\partial w}\left(\frac{\partial y}{\partial x}\right) = y(1-y)(1+wx(1-2y)) \qquad -(2.26)$$

Substituting the above equations into eq 2.24 gives the weight update rule for y = logistic sigmoid function:

$$\Delta w \propto w^{-1} + (1-2y)x \qquad -(2.27)$$

Similarly, the bias weight update, Δw_0, can be evaluated as:

$$\Delta w_0 \propto 1 - 2y \qquad -(2.28)$$

Following similar steps we can derive the learning rules for multivariate data [4] for a sigmoid function:

$$\Delta \mathbf{W} \propto [\mathbf{W}^{\mathrm{T}}]^{-1} + (1\text{-}2\mathbf{y})\mathbf{x}^{\mathrm{T}} \qquad -(2.29)$$

$$\Delta \mathbf{w}_0 \propto \mathbf{1} - 2\mathbf{y} \qquad -(2.30)$$

2.10 The Matlab Implementation

Equations 2.29 and 2.30 give the learning rules for updating the weights to perform ICA. Implementing them directly into Matlab will involve performing the inverse function, which is computationally very intensive. We therefore modify eq 2.29 by multiplying it by $\mathbf{W}^T\mathbf{W}$ (this does not change anything since $\mathbf{W}$ is orthogonal):

$$\Delta\mathbf{W} \propto \frac{\partial H(y)}{\partial W}\mathbf{W}^T\mathbf{W}$$

$$\Rightarrow \quad \Delta\mathbf{W} \propto ([\mathbf{W}^T]^{-1} + (\mathbf{1}\text{-}2\mathbf{y})\,\mathbf{x}^T)\,\mathbf{W}^T\mathbf{W}$$

$$\Rightarrow \quad \Delta\mathbf{W} \propto (\mathbf{I} + (\mathbf{1}\text{-}2\mathbf{y})\,\mathbf{x}^T\mathbf{W}^T)\,\mathbf{W}$$

$$\Rightarrow \quad \Delta\mathbf{W} \propto (\mathbf{I} + (\mathbf{1}\text{-}2\mathbf{y})\mathbf{u}^T)\,\mathbf{W}, \quad \text{where } \mathbf{u} = \mathbf{xW} \qquad -(2.31)$$

The bias weight update rule remains the same:

$$\Delta\mathbf{w}_0 \propto \mathbf{1} - 2\mathbf{y} \qquad -(2.32)$$

The proportionality constant in eq 2.31 and 2.32 is called the learning rate (**lrate**).

In summary, the following two weight update rules are used to perform ICA in Matlab:

$$\mathbf{W}_{new} = \mathbf{W}_{old} + \mathbf{lrate}\,[(\mathbf{I} + (\mathbf{1}\text{-}2\mathbf{y})\mathbf{u}^T)\mathbf{W}] \qquad -(2.33)$$

$$\mathbf{w}_{0new} = \mathbf{w}_{0old} + \mathbf{lrate}\,[\mathbf{1} - 2\mathbf{y}] \qquad -(2.34)$$

where:

lrate = learning rate

W = weights matrix

$\mathbf{w}_0$ = bias weight

I = identity matrix

$\mathbf{y}$ = logistic sigmoid = $\dfrac{1}{1+e^{-\mathbf{u}}}$

$\mathbf{u} = \mathbf{W}\times\text{data} + \mathbf{w}_0$

2.11 The EEG toolbox

We used the EEG toolbox for Matlab available from the Salk Institute (http://www.sccn.ucsd.edu/~scott/ica-download-form.html) to perform artifact corrections on the EEG data. The Matlab function runica.m incorporates the Bell Sejnowksi ICA algorithm derived in **section 2.9.3**. Its flowchart is given in **APPENDIX–B**. The EEG recordings are read as a matrix with the different electrode recordings in rows format. The strategy we used for artifact correction is as follows:

1) Plot the data using eegplot.m
2) Perform ICA using runica.m
3) Study the relative strengths of the independent components projected back onto the scalp using topoplot.m. Make decisions on which independent components might be artifacts using generally accepted heuristics [7].
4) Remove selected artifacts using icaproj.m. Plot the corrected EEG data.

2.11.1 Visualizing EEG artifacts

The function topoplot.m is used to study the relative strengths of the independent components projected back onto the scalp. Assuming **W** is the weights matrix obtained after running the ICA algorithm (runica.m), the columns of the inverse matrix, inv(**W**), give the relative projection strengths of the respective components at each of the scalp sensors. These plots help in visualizing the components' physiological origins. Knowing certain properties of different artifacts help in deciding which components can be classified as probable artifacts:

1) Eye movements and eye blinks project mainly to frontal sites (near electrodes FP_1 and FP_2)
2) Temporal muscle activity should project to the temporal sites (near T_3 and T_4)
3) Occipital (rear head) movements project to the back (electrodes O_1 and O_2)

An example of a topographical plot (viewed from the top of the head looking down) of an independent component obtained from topolot.m is shown below. Regions of high

magnitude denote concentrated EEG activity. It would tempting to classify the independent component in **Figure 2.6** as an artifact but a closer look reveals that the EEG activity is dispersed towards both the frontal and temporal regions. Therefore it cannot be called an artifact with certainty.

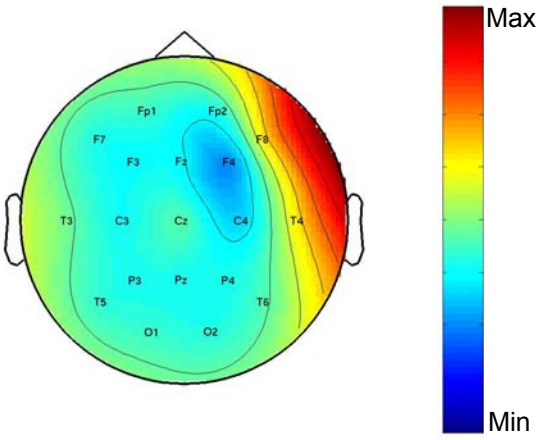


Figure 2.6 Topographical view of the brain showing the intensity of EEG recordings at a time instant

At this point it is important to recall the first ambiguity of the ICA solution discussed in **section 2.5**. Since we can multiply a component by –1 without changing the solution, both strong red and strong blue regions demonstrate strong EEG activity.

2.11.2 Performing Artifact Correction

Artifact correction simply means removing a selected independent component from the observed EEG data. For the observed EEG data **X,** and the evaluated weights matrix **W**, the corrected EEG data, **clean_data**, is given by:

$$\mathbf{clean\_data} = \mathbf{W}_{inv}(:,\mathbf{a}) \times \mathbf{ica}(\mathbf{a},:) \qquad -(2.35)$$

where:

$\mathbf{W}_{inv}$ = inverse of **W**

a = vector of independent components to keep

ica = independent components. Obtained from **W**×**X**.

For example, let $\mathbf{H} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix}$ be the blind mixing matrix on the blind sources

$\mathbf{S} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}$ producing the recorded EEG data $\mathbf{X} = \begin{bmatrix} s_1 + 2s_2 + 3s_3 \\ 3s_1 + s_2 + 2s_3 \\ 2s_1 + 3s_2 + s_3 \end{bmatrix}$.

Now, suppose we wanted to remove the independent component, s_2, from the observed EEG data **X**, after evaluating **W** from runica.m. Then **a** = [1 3] and **clean_data** is,

$$\mathbf{clean\_data} = \mathbf{W}_{inv}(:,[1\ 3]) \times \mathbf{ica}([1\ 3],:)$$

$$= \mathbf{H}(:,[1\ 3]) \times \begin{bmatrix} s_1 \\ s_3 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 3 \\ 3 & 2 \\ 2 & 1 \end{bmatrix} \times \begin{bmatrix} s_1 \\ s_3 \end{bmatrix}$$

$$= \begin{bmatrix} s_1 + 3s3 \\ 3s_1 + 2s_3 \\ 2s_1 + s_3 \end{bmatrix} \text{ and } s_2 \text{ is removed.}$$

The function icaproj.m performs artifact correction in the EEG toolbox.

3. Results

3.1.1 Data Set I – EEG Data

Data Set I (**Figure 3.1**) contains 60 seconds of data with sampling frequency Fs = 239.75 Hz. There are 26 channels of data (although only channels 1-21 are useful scalp recordings). The data was collected from electrodes placed on the scalp at standard locations using the international 10-20 system. The EEG data is on the next page (plotted using eegplot.m)

This data contains a seizure onset around t = 299.5 evident on T3-T5 channel with the appearance of rhythmic waves. Muscle artifacts appear on all channels from about t = 305-315 and continue longer on some (e.g., T4, F8). Occipital (rear) head movement artifacts occur around t = 280-283, eye blink artifacts (Fp1, Fp2 and surrounding) around t = 290 and 298.

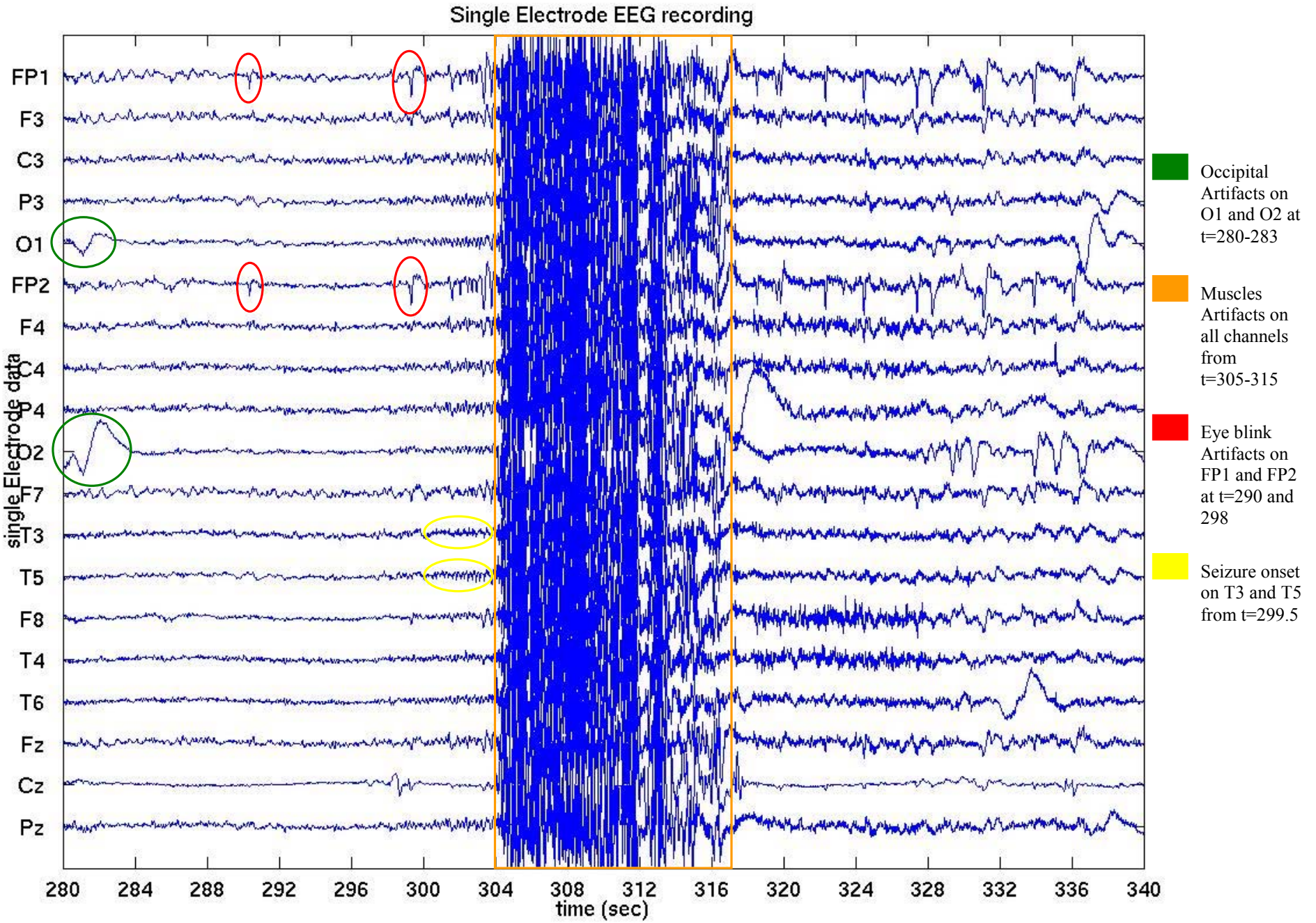


Figure 3.1 EEG Data from Data Set I

3.1.2 Data Set I – Independent Components

We ran the data through runica.m in the EEG toolbox. The resulting independent components are shown below.

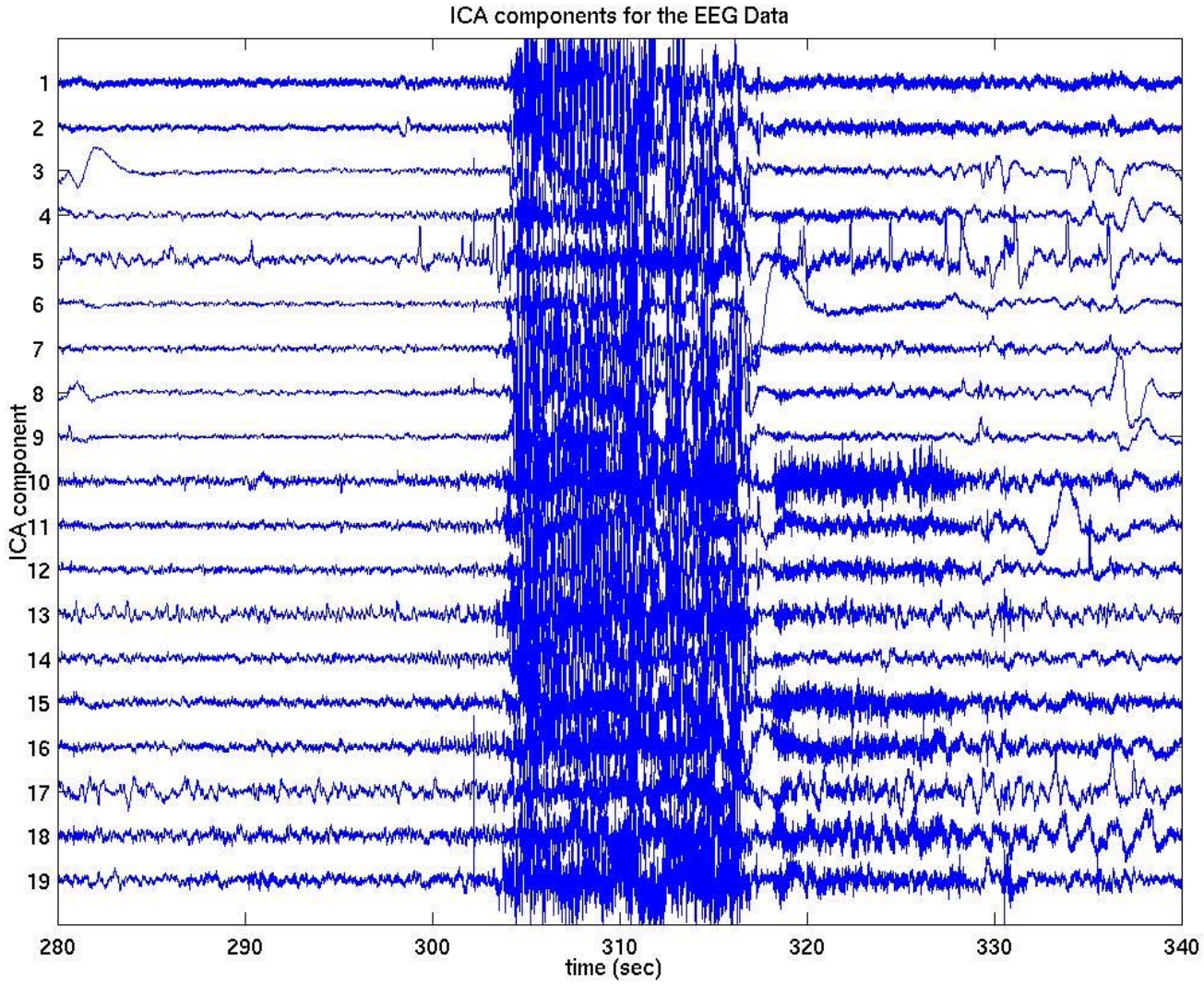


Figure 3.2 Independent Components of Data Set I

3.1.3 Data Set I – Topographical Projections

The Matlab function topoplot.m was then used for plotting the topographical projections of the independent components on the next page. Following the guidelines set forth in **section 2.11.1** about selecting artifacts, we identified ICA components 3, 5, and 8 as the right occipital, eye (frontal), and left occipital artifacts (**Figure 3.3(a),(b)**). Note that both high red and high blue regions are artifacts due to the sign ambiguity discussed in **section 2.5**.

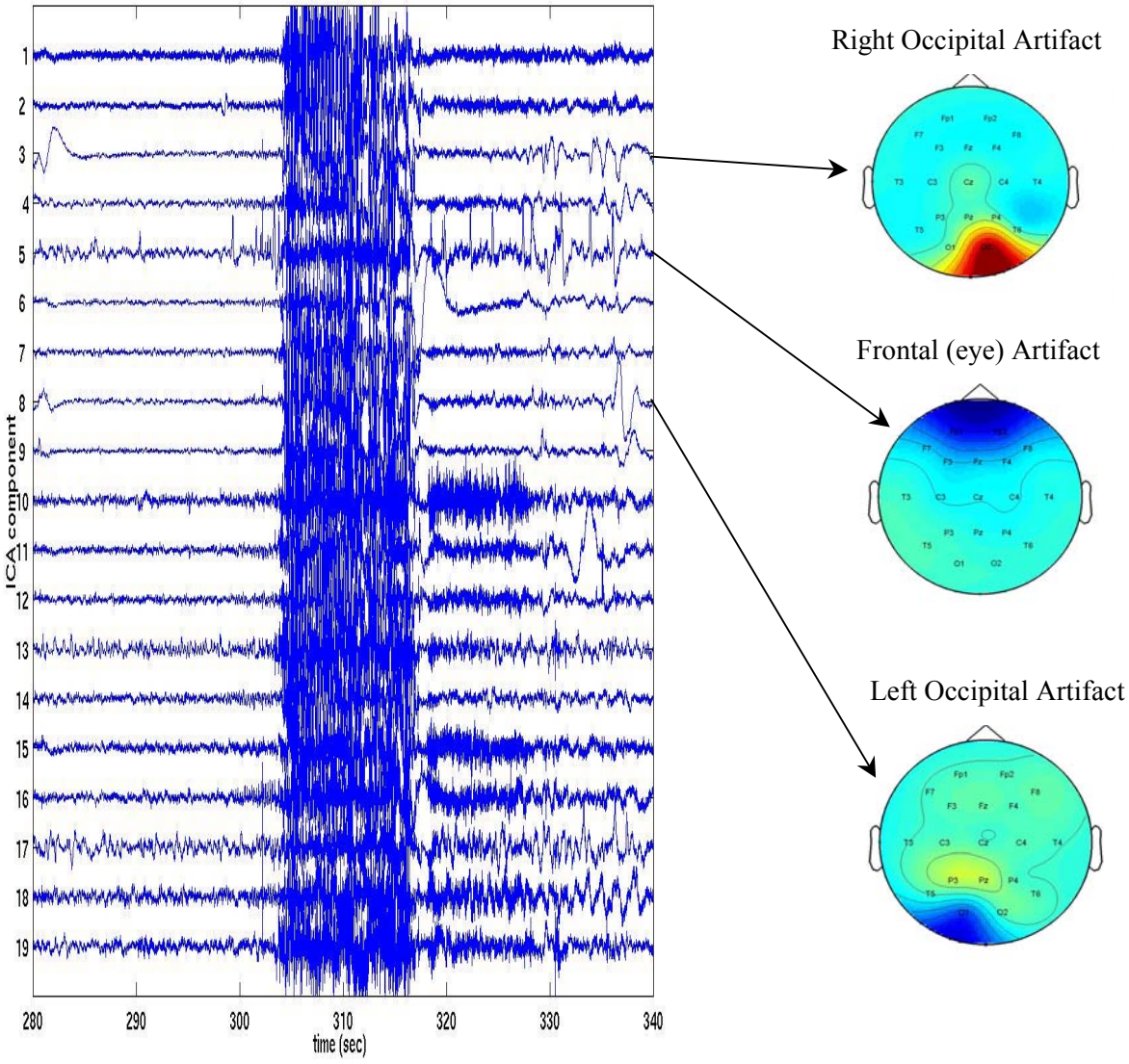


Figure 3.3(a) Independent Components with their respective topographical projections of Data Set I

Topographical Projections of the ICA components of Data Set I using topoplot.m
(Components 3, 5, and 8 are muscle artifacts)

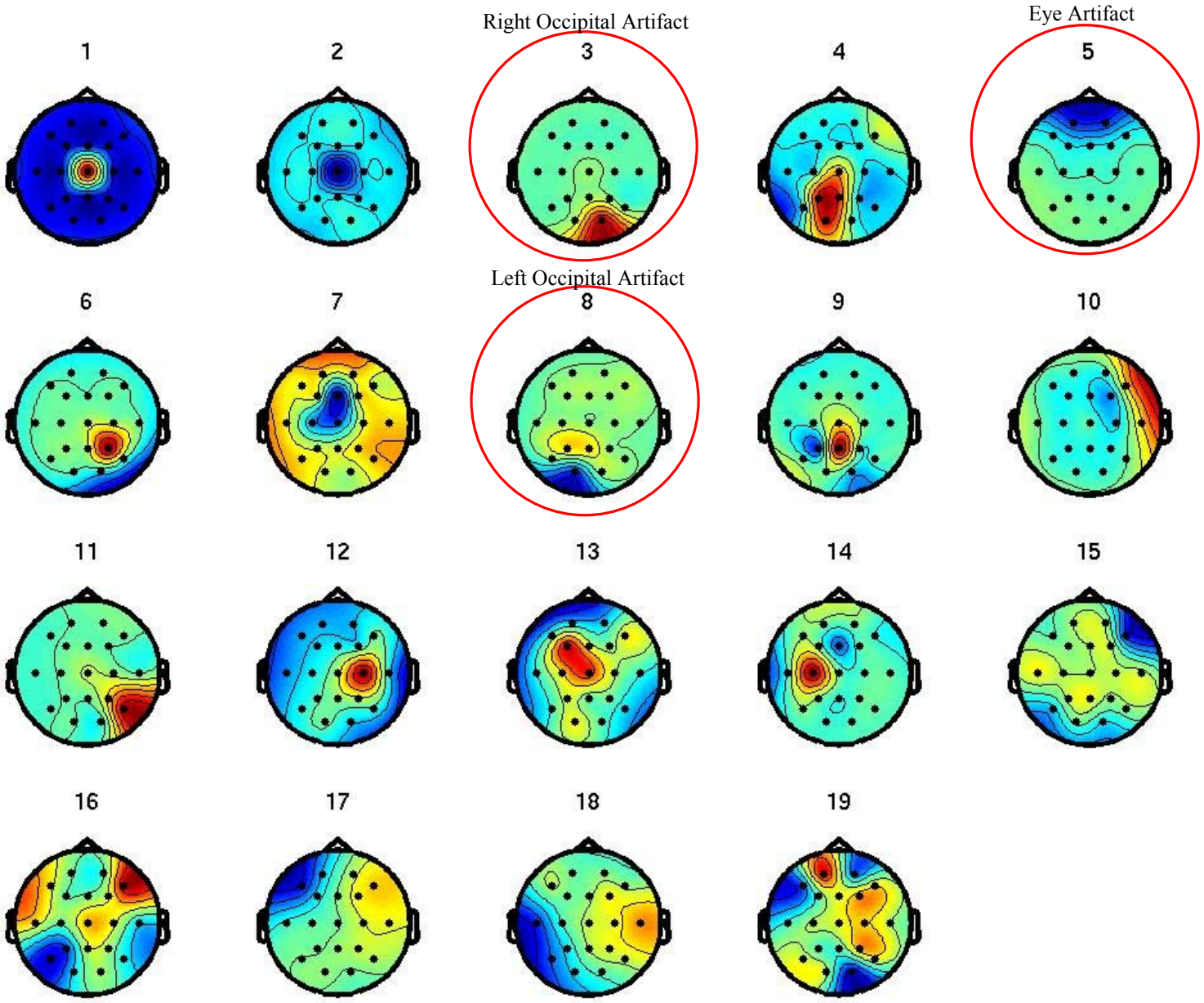


Figure 3.3(b) Topographical Projections of the ICA components of Data Set I

3.1.4 Data Set I – Corrected EEG Data

The 3 selected artifacts were then removed from the EEG data using icaproj.m. The resulting artifact corrected EEG data is shown on the next page (**Figure 3.4**). A comparison with the original EEG data (**Figure 3.1**) clearly shows that the identified muscle artifacts have been greatly reduced. On of the artifacts of interest that could not be removed were the muscle artifacts on all channels from t = 305 – 315. We tried running the ICA algorithm with different learning rates and with different data lengths but we were not able to isolate them properly.

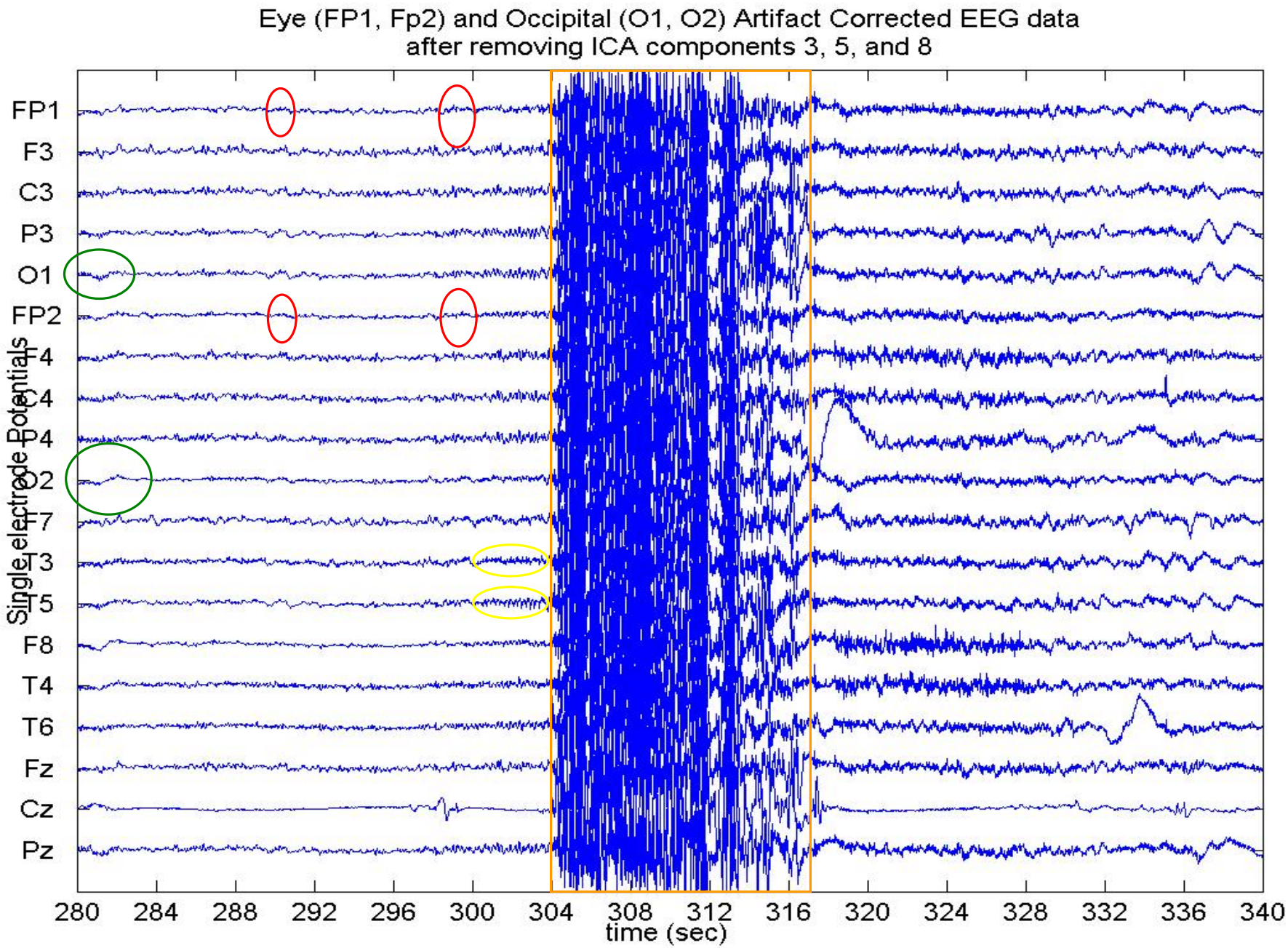


Figure 3.4 Corrected EEG Data of Data Set I (compare with original data in figure 3.1)

3.2.1 Data Set II – EEG Data

Data set II is a 10-minute 22-channel EEG Data. There are three seizure onsets at t = 78, 296, and 582 respectively. Due to the large size of the data set, we divided the data into three sections (t =1-90, t = 90-312, & t = 312-592) and analyzed them separately.

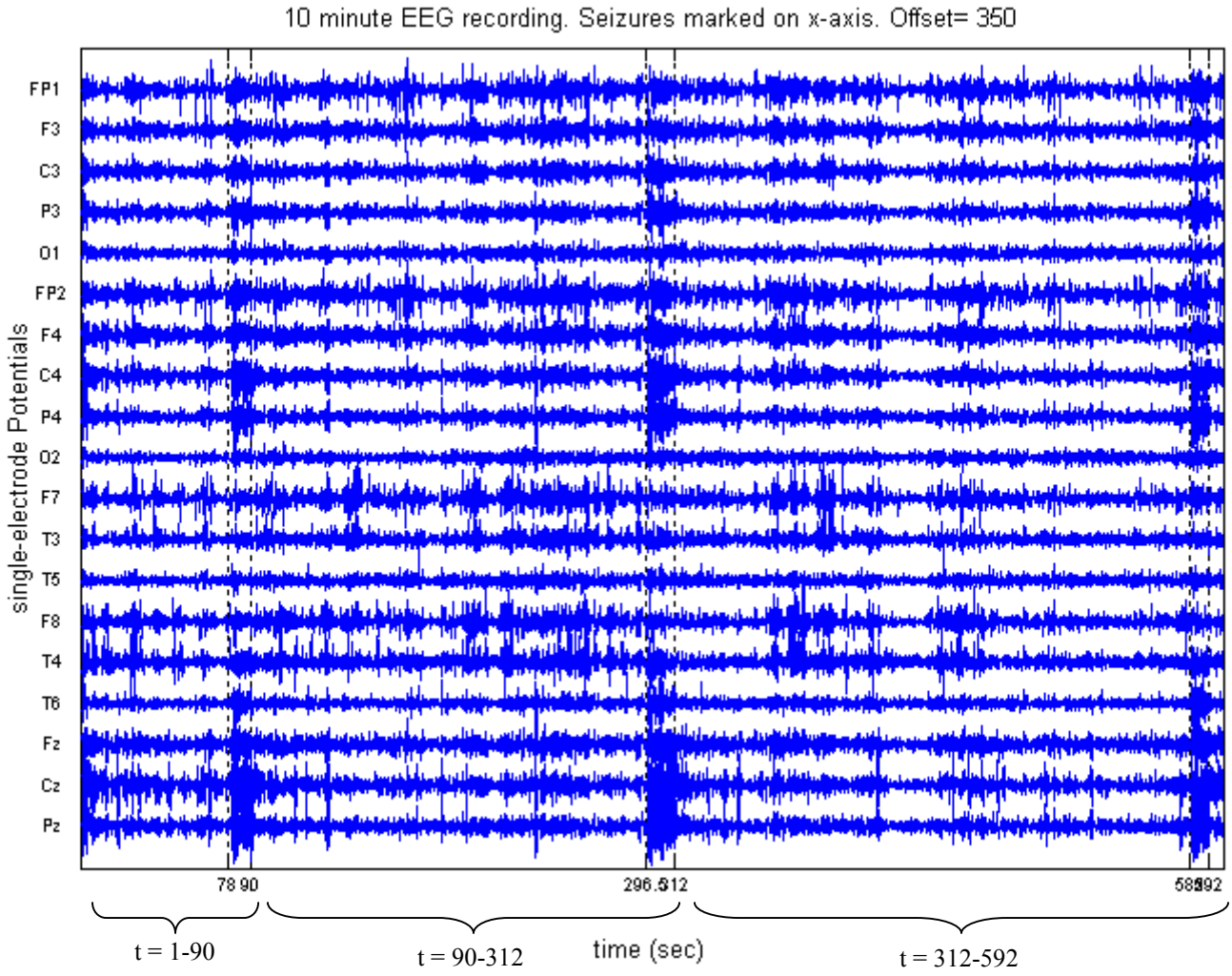


Figure 3.5 Corrupted EEG Data Set II

3.2.2 Data Set II – Independent Components/ Topographical projections

The three sections were separately run through the ICA algorithm and their respective independent components and the projections of the identified artifacts are shown below.

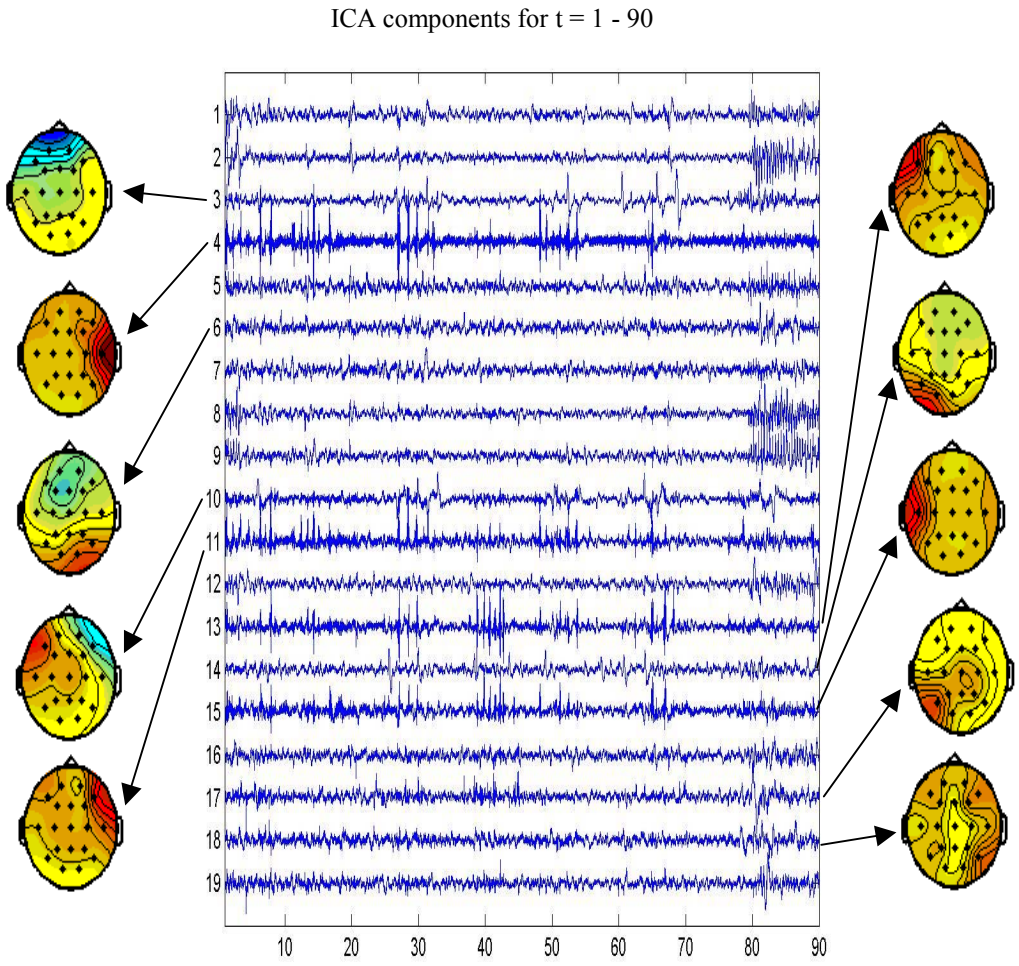


Figure 3.6 ICA components and projections of selected artifacts for t=1-90

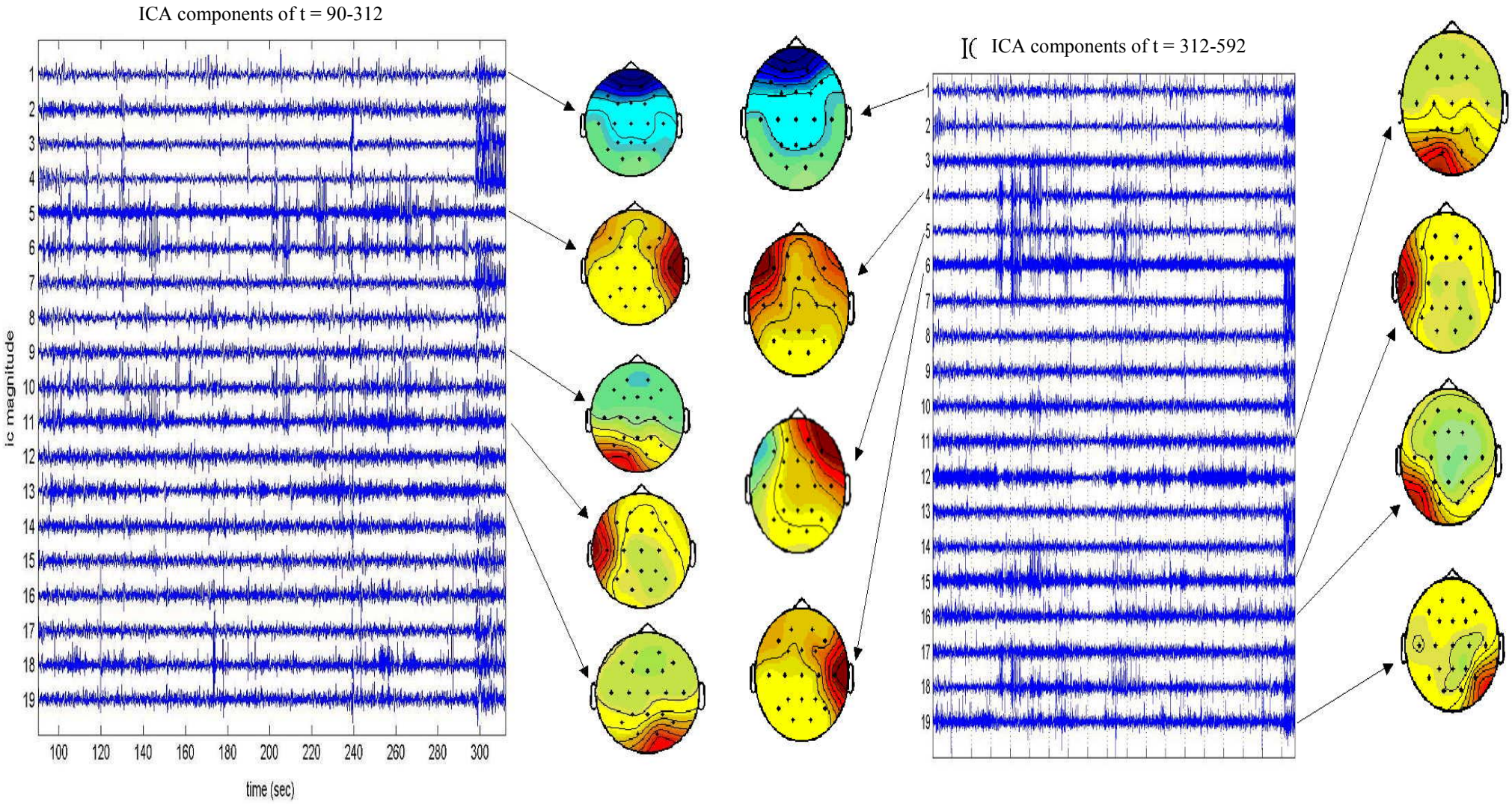


Figure 3.7 Independent Components and projections of selected artifacts for t=90-312 and t=312-592

3.2.3 Data Set II – Corrected EEG Data

As with Data Set I, we removed the selected artifacts of **Figure 3.6** and **Figure 3.7** from the corrupted EEG data given in **Figure 3.5**. **Figures 3.8(a)–(f)** show the corrected EEG data alongside the initial corrupted data. For t = 1-90, we were able to reveal the rhythmic waves (**Figure 3.7(b)**) from the corrupted EEG data after ICA. Rythmic waves are important indicators of the onset of seizures and in this case we see that ICA was successful in removing it from the muscle artifacts.

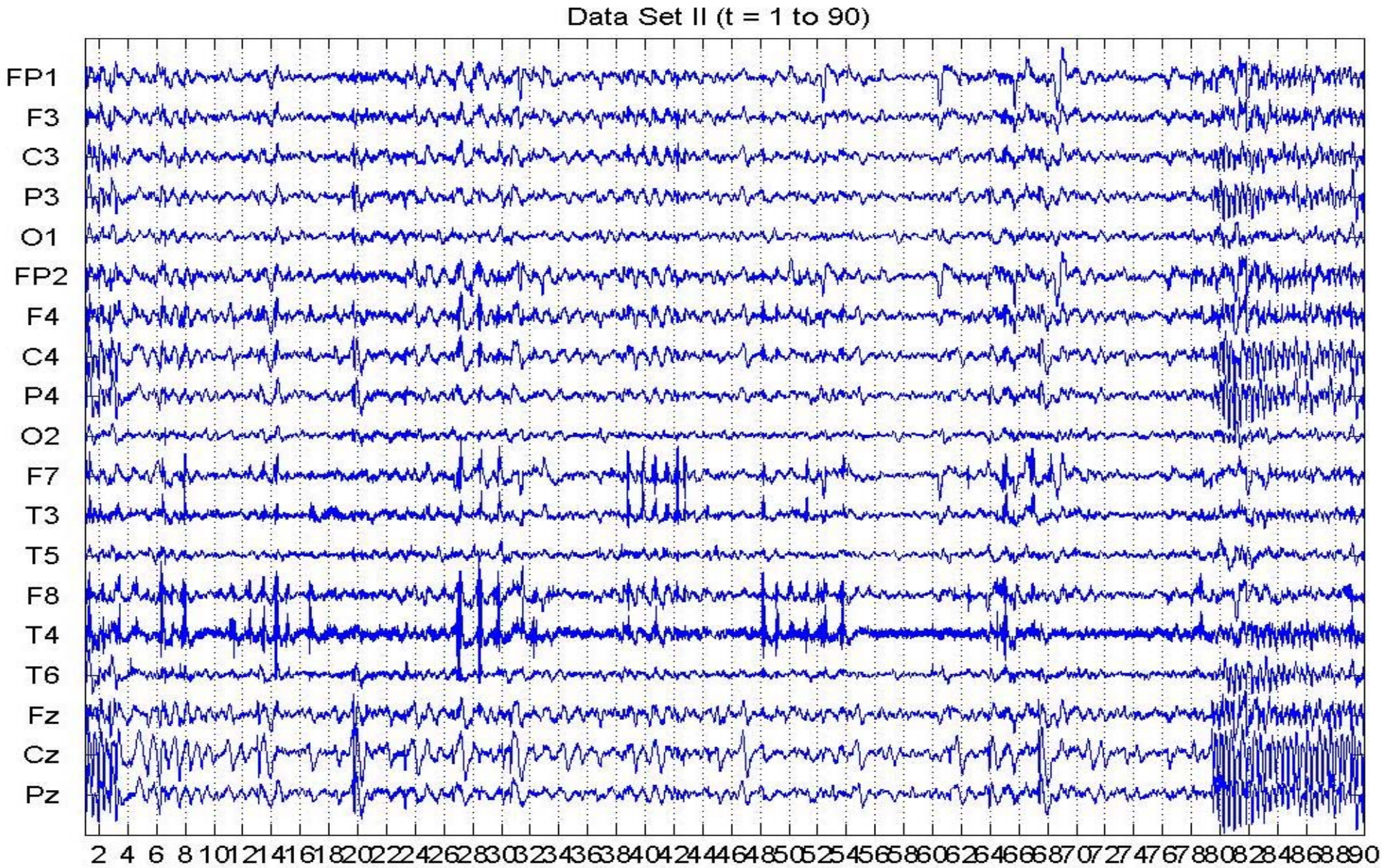


Figure 3.8(a) Corrupted EEG Data (t=1-90)

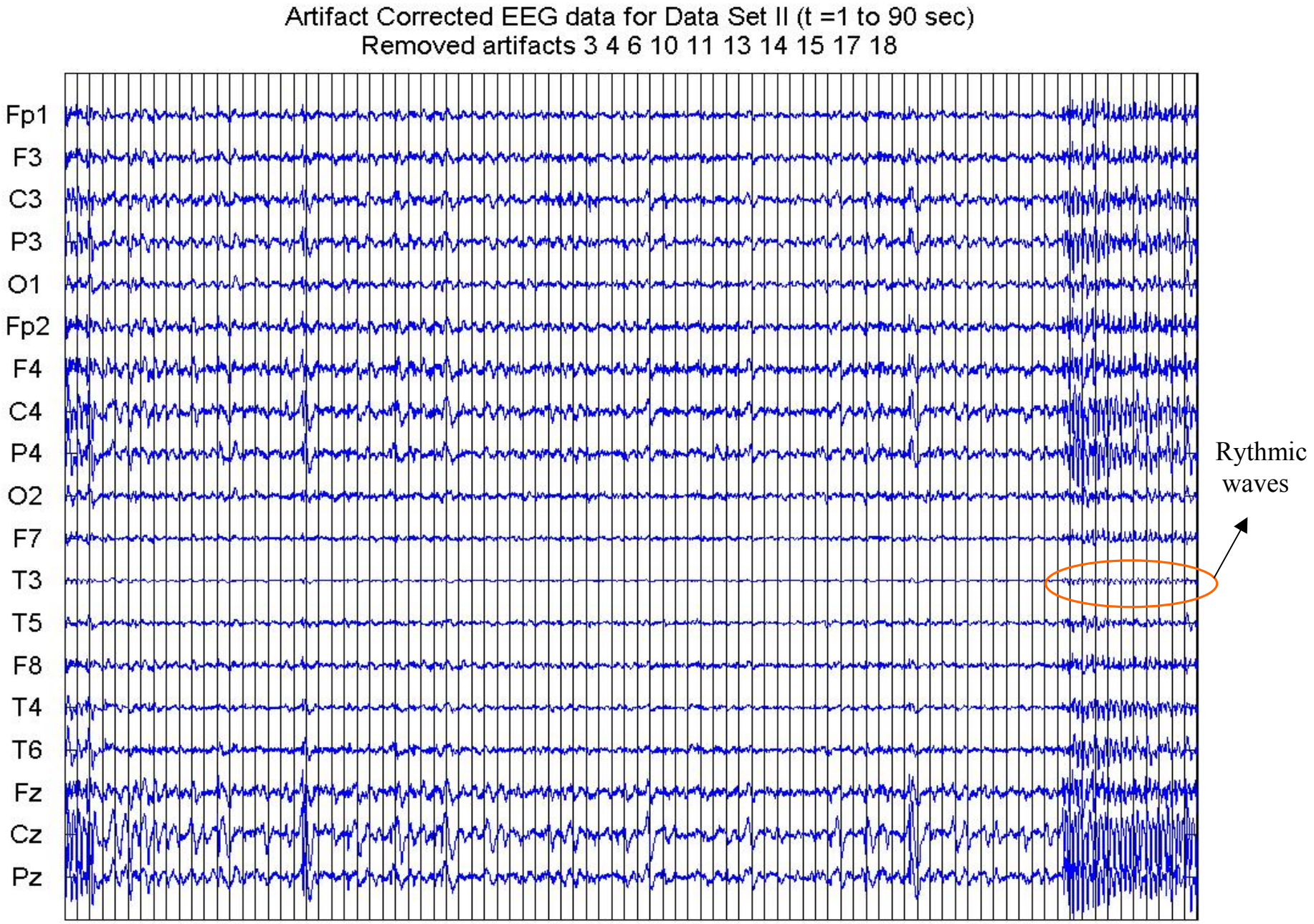


Figure 3.8(b) Corrected EEG Data (t=1-90)

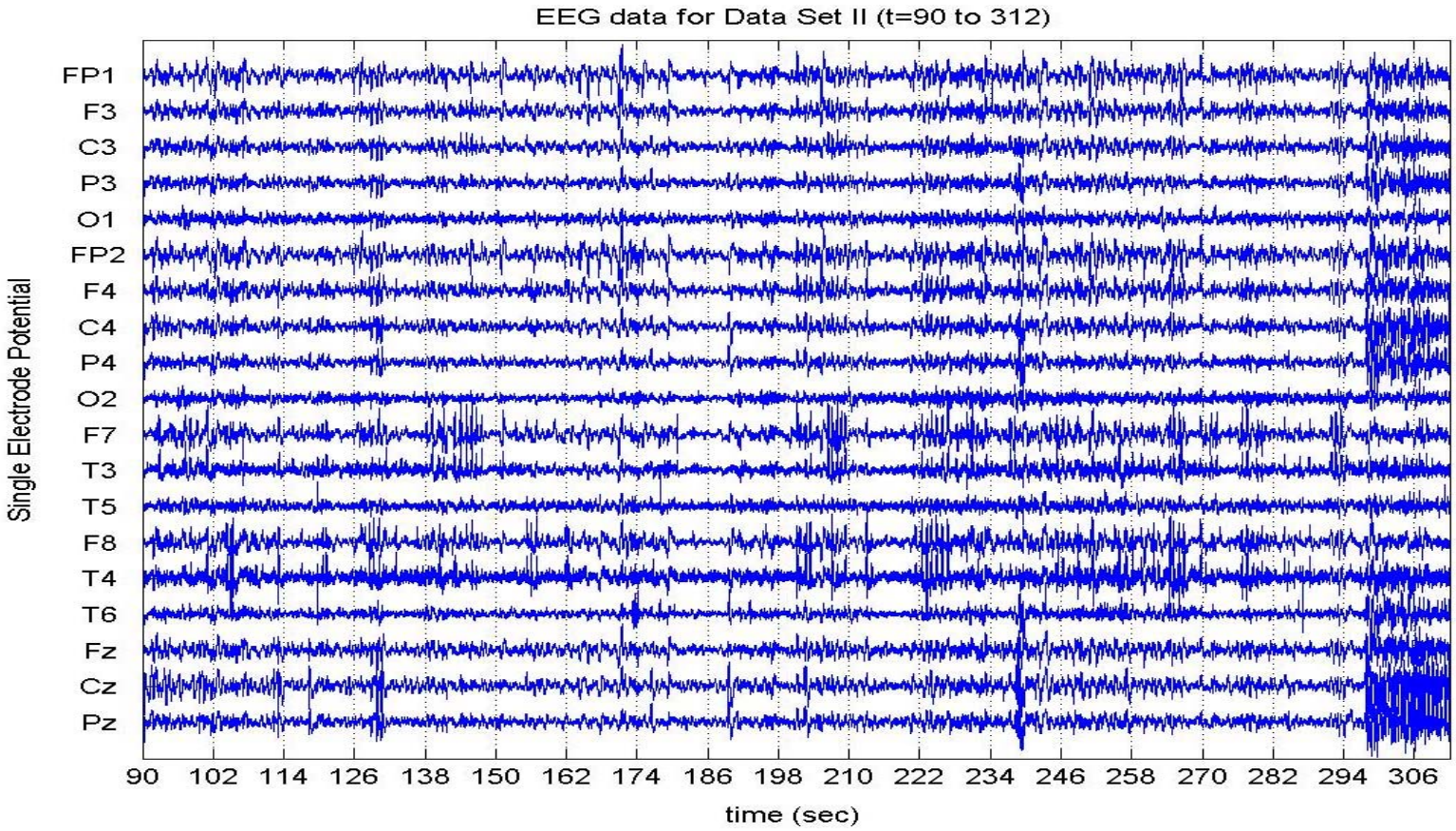


Figure 3.8(c) Corrupted EEG Data (t=90-312)

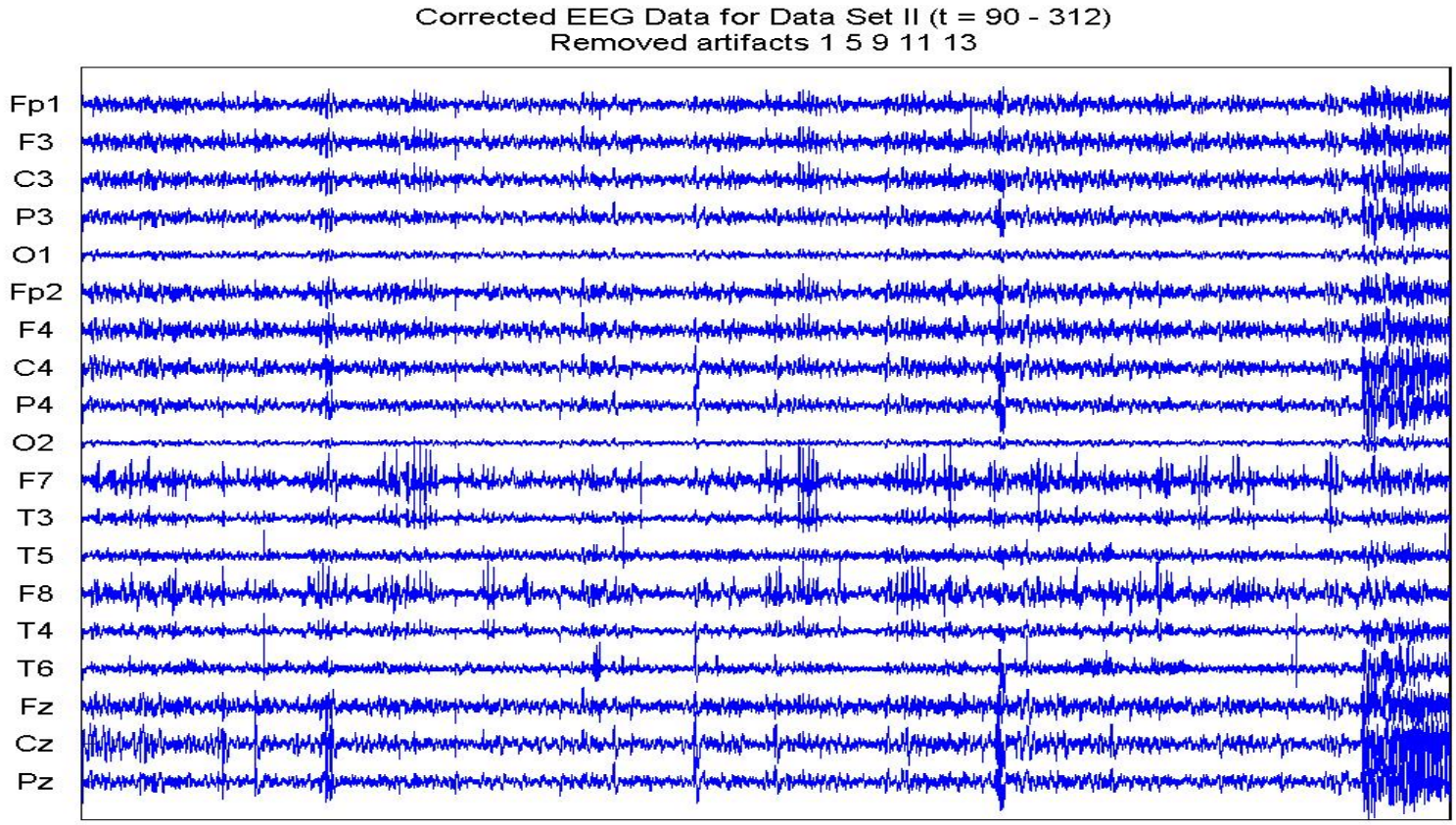


Figure 3.8(d) Corrected EEG Data (t=90-312)

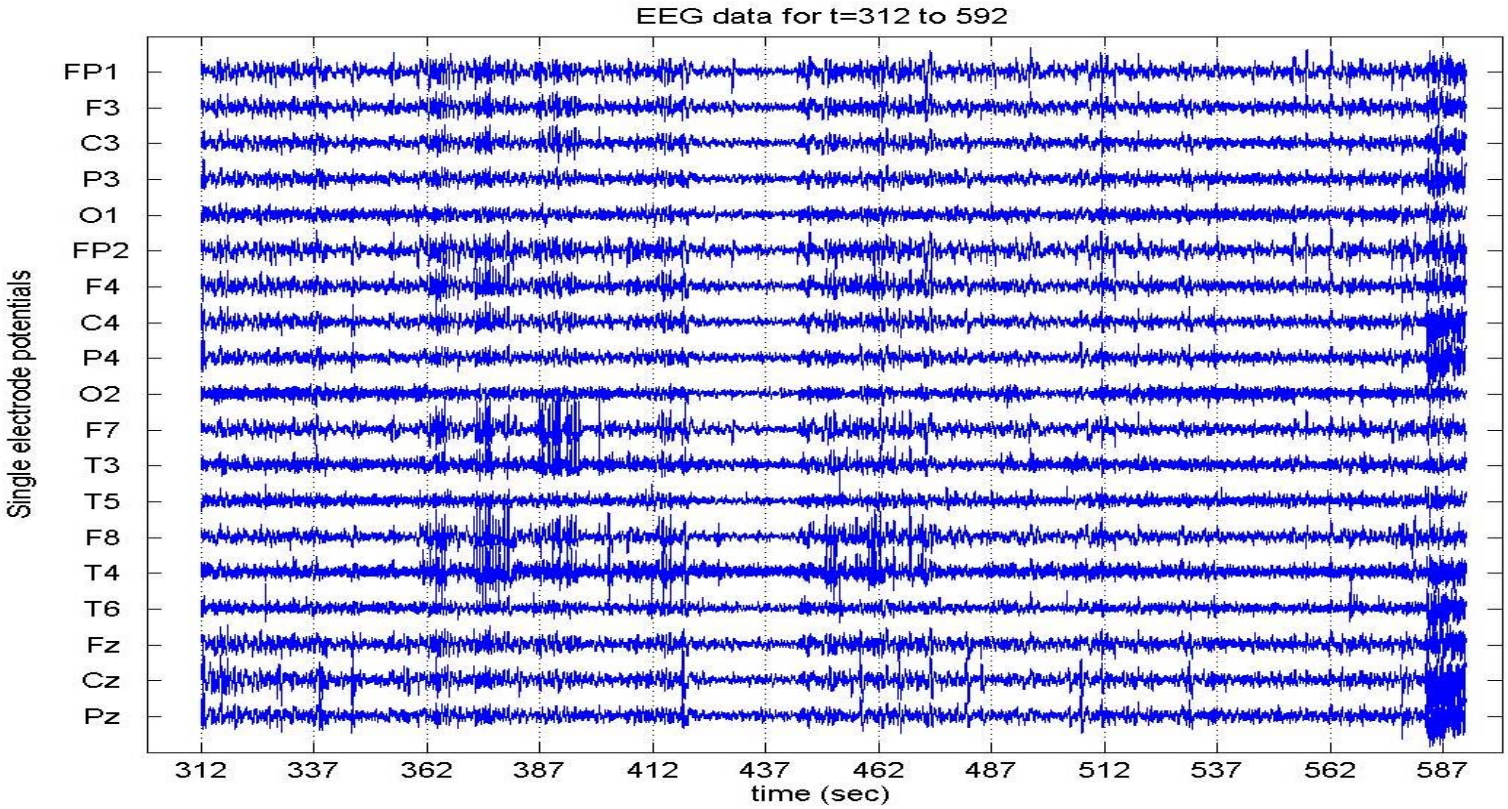


Figure 3.8(e) Corrupted EEG Data (t=312-592)

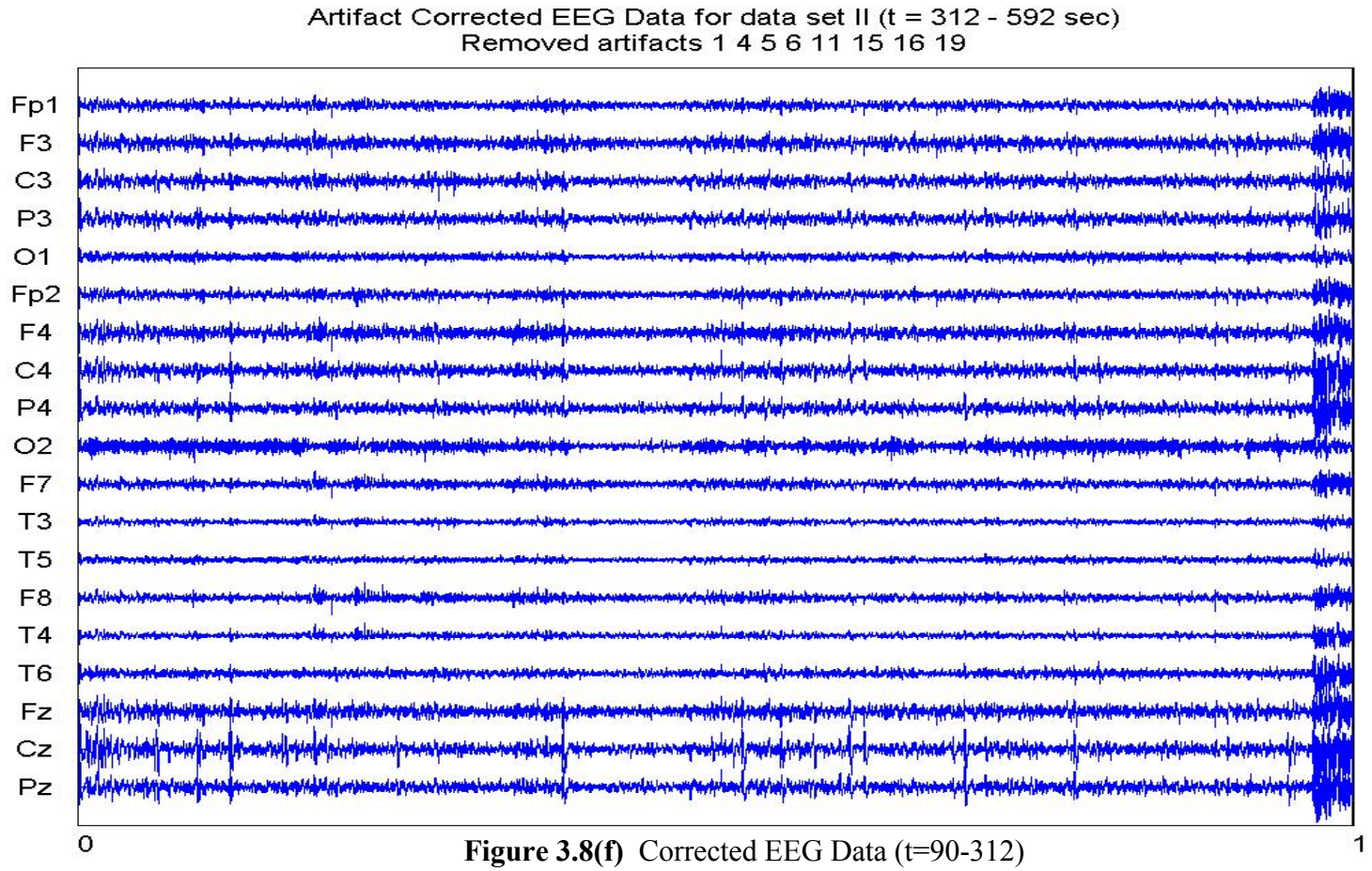


Figure 3.8(f) Corrected EEG Data (t=90-312)

4. Conclusion

Studying the results from the previous chapter, it is clear that Independent Component Analysis is well suited to perform artifact correction on EEG data. The topographical views provided the first clues as to which components might be artifacts. These plots together with the time plots of the independent components were used to identify the eye and occipital artifacts. One of the unique properties of ICA is that it can eliminate the artifacts alone without disturbing the surrounding EEG activity. An alternate approach for artifact extraction could be simply subtracting the frontal, temporal, and occipital readings from the EEG data. But this would lead to considerable loss in collected information.

We were successfully able to identify and eliminate eye and occipital artifacts from Data Set I and II. For Data set II, ICA was also able to reveal the rhythmic waves embedded in the artifacts just before the seizure onset at t = 90 sec.

The muscle artifacts appearing on all channels for Data Set I (t=305–315 sec) and Data Set II (at t=78-90, t=296-512, & t=582-592 sec) after a seizure onset could not be removed or reduced significantly. One reason could be that these artifacts are not concentrated in any one region alone and hence the ICA algorithm cannot interpolate them as originating from any single electrode. Which is why it is difficult to get a single topographical or time plot of an independent component containing the muscle artifacts after a seizure onset. Another reason is that since the person goes into severe spasms on the onset of the seizures, the muscle artifacts following it are of such large magnitude that they completely overshadow the EEG activity originating from within the brain.

APPENDIX – A

Mathematical Preliminaries

For this project, it is assumed that the reader is proficient in the language of general probability theory including the concepts of joint probability, probability density functions (*pdf*), joint gaussian *pdfs*, expectations and moments, statistical independence and correlatedness, and transformation of probability density functions. Since we have employed several results from information theory, we will present a brief introduction and definition of terms such as entropy and mutual information, which were used in the derivation of the Bell Sejnowski Infomax algorithm. The following notes on information theory are taken from Tom Schneiders paper called *Information Theory Primer* [8].

(A-1) Information Theory

Information and uncertainty are technical terms that describe any process that selects one or more object from a set of objects. Suppose we have a device that can produce 3 symbols A, B, and C. As we wait for the next symbol we are *uncertain* as to which symbol will arrive next. Once a symbol arrives and we see it, our uncertainty decreases, and we remark that we have some information. That is, information is a decrease in uncertainty. Uncertainty is measured as the log (base 2) of the possible symbols. Thus, a device producing M symbols has an uncertainty, H, of

$$H = \text{lob}_2(M) \qquad -(A1)$$

The above formula for uncertainty is valid if all the symbols are equally likely. For an unequally likely device with symbols M_i, the uncertainty, H, is given by

$$H = -\sum_{i=1}^{M} P_i \log_2 P_i \qquad -(A2)$$

Where $P_i = \frac{1}{M_i}$ is the probability of the M_i^{th} symbol appearing. It is very simple to show that if the symbols are equally likely (i.e. $M_i = M$ for all i) then (A2) is equal to (A1)

Proof:

If $M_i = M$ for all i, then $P_i = \frac{1}{M_i}$

$$H = -\sum_{i=1}^{M} P_i \log_2 P_i$$

$$= -\sum_{i=1}^{M} \frac{1}{M} \log_2 \frac{1}{M}$$

$$= -\left(\frac{1}{M} \log_2 \frac{1}{M}\right) \sum_{i=1}^{M} 1$$

$$= -\left(\frac{1}{M} \log_2 \frac{1}{M}\right) M$$

$$= -\log_2 \frac{1}{M}$$

$$= \log_2 M$$

H is called the **entropy** of the system. For continuous variables (A2) is written as,

$$H(X) = -\int p(x) \log p(x) = -E\{\log[p(x)]\} \qquad -(A3)$$

Where $p(x)$ is the probability density function of the random variable x.

If there is more than one random variable, then their joint entropy, $H(x,y)$, is defined in terms of their joint probability density function, $p(x, y)$, as

$$H(X,Y) = -\int\int p(x, y) \log p(x, y) = -E\{\log[p(x, y)]\} \qquad -(A4)$$

The last concept of interest is the mutual information, I(X,Y), between two random variables. The mutual information, I(X,Y), is the relative entropy between the joint *pdf*s and the product of the product of the marginal *pdf*s.

$$I(X,Y) = \int\int p(x, y) \log \frac{p(x, y)}{p(x) p(y)} \qquad -(A5)$$

Note: When X and Y are independent ($p(x,y) = p(x)p(y)$) and the mutual information is equal to zero, I(X,Y) = 0. Hence, mutual information can be thought of as a measure of the dependence of random variables on one another.

$$I(X_1,X_2,X_3,\ldots X_n) = 0 \Rightarrow X_1, X_2, X_3, \ldots \text{Xn are statistically independent.} \quad -(A6)$$

Finally, it can be shown that I(X,Y) and H(X,Y) are related as,

$$I(X,Y) = H(X)+H(Y)-H(X,Y) \quad -(A7)$$

(A6) and (A7) are the fundamental relationship used in the deriving the information maximization algorithm of Bell & Sejnowski [4]

APPENDIX – B

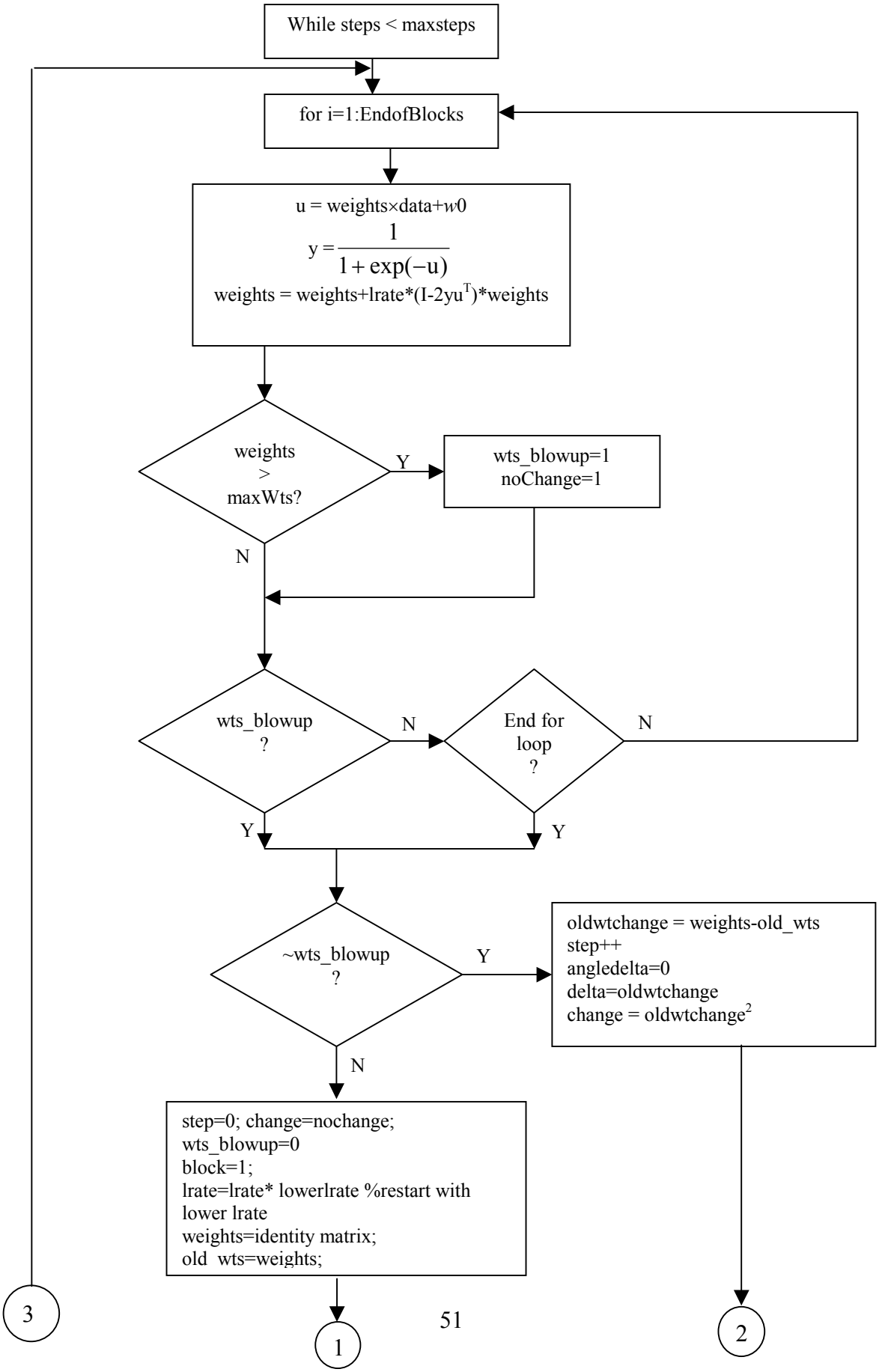
The ICA loop Flow-chart

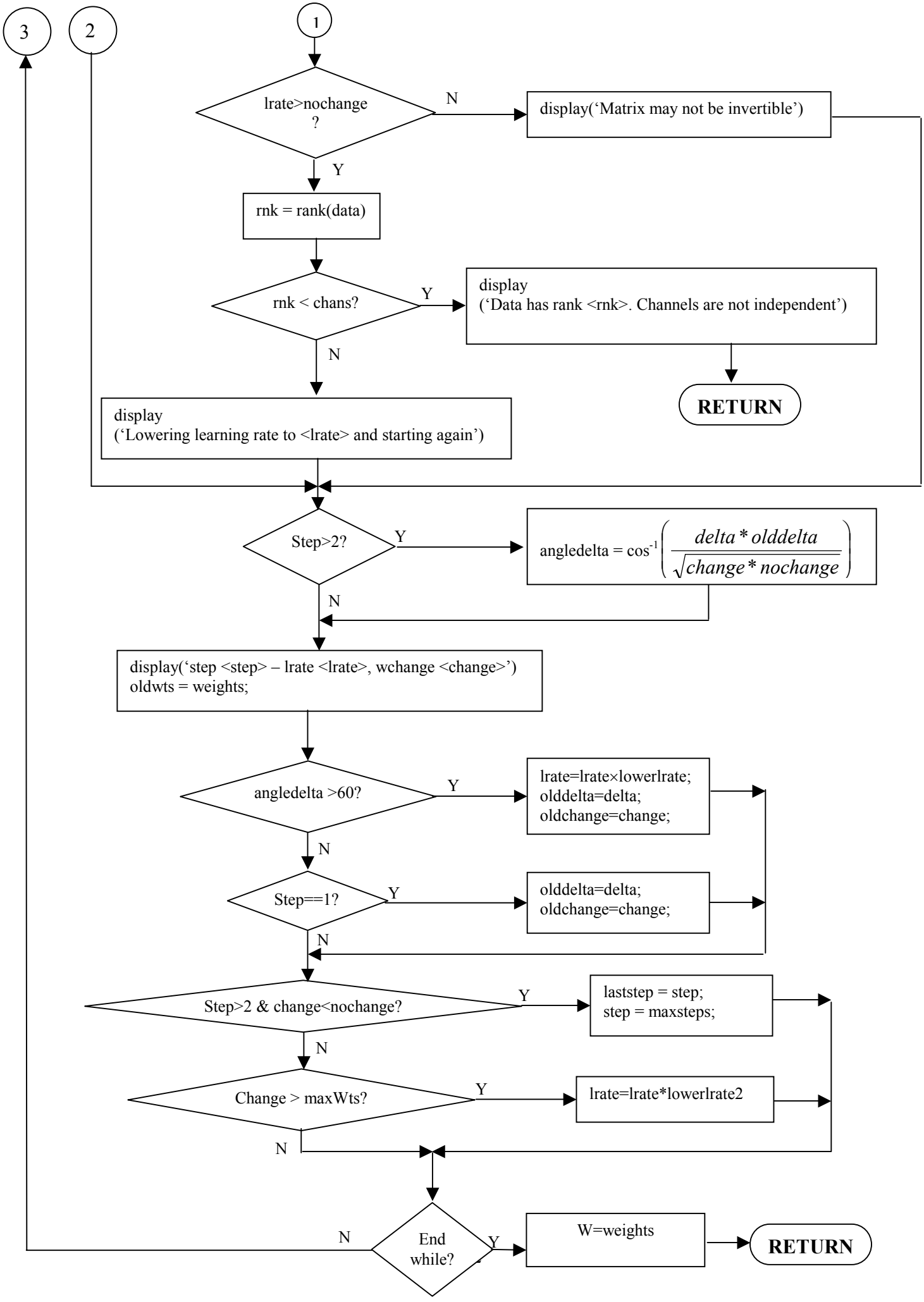
The following ICA loop flow-chart of runica.m incorporates the Bell Sejnowkski ICA algorithm. The following variables are used in the flow-chart:

1. lrate = the learning rate of the algorithm. Taken as 1e-6
2. blocks = the data is grouped into blocks and then processed.
3. maxWts = 1e6. Maximum value of a weight in **W** at which point the ICA is restarted with a lower learning rate
4. lowerlrate = 0.9. If the weights blow up
5. lowerlrate2 = 0.8. If the change in weights ($\Delta\mathbf{W}$) is greater than maxWts
6. maxsteps = 512. Any data with more than 512 steps will probably not converge.
7. angledelta = 60. The rotation angle of the probability density function

All of the above constants are heuristic values recommended by people doing ICA [7].

The Bell & Sejnowski Infomax algorithm flowchart




3
2
1
lrate>nochange ?
N
display('Matrix may not be invertible')
Y
rnk = rank(data)
rnk < chans?
Y
display ('Data has rank <rnk>. Channels are not independent')
N
RETURN
display ('Lowering learning rate to <lrate> and starting again')
Step>2?
Y
angledelta = $\cos^{-1}\left(\frac{delta * olddelta}{\sqrt{change * nochange}}\right)$
N
display('step <step> – lrate <lrate>, wchange <change>')
oldwts = weights;
angledelta >60?
Y
lrate=lrate×lowerlrate;
olddelta=delta;
oldchange=change;
N
Step==1?
Y
olddelta=delta;
oldchange=change;
N
Step>2 & change<nochange?
Y
laststep = step;
step = maxsteps;
N
Change > maxWts?
Y
lrate=lrate*lowerlrate2
N
End while?
N
Y
W=weights
RETURN

Bibliography

[1] Aapo Hyvärinen, J. Karhunen, *Independent Component Analysis*, 2001 John Wiley & Sons.

[2] Aapo Hyvarinen, *Survey on Independent Component Analysis*
http://www.cis.hut.fi/~aapo/

[3] Aapo Hyvarinen, *Beyond Independent Components*
http://www.cis.hut.fi/~aapo/

[4] Anthony J Bell & Terrance J Sejnowski, *An information-Maximisation approach to blind separation and blind deconvolution*, Neural Computation, 7,6, 1004-1034 (1995)

[5] Dominic Chan, *Blind Signal Separation*, PhD Dissertation, University of Cambridge

[6] Jean-Francois Cardoso *Blind Signal Separation: Statistical principles* Proceedings of the IEEE, vol 9, no 10. p 2009-2025 Oct 1998

[7] Scott Makeig, *Independent Component analysis of Electroencephalographic Data* Advances in Neural Information Processing Systems 8''. MIT Press, Cambridge MA 1996
http://www.sccn.ucsd.edu/~scott/index.html

[8] Thomas Schneider, *Information Theory Primer*,
http://www.lecb.ncifcrf.gov/~toms/paper/primer