# PATTERN RECOGNITION THROUGH SOURCE SIGNAL SEPARATION

---

A Dissertation presented to the Faculty of the Graduate School
University of Missouri

---

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in Electrical and Computer Engineering

---

by

**LUIS ALBERTO RIVERA**

Dr. Guilherme N. DeSouza, Dissertation Supervisor

July 2016

The undersigned, appointed by the dean of the Graduate School, have examined the dissertation entitled

PATTERN RECOGNITION THROUGH SOURCE SIGNAL SEPARATION

Presented by Luis Alberto Rivera,
a candidate for the degree of Doctor of Philosophy,
and hereby certify that, in their opinion, it is worthy of acceptance.

---

Dr. Guilherme DeSouza, Associate Professor, Dept. of Electrical and Computer Engineering

---

Dr. Satish Nair, Professor, Dept. of Electrical and Computer Engineering

---

Dr. Chi-Ren Shyu, Professor, Dept. of Computer Science

---

Dr. Alina Zare, Assistant Professor, Dept. of Electrical and Computer Engineering

---

Dr. Dominic Ho, Professor, Dept. of Electrical and Computer Engineering

To my family and friends.

# ACKNOWLEDGMENTS

I would like to thank:

God, for all the blessings I have received throughout my life.

My family and friends, for their support and encouragement.

My advisor, Professor Guilherme DeSouza, for his guidance and support throughout my graduate studies.

All those who collaborated in this research, including Darren Gabbert, Dinal Andreasen, Akshay Jain, Siavash Farzan, Nick Smith, Maria Dietrich, Chi-Ren Shyu, Matthew Page, Magda El-Shenawee and her research team, Changzhe Jiao, my committee members, and all the test subjects, including the members of the ViGIR lab.

The Coulter Foundation, the ECE department, the Mathematics department, Academic Retention Services, and the Department of Romance Languages and Literatures, for their financial support.

# Contents

# List of Figures

# List of Tables

# ABSTRACT

In typical problems involving pattern recognition, the challenge lies in selecting a good set of features and in devising a reliable algorithm to identify the class of learned patterns that most resembles the observed feature vector. Some times, however, the observed vector is not a single, but a mixture of multiple learned patterns and the challenge becomes to recognize all the present patterns and not just one of them. In order to do so, the patterns in the observed feature vector must first be separated – an apparent paradox since the actual patterns forming the observed vector are hitherto unknown and should probably be identified first. At the same time, many techniques to separate mixture of signals have emerged from the literature in signal processing, but they require multiple and independent observations of the mixture of patterns, which is not usually possible or desirable in a pattern recognition setting. However, we believe that these two problems – pattern separation and recognition – are one the same, and it can benefit from a hybrid technique derived from both contexts. So, in this research, we propose a technique based on Source Separation for recognizing patterns in mixtures of signals. From the signal processing perspective, our method can handle extremely under-determined cases, i.e., cases where one measurement is required despite the existence of multiple patterns mixed in the measurement – a typical scenario from the pattern recognition perspective. We have run extensive tests to demonstrate the robustness and effectiveness of the method. We have also proposed frameworks for applications in various areas such as classification of chemical compounds using terahertz signatures; root phenotyping using terahertz imaging; recognition of muscle activity patterns using surface electromyographic signals (sEMG) for Robotic Assistive Technology; detection of vocal dysfunctions; and Hyperspectral Image analysis.

# Chapter 1

# Introduction

## 1.1 Problem Statement

The ability to detect specific patterns in a signal is beneficial in many different areas. One such area is Terahertz (THz) technology, which has attracted the attention of researchers for many years, and it has seen many advances over the past decades. Special interest for the use of this technology is actually directed to material detection, where being able to recognize signatures indicating the presence of explosives or illegal substances is very important. Illegal drugs, explosives and other hazardous materials exhibit characteristic signatures at terahertz wavelengths which may be used to identify them. Much effort has been made to detect improvised explosive devices (IEDs) in the fields, as well as drugs or hazardous substances in airports and other public places. Furthermore, terahertz imaging is becoming a valuable tool in many areas, including medicine, pharmacology, security, etc. and has the potential for nondestructive evaluation technique used for analyzing non-conducting materials.

The human body generates various types of signals. For example, electromyographic (EMG) signals are electrical signals generated when there is muscle activity. These signals find applications in many areas such as rehabilitation, prosthesis and human-machine interaction, Otolaryngology, and others. Systems reliant on them require various forms of machine learning algorithms for recognition of specific components or patterns. Those systems vary in terms of the signal detection methods, the feature

selection and the classification algorithm used.

We believe that signal processing can offer powerful insights into solving the problem of recognizing multiple patterns present in single observations. In this work, we propose and develop a technique for pattern recognition based on a classical signal processing problem – Source (Signal) Separation – and frameworks for applying that technique in areas such as the ones mentioned above. Source Separation aims at estimating source signals that may combine to form mixture signals. These mixtures are observations obtained at the output of a set of sensors, where each sensor receives a different combination of the source signals.

## 1.2 Contributions of the Thesis

In this work we greatly enhanced the Guided Under-determined Source Signal Separation (GUSSS) technique that we presented in [1]. We developed GUSSS based classifiers and frameworks that we tested using public THz databases, as well as transmission and reflection THz signals from samples including carrots, sweet potatoes and others buried in sand and in potting soil. We achieved high true positive and true negative percentages.

We have also tested our methods is Assistive Technology using surface electromyographic (sEMG) signals. Through our technique, we are able to find patterns in muscle activity, recognize different "gestures" that can be used to control systems such as power wheelchairs and other assistive devices for people with disabilities.

We have also applied our method in the area of Otolaryngology. We investigated the possibility of detection of normal and maladaptive extralaryngeal patterns associated with voice problems. Our results are very promising, showing that despite the complexity of the muscle groups on the neck, meaningful detection of vocal dysfunctions through the recognition of sEMG signals is possible, at high levels of accuracy.

Research questions have been proposed to investigate the generality, robustness, and limitations of our method. The answers to those questions can serve as a foundation and guidelines for applying our method in other pattern recognition problems.

To summarize, the contributions of this research are as follows:

- Pattern recognition technique based on source separation, which can handle cases of extreme under-determination.
- Frameworks for application in different areas:
  - Material detection using THz signatures.
  - Root phenotyping using THz imaging.
  - Classification of muscle activity for applications in Assistive Technology and Otolaryngology.
  - Hyperspectral Image Analysis.
- Extensive experimentation on different metrics of independence with impact on Independent Component Analysis and on the proposed pattern recognition method.
- Theoretical study of the strengths and limitations of the method.

The organization of this work is as follows. Chapter 2 presents an overview on the topics and techniques related to this thesis. Chapter 3 presents the GUSSS method and the classification frameworks that have been developed and applied in different areas. Chapter 4 presents relevant background and methods found in the literature for the specific areas mentioned above, as well as the experimental results that have been obtained using the GUSSS method. A more formal validation of the method is presented in chapter 5, where fundamental research questions are raised and extensive tests are performed to address them. Chapter 6 presents a final discussion and concluding remarks, and Chapter 7 discusses possible future directions for this research. The complete set of results are shown in Appendices A - C.

# Chapter 2

# Background and Related Work

In this chapter we present an overview on the topics and techniques related to this thesis. We begin with a classical problem known as the Cocktail Party Problem, which illustrates some of the ideas that we further develop later on. We then discuss Source Separation and Independent Component Analysis, both of which are key concepts in our approach.

The methods and frameworks that will be presented in chapter 3 have been used in various areas, such as Terahertz Technology and Robotic Assistive Technology. Background on those areas are not presented here, but in chapter 4, instead.

## 2.1 The Cocktail Party Problem

The cocktail party problem is the task of hearing a sound of interest in an environment where many sources emit sounds concurrently. The sounds are added together generating the mixed signal that enters the ear (microphone, sensor). At different locations, individual sounds would have different intensities [2, 3].

Consider 2 speakers and 2 microphones, as illustrated in Figure 2.1. Each sensor captures a different fraction of the signals emitted by the sources. So, the sensed mixed signals can be expressed as linear combinations of the sources:

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\ x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) \end{aligned} \tag{2.1}$$

where $s_i(t)$ denotes the signals emitted by the $i^{th}$ source, $x_j(t)$ denotes the mixed signal captured by the $j^{th}$ sensor, and the $a_{ij}$ are the mixing coefficients of the linear combination model, which depend on the distances of the microphones from the speakers.

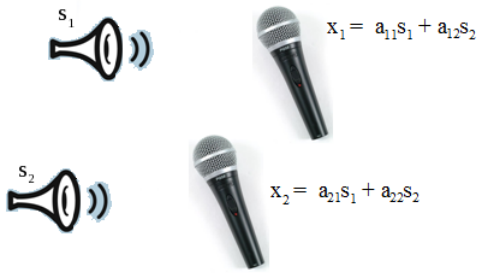



Figure 2.1: Two independent sources (speakers) emit sounds that mix in air and reach the microphones.

The important question becomes how to recover the original source signals $s_i(t)$ using the sensed signals $x_j(t)$. If there is a prior knowledge of the mixing coefficients, the linear system in eq. 2.1 can be solved by classical methods. However, it is rarely the case that information on the coefficients is available. The problem can then be addressed by estimating the coefficients $a_{ij}$ using some information on the statistical properties of the signals $s_i(t)$. Specifically, if these source signals are statistically independent at each time instant, then it is possible to estimate the mixing coefficients [3, 4].

## 2.2 Source (Signal) Separation

Let $S = \{s_1(t), \ldots, s_N(t)\}$ be a set $N$ source signals, and let $\mathbf{S}(t) = [s_1(t), \ldots, s_N(t)]^T$ be a vector representation of the set $S$. As those sources pass through a medium, they

may combine somehow. Let $\mathbf{X}(t) = [x_1(t), \ldots, x_M(t)]^T$ be the vector representation of the set of $M$ observed signals $X = \{x_1(t), \ldots, x_M(t)\}$, which are mixtures of the sources signals. Typically, the observations are obtained at the output of a set of sensors, and each sensor receives a different combination of the source signals. In general, the relationship between $\mathbf{S}$ and $\mathbf{X}$ can be formulated as

$$\mathbf{X}(t) = \mathcal{A}(\mathbf{S}(t))$$

where $\mathcal{A}(\cdot)$ represents the mixing process. Source or Signal Separation (SS) aims at estimating the sources $S$ given the observations $X$. For convenience in the notation, we drop the dependency on time.

Many assumptions or constraints can be considered for the sources or the mixing process [5–11]. For instance, sources can be modeled as mutually independent random variables, they can be considered as Gaussian or non-Gaussian processes, etc. Similarly, the mixing process can be a nonlinear mapping, or it can be a linear model. The latter case is expressed as

$$\mathbf{X} = A\mathbf{S}$$

where $A$ is an $M \times N$ mixing matrix.

## 2.3 Blind Source Separation

The idea of Blind Source Separation (BSS) is to process observations in order to recover the original sources without a prior knowledge about the mixing operation or the sources themselves, hence the term blind. This so-called blindness, or weakness of the prior information, can be seen as a strength of the BSS model, as it can be a versatile tool for exploiting the spatial diversity provided by an array of sensors [10].

Over the past few decades, BSS has gained much attention in many different areas, such as communications, geophysical exploration, airport surveillance, biomedical signal analysis, fall detection, chemical analysis, and signal processing in gen-

eral. [3, 4, 8, 11–16]. Many approaches have been developed to solve the BSS problem. Most are based on the second or higher order statistics of the data. So called block-based methods exploit the temporal correlations of the source signals [6, 17].

The majority of BSS techniques are based on specific assumptions regarding either the deterministic or statistical properties of the sources or the sensor and channel characteristics, e.g. how many observations, $M$, vs. how many source signals, $N$. Most techniques are well suited for over and well determined cases ($M \geq N$).

## 2.4 Under-determined SS

For the under-determined cases, that is, when the number of sensors is smaller than the number of independent sources ($M < N$), methods for source separation have also been proposed and referred to as Under-determined Source Separation (USS) [18–24]. In this under-determined case, the mixing matrix $A$ (assuming a linear model) is non-invertible, leading to a more challenging problem than the well determined case. The methods may produce losses in the recovered (separated) signals which increase with the reduction of the number of sensors. This can drastically affect the accuracy of any subsequent classification of the source signals.

The many existing methods for USS impose additional assumptions on the sources or mixing processes. Some assume that the source signals are sparse, i.e., not all of them are active at any given time [18, 25]. Other methods assign prior probabilities to the sources or exploit their statistical properties [22, 26, 27].

The extreme case of under-determination, i.e., when $M = 1$, has been addressed by a few studies. In [28], the problem of audio source separation with a single sensor was discussed, with the limitation of considering only two sources. Prior information about characteristic Power Spectral Densities of each source is used the that framework. The method proposed in [29] assumes stationary sources that are also disjoint in the frequency domain. Those assumptions are very restrictive for practical applications. The main idea of [29] is to break up the single signal into a sequence of contiguous blocks, and apply Independent Component Analysis (ICA) tho those blocks. ICA is

perhaps the most common method for BSS, and will be discussed thoroughly in the next sections. More recent methods for dealing with single observations are those in [30] and [31]. In both cases, the single observation is decomposed into a set of components through empirical-mode decomposition (EMD) or Local Mean Decomposition (LMD), respectively, and then ICA is applied to those components.

## 2.5 BSS using Independent Component Analysis

Independent Component Analysis (ICA) is a powerful technique for blind source separation [4, 30–33]. It is assumed that sensed signals contain various statistically independent components. It is important to emphasize that each component actually originates from a different source. We present more details on statistical independence and how to measure its validity in section 2.5.2.

Mathematically, the goal of BSS-ICA is to recover $N$ source signals, $S$, assumed to have been linearly mixed, producing the observed signals $X$. A typical example would be $N$ independent sounds emanating from different sources and being detected as mixed signals by $M$ microphones spread over the space [13], as illustrated in Figure 2.1. The figure depicts this example for $N = 2$ sound sources $M = 2$ and microphones.

A very simple, artificial example is shown in Figure 2.2. On the left there are two mixed signals that could have been obtained from two sensors. On the right we see two separated, independent signals recovered using BSS-ICA. It is not difficult to recognize the mixtures as having been produced by linear combinations of these two source signals.

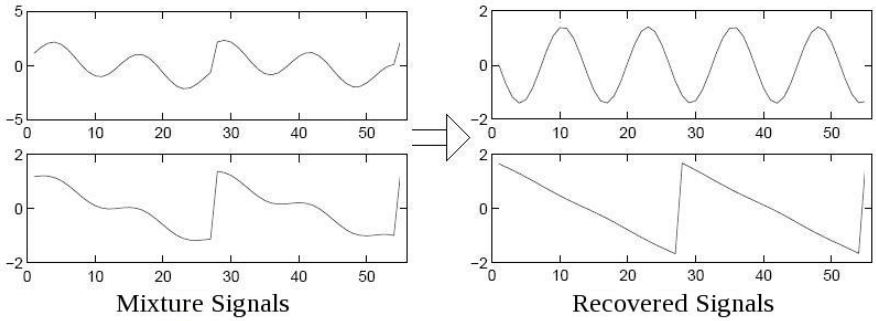



Figure 2.2: Out of two recorded mixture signals we obtain the original, independent signals which generated the mixtures in the first place.

### 2.5.1 Definition of ICA

A general definition of ICA as stated in [4] is the following: ICA of the random vector $\mathbf{X}$ consists of estimating the generative model for the data: $\mathbf{X} = A\mathbf{S}$, where the components $s_i$ in vector $\mathbf{S} = [s_1, \ldots, s_N]^T$ are assumed independent and matrix $A$ is a constant $M \times N$ matrix, usually called mixing matrix.

The model being identifiable can be assured if (1) all the independent components $s_i$, with the possible exception of one component, are non-Gaussian; (2) the number of observed linear mixtures $M$ is at least as large as the number of independent components $N$, i.e $M \geq N$; and (3) the matrix $A$ is full column rank [4].

Traditional ICA methods are able to separate the signals whenever $M \geq N$. (If $M > N$, the dimension of the observed vector $\mathbf{X}$ can always be reduced so that $M = N$). In those cases, the source signals and the observed signals can be related in a matrix form such as $\mathbf{X} = A\mathbf{S}$ – where $A$ contains the coefficients of the linear combination of the sources. The methods can solve the (overdetermined) system of equations through the expression $\mathbf{S} = A^{-1}\mathbf{X} = W\mathbf{X}$. The solution is found using a constrained optimization algorithm that maximizes the independence of the signals in $\mathbf{S}$. We present some measures in the next few sections.

### 2.5.2 Principles of ICA Estimation

In the previous sections we have mentioned the importance of the statistical independence of the components (i.e. source signals). In this section we present formal details on this key concept and the principles of an ICA estimation. Different algorithms for ICA are referenced in Section 2.5.8.

Let $y_1, \ldots, y_N$ be random variables. Let $p(y_1, \ldots, y_N)$ be the joint probability density function (pdf) and let $p_1(y_1), \ldots, p_N(y_N)$ be the marginal pdfs of $y_1, \ldots, y_N$, which can be obtained as

$$p_i(y_i) = \int p(y_1, \ldots, y_N)\, dy_1 \cdots dy_{i-1} dy_{i+1} \cdots dy_N$$

The variables $y_1, \ldots, y_N$ are said to be statistically independent if and only if the joint pdf is factorisable as the product of the $N$ individual marginal pdfs, i.e.

$$p(y_1, \ldots, y_N) = p_1(y_1)\, p_2(y_2) \cdots p_N(y_N)$$

Two random variables $y_i$, $y_j$, $i \neq j$, are uncorrelated if their covariance is zero, i.e.

$$E\{y_i y_j\} - E\{y_i\}\, E\{y_j\} = 0$$

where $E\{\cdot\}$ is the expectation operator. It is well known that if two variables are independent, then they are uncorrelated. However, the inverse implication is not true. In general, uncorrelated variables are not necessarily independent. The equivalence between independence and uncorrelatedness holds for Gaussian random variables, though.

A restriction imposed in ICA is that independent components have non-Gaussian distributions. The reason is that the distribution of any orthogonal transformation of independent Gaussian variables $y_i$, $y_j$ has the same distribution as the original $y_i$, $y_j$ variables. Therefore, the ICA model can only be estimated up to an orthogonal transformation. The mixing matrix $A$ will not be identifiable for Gaussian independent components. It must be pointed out, however, that if only one of the independent

components is indeed Gaussian, the ICA model can still be estimated [2–4, 33].

ICA has some ambiguities. First, the independent components can only be estimated up to a multiplicative constant. Typically, methods assume components to have unit variance, i.e., $E\{s_i^2\} = 1$. Thus, the ambiguity is in a multiplicative sign. Second, unlike Principal Component Analysis (PCA), there is no order of the independent components. These ambiguities are inconsequential for most applications.

### 2.5.3 Nongaussianity and Independence

According to the Central Limit Theorem, under certain conditions, the distribution of a sum of independent random variables tends toward a Gaussian distribution. In that sense, the distribution of the sum of **two** independent variables can already be closer to a Gaussian distribution than any of the two original distributions.

In [3] it is shown that maximizing the nongaussianity leads to finding component as independent as possible. The basic ideas are the following. Consider a data vector $\mathbf{X} = [x_1, \ldots, x_M]^T$ distributed according to the ICA model $\mathbf{X} = A\mathbf{S}$, where $\mathbf{S}$ is the vector representation of the set of independent components $S = \{s_1, \ldots, s_N\}$. Consider the linear combination $y = \mathbf{w}^T\mathbf{X} = \sum_i w_i x_i$, where $\mathbf{w}$ is a vector that needs to be determined. $A$ is the mixing matrix, so if $\mathbf{w}$ were one of the rows of $A^{-1}$, then $y$ would equal one of the independent components in $S$. However, there is no prior knowledge of $A$, so there is a need for a good approximation.

Let $\mathbf{z} = A^T\mathbf{w}$. Then, $y = \mathbf{w}^T\mathbf{X} = \mathbf{w}^T A\mathbf{S} = \mathbf{z}^T\mathbf{S}$, a linear combination of the $s_i$. The coefficients or weights are the elements $z_i$ of vector $\mathbf{z}$. Due to the fact that the sum of independent random variables is more Gaussian than the original variables, the variable $y = \mathbf{z}^T\mathbf{S}$ is more Gaussian than any of the individual $s_i$. Conversely, $y$ would be the least Gaussian if only one of the elements $z_i$ was nonzero. This would be the case if $y$ was in fact equal to one of the $s_i$. So, it would be desired to find a vector $\mathbf{w}$ that maximizes the nongaussianity of $\mathbf{w}^T\mathbf{X}$. This maximization would provide one of the independent components. In general, the optimization of nongaussianity in the $N$-dimensional space of vectors $\mathbf{w}$ results in $2N$ local maxima, two for each independent component. Those correspond to $s_i$ and $-s_i$. This implies that the independent components can be

estimated only up to a multiplicative sign.

We have seen that nongaussianity is key to ICA estimation. So, there is a need for a quantitative measure of the nongaussianity of a random variable. Next we present a brief review of some of these measures. It is important to mention that if a random variable is centered (i.e. it has a zero mean) and has a unity variance, the calculations are simplified. Traditional ICA algorithms include pre-processing steps such as centering and whitening so that the ICA estimation becomes simpler and better conditioned [3, 4, 33].

### 2.5.4 Some Measures of Nongaussianity

#### 2.5.4.1 Kurtosis

The kurtosis of the random variable $y$ is defined by

$$kurt\left(y\right) = \frac{E\left\{\left(y-\mu\right)^4\right\}}{\sigma^4}$$

where $\mu$ is the mean and $\sigma^2$ is the variance of $y$ [34]. If $y$ is a zero mean Gaussian, then $kurt\left(y\right) = 3$, since $E\left\{y^4\right\} = 3\left(E\left\{y^2\right\}\right)^2 = 3\left(\sigma^2+\mu^2\right)^2 = 3\left(\sigma^2\right)^2 = 3\sigma^4$.

Some authors define the kurtosis so that Gaussians have kurtosis equal to zero. In [3], kurtosis is defined as

$$kurt\left(y\right) = E\left\{y^4\right\} - 3\left(E\left\{y^2\right\}\right)^2$$

With this definition, it is clear that kurtosis is zero if $y$ is a zero mean Gaussian. It is nonzero for most nongaussian random variables. It can be positive or negative. The absolute value and the square of the kurtosis are typically used as a nongaussianity measures. They have been used in ICA and related fields, mainly due to their simplicity: both computational and theoretical. In simple terms, given a data vector $\mathbf{X}$, the ICA algorithm would search for the weight vectors $\mathbf{w}$ that maximize the absolute value of the kurtosis of $y = \mathbf{w}^T\mathbf{X}$ . It is worth recalling that ICA algorithms pre-process the data vector $\mathbf{X}$ to center it. An important drawback for the use of kurtosis is its sensitivity to outliers and therefore, it is not a very robust measure of nongaussianity [3].

#### 2.5.4.2 Negentropy

The concept of entropy is very important in information theory. For a random variable, the more unpredictable and unstructured (i.e. the more *random*) it is, the larger its entropy is. For a discrete random variable $Y$, entropy is defined as

$$H(Y) = -\sum_i P(Y = a_i) \, log\left[P(Y = a_i)\right]$$

where $a_i$ are the possible values of $Y$. The generalization for continuous random variables is usually called differential entropy, and is defined as

$$H(y) = -\int p(y) \, log\left[p(y)\right] dy$$

where $p(y)$ is the density function of the random variable $y$. It turns out that a Gaussian variable has the largest entropy among all random variables of equal variance [35]. Therefore, it could be used as a measure of nongaussianity.

A modified version of the differential entropy is the negentropy. It is defined as

$$J(y) = H(y_{gauss}) - H(y)$$

where $y_{gauss}$ is a Gaussian random variable of the same covariance matrix as $y$. Negentropy has the properties of being always non-negative, and being zero if and only if $y$ has a Gaussian distribution. As a measure of nongaussianity, negentropy is a well justified by statistical theory. Nevertheless, the use of this measure is problematic in practice, since it is computationally difficult to calculate. Therefore, approximations of negentropy have to be used. A classical approximation of negentropy is

$$J(y) \approx \frac{1}{12} E\left\{y^3\right\}^2 + \frac{1}{48} kurt(y)^2$$

where $y$ is assumed to be of zero mean and unit variance. This approximation has similar problems to those of kurtosis, particularly the lack of robustness [3].

A more robust approximation is

$$J(y) \approx \sum_{i=1}^{p} k_i \left[E\left\{G_i(y)\right\} - E\left\{G_i(\nu)\right\}\right]^2$$

with $k_i$ positive constants, $G_i$ non-quadratic functions, $\nu$ is a standardized Gaussian and $y$ is assumed to be zero mean and unit variance also. This approximation allows the construction of a measure with the properties of being non-negative and equal to zero only for Gaussian distributions. If only one non-quadratic function $G$ is used, then

$$J(y) \propto \left[E\left\{G(y)\right\} - E\left\{G(\nu)\right\}\right]^2$$

Robust approximations of negentropy are obtained by using non-quadratic functions that do not grow too fast. For example, the functions $G_1(u) = \frac{1}{a_1} log\left[cosh\left(a_1 u\right)\right]$, $1 \leq a_1 \leq 2$, and $G_2(u) = -exp\left(-\frac{u^2}{2}\right)$ have been used [3]. ICA algorithms have been developed using these approximations of negentropy as objective functions to be maximized.

### 2.5.5 Mutual Information

Another measure of the dependence between random variables is mutual information. For the random variables $y_1, \ldots, y_N$, the mutual information $I$ between them is defined as

$$I(y_1, \ldots, y_N) = \sum_{i=1}^{N} H(y_i) - H(\vec{y})$$

where $H(y_i)$ is the differential entropy of the individual random variables and $H(\vec{y}) = H(y_1, \ldots, y_N)$ is the "joint" differential entropy. $I$ is non-negative and zero if and only if the variables are statistically independent [3].

ICA can be defined by the mutual information. Given the data vector $\mathbf{X}$, the goal is to find an invertible transformation $W$ that minimizes the mutual information of the transformed components $s_i$, in the model $\mathbf{S} = W\mathbf{X}$. It turns out that ICA estimation by minimizing mutual information is equivalent to maximizing the sum of nongaussianities of the estimates when those estimates are constrained to be uncorrelated [3]. This is

due to the following relationship between negentropy and mutual information:

$$I(y_1, \ldots, y_N) = C - \sum_{i=1}^{N} J(y_i)$$

where it is assumed that the $y_i$'s are uncorrelated and of unit variance. $C$ is a constant. It is worth mentioning that the uncorrelatedness constraint is not necessary, but it simplifies the computations. As it has been mentioned before, ICA algorithms include pre-processing steps of centering and whitening of the data, precisely to simplify the calculations.

### 2.5.6 Distance Correlation

Distance correlation ($dCor$) is a measure of dependence between random variables $y_1$ and $y_2$. It satisfies $0 \leq dCor(y_1, y_2) \leq 1$, and $dCor(y_1, y_2) = 0$ only if $y_1$ and $y_2$, are independent. This measure can be extended to random vectors from an arbitrary dimension, assuming they have finite first moments. The formal definition, empirical equations, properties and corresponding proofs can be found in [36].

### 2.5.7 Kernel Generalized Variance

The Kernelized Canonical Correlation (KCC) is a regularized estimate of the spectral norm of the correlation operator [37]. The Kernel Generalized Variance (KGV) is an extension to the KCC, which has a close relationship to the mutual information, and is a kernel based measure of independence [38].

### 2.5.8 Algorithms for ICA

A number of algorithms for ICA have been developed over the past decades. As will be explained in chapter 5, we perform experiments comparing several ICA methods. Here we briefly describe some of them and provide corresponding references.

One of the first algorithms for blind identification is the Joint Approximate Diagonalisation of Eigen-matrices (JADE) [39, 40], which is specifically a statistically technique, based on high order contrasts. A very popular algorithm is FastICA [41, 42],

a fixed-point iteration algorithm originally based on kurtosis, but later generalized for other contrast functions. An algorithm based on estimates of entropy called Robust, Accurate, Direct ICA aLgorithm (RADICAL) was presented in [43]. Later, the Mutual-Information-based Least-dependent Component Analysis (MILCA) was introduced [44]. A few Kernel based methods have also been proposed [38, 45–47]. Those methods employ criteria to measure dependence using the spectrum of a covariance operator between mappings of the variables to high dimensional feature spaces. Some extensions or modifications to previous methods have also been proposed. For example, EFICA [48] is a variant of FastICA that attains the Cramer-Rao lower bound, a theoretical limit for the residual error variance.

## 2.6 Applications of Source Separation

As we have mentioned, blind source separation (BSS) defines a class of applications for the ICA model. A typical example is the cocktail party problem described in section 2.1, but BSS-ICA has also been used for separation of electroencephalographic (EEG), magnetoencephalographic (MEG) and electromyographic (EMG) data [3, 4, 13, 15, 32].

In financial data, there are situations in which parallel time series are available, such as currency exchange rates or daily returns of stocks, that may have some common underlying factors. In [12], for example, ICA was used for decomposing parallel financial time series of weekly sales into basic factors. The cash flow of several stores belonging to the same retail chain was investigated, trying to find the fundamental factors common to all stores that affect the cash flow data. The effect of the actions taken at the individual stores and in its local environment could therefore be analyzed.

Noise reduction in natural images is another application discussed in [3]. A noise cleaning result is presented there, comparing the results of the Sparse Code Shrinkage method and classic wiener filtering to an ICA based noise filter.

ICA has also been used for feature extraction [4]. The columns of the mixing matrix $A$ would represent features, and $s_i$ would be the coefficient of the $i^{th}$ feature in an observed data vector $\mathbf{X}$.

Other areas where ICA can be applied include telecommunications, psychology and other social sciences. ICA could be considered in applications where projection pursuit and factor analysis are used.

# Chapter 3

# Proposed Methods

This research work aims at developing pattern recognition frameworks through Source Separation. Our first contribution is the development of a method named Guided Under-determined Source Signal Separation (GUSSS). This method handles the extreme case of under-determination where the number of sensors is actually equal to one – i.e. $M = 1$. Unlike BSS, where the source signals to be separated are unknown, in the proposed method, it is assumed that the source signals are one of a number of expected *signatures*, hence the term "guided" in GUSSS.

In this chapter we present the GUSSS method and a feature derived from it, the GUSSS ratio. We then present classification frameworks that we have applied in different areas using GUSSS. The results obtained so far will be presented in chapters 4 and 5.

## 3.1 GUSSS and GUSSS Ratio

In order to explain the proposed method, let $x$ be a linear combination of $N$ independent components. That is, $x$ represents an observed signal from a single sensor of a given system. Such a system can be, for instance, an Improvised Explosive Device (IED) detector trying to determine the presence of a hazardous substance, or a classifier of electrical signals generated by muscle activity. Next, let $s_p$ be a particular known component, or signature, that the system is trying to identify within the observed

signal $x$. Since the sensor captures not only $s_p$, but also various other components $s_i$, we can write:

$$\begin{aligned} x &= c_1 s_1 + c_2 s_2 + \cdots + c_p s_p + \cdots + c_N s_N \\ &= c_p s_p + \sum_{i \neq p} c_i s_i \\ &= c_p s_p + \tilde{s} \end{aligned} \tag{3.1}$$

where $c_i$, $i = 1, \cdots, N$ are unknown mixing coefficients. It is assumed that $c_i \geq 0$. The expression stresses the fact that $x$ can be considered a linear combination of the signature of interest and a set $\tilde{s}$ of scaled signatures of other components. Since initially we are interested in separating or identifying only $s_p$ from the observed signal, it is assumed that $\tilde{s}$ is independent from $s_p$. This assumption is an obvious consequence of the assumption that all $N$ components are independent – i.e. if $N$ components can be regarded as independent, any linear combination of $N-1$ components must also be independent of the remaining one. Moreover, the algorithm for GUSSS will successfully identify $s_p$ within $x$ whenever $c_p \neq 0$. This could mean, for instance, that a chemical compound associated with the signature $s_p$ is present in a scene. So, the question remaining becomes how to determine $c_p$.

As it has just been implied, two situations may arise: the desired signature is indeed present in the mixed signal $x$, or it is not. In order to distinguish between those two situations, the algorithm creates a second synthesized signal $x_p$ by *injecting* a weighted copy of the particular signature $s_p$ into the sensed signal $x$. That is:

$$x_p = w_1 x + w_p s_p \tag{3.2}$$

where $w_1$ and $w_p$ are arbitrarily chosen constants. Substituting eq. (3.1) in eq. (3.2), we obtain:

$$x_p = w_1 \left(c_p s_p + \tilde{s}\right) + w_p s_p = w_1 \tilde{s} + \left(w_1 c_p + w_p\right) s_p \tag{3.3}$$

which now allows us to write two equations instead of one. That is

$$\begin{aligned} x &= \tilde{s} + c_p s_p \\ x_p &= w_1 \tilde{s} + k_p s_p \end{aligned} \tag{3.4}$$

where $k_p = w_1 c_p + w_p$. Finally, we can express these equations in matrix form as

$$\mathbf{X}_p = A\mathbf{S}$$

where

$$\mathbf{X}_p = \begin{bmatrix} x \\ x_p \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & c_p \\ w_1 & k_p \end{bmatrix}$$

$$\mathbf{S} = \begin{bmatrix} \tilde{s} \\ s_p \end{bmatrix}$$

The last step of the algorithm is to solve for $\mathbf{S}$. Since we now have two independent components and two linear equations on $s_p$ and $\tilde{s}$, we can apply any traditional ICA algorithm to separate $s_p$ and $\tilde{s}$ from $x$ and $x_p$. Moreover, a sub product of the ICA algorithm is the mixing matrix $A$, and the coefficients of such matrix can be used to infer whether or not a particular signature was present in the originally sensed signal $x$. For example, if we consider the case when the particular signal $s_p$ is not present in the mixture signal $x$, the mixing coefficient $c_p$ should be in theory zero. On the other hand, if $s_p$ is indeed present in the mixture $x$, that coefficient must be different than zero.

In practice, mainly due to noise, the coefficient $c_p$ is never zero. However, it should be very small (close to zero) whenever the particular signature is not present in $x$ and it should be large otherwise. We define the GUSSS ratio as:

$$r_p = \left| \frac{1}{c_p} \right| \tag{3.5}$$

If the particular signature is present, the GUSSS ratio should be small. If the signature is not present, the GUSSS ratio should be large. Finally, while what constitutes a "large" or a "small" value may not be obvious, it is clear that the derived GUSSS ratio can be used as a criterion for determining whether a particular signature is present or not in the sensed signal. It can become a feature characterizing sensed signals. Note that the use of the absolute value in eq. 3.5 is due to the fact that ICA can estimate the independent components up to a multiplicative sign, as explained in section 2.5.2.

## 3.2 Classification Frameworks Using the GUSSS Ratio

Next we present classification frameworks that use the GUSSS ratio as a feature. These frameworks have already been successfully applied in different areas, as will be discussed in the next chapter.

### 3.2.1 Present or Not: Two-class problems

The simplest use of the GUSSS ratio is to determine if a signature is present or not in a given mixture signal, as suggested in section 3.1. This can be seen as a two-class problem: class "present" and class "not present".

#### 3.2.1.1 Hard Thresholds

Using a hard threshold is a very simple classification method for two-class, one dimensional problems. Figure 3.1 illustrates a set of 1000 GUSSS ratios from one of our experiments in [49]. They belong to 1000 mixtures of Terahertz signals. The first 500 GUSSS ratios, shown in red, belong to the class "present", i.e., the class of mixtures for which a certain signature $s_p$ was present. Similarly, the remaining 500 ratios, shown in blue, belong to the class "not present". The ratios are shown in log scale. The horizontal axis is just the mixture index. As the figure indicates, it would not be difficult to choose a threshold for separating the two classes.

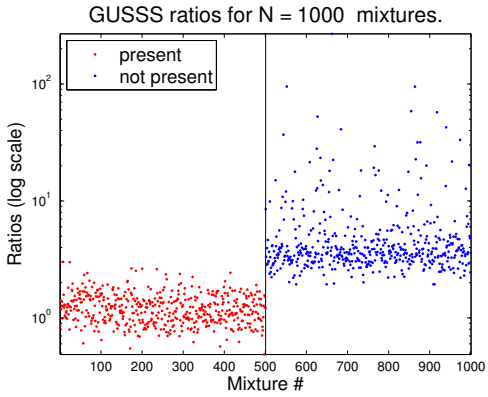



Figure 3.1: Ratios corresponding to 500 mixture signals containing a particular signature of interest, and 500 mixture signals without the particular signature. The vertical line separates those two groups. Y-axis (ratio axis) is in log scale.

One method for obtaining such a threshold is based on geometric means. We can calculate the geometric mean of the GUSSS ratios of each class, and then the threshold would simply be the geometric mean of those two. If the number of training ratios is the same for each class, then the threshold is simply the geometric mean of all the ratios. Using geometric means instead of simple (arithmetic) means fits better the distribution of the ratios.

A second method for obtaining the threshold relies on modeling the distributions of the ratios. For some applications, we noticed that it is better to use the logarithm of the ratios to learn the threshold. That is, given samples from both classes of GUSSS ratios, we calculate their logarithms. We assumed these logarithms follow a specific distribution (e.g. Normal) and for each set of ratios we calculate the parameters (e.g. sample mean and variance), so that we can find the intersections of the two distributions. Figure 3.2 illustrates this idea. Note that the ratio's axis was made horizontal (i.e. rotated with respect to Figure 3.1) for visualization purposes. Now if we let $\rho$ be the value of the intersection of interest (there could be more than one intersection) and since the parameters used to calculate the intersection corresponded to the base-10 logarithms of the GUSSS ratios, the threshold can be obtained from: $thr = 10^{\rho}$.

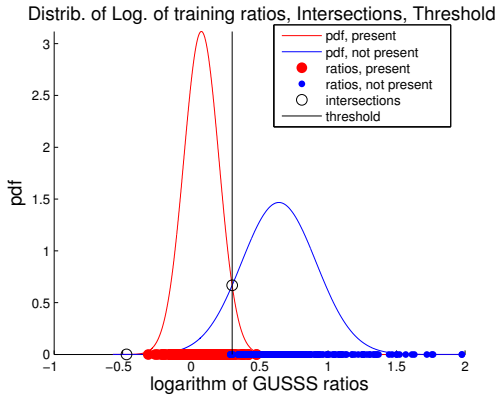



Figure 3.2: Obtaining the threshold approximating the logarithms of the GUSSS ratios via Gaussian distributions. The ratios are the same as those in figure 3.1, but in this figure we plotted their base 10 logarithms, all on the same horizontal axis.

#### 3.2.1.2 Support Vector Machine

Another approach to deal with two class problems is to use Support Vector Machine (SVM). The GUSSS ratios (or their logarithms), and, possibly, other features put together in $d$-dimensional feature vectors, can be used as input for the SVM algorithms. With this approach we don't actually learn a threshold, but we get a set of support vectors that are later used for classifying the testing ratios or, more generally, the feature vectors.

SVM algorithms are convenient for two class problems. They rely on pre-processing the data to represent patterns in a high dimensional space, typically much higher than the original feature space. With an appropriate nonlinear mapping to a sufficient high dimension, data from two categories can always be separated by a hyperplane. SVM is therefore a machine learning method able to handle problems of nonlinear classification. It is based on the rule of structure risk minimum. Therefore, it becomes a decision machine as opposed to Bayesian learning methods that provide posterior probabilities [50].

### 3.2.2 Multi-class problems

We can also use the GUSSS method in problems involving multiple classes, for example, recognizing multiple materials in a chemical mixture, or distinguishing between differ-

ent hand gestures by analyzing muscle activity patterns. Such examples and others will be presented in chapter 4.

#### 3.2.2.1 GUSSS ratios of Multiple Signatures

In order to identify the presence or not of all possible signatures, we can employ an iterative method. That is, first, we assume that the system needs to identify $n$ signatures (we will explain how to obtain the signatures out of training signals shortly, and we will explore the concept further in Chapter 5). Next, from the test signal $x$, we obtain $n$ ratios by injecting iteratively the desired signature into $x$ – equations (3.2)-(3.5) . That is, we find

$$x_p = w_1 x + w_p s_p \qquad for\ p = 1\ to\ n$$

and once again, we apply the ICA algorithm to each

$$X_p = \begin{bmatrix} x^T \\ x_p^T \end{bmatrix} \qquad for\ p = 1\ to\ n$$

to obtain the GUSSS ratios $r_1$, $r_2$, ..., $r_n$. Finally, it should go without saying that if $r_i$ is the smallest of the $n$ ratios found by the GUSSS method, it is likely that the component with the strongest presence in the sensed signal $x$ is that associated to signature $s_i$.

#### 3.2.2.2 Additional Features and Compound Feature Vectors

We can enhance the classification performance of any classifier by adding more features. Next we describe some processing and features that we have used in our experiments.

**Segmentation of the Signals**

To capture the structural information of observed signals, we can divide them into $D$ segments of equal length. The features described next are calculated for each segment of any given signal.

### Mean Absolute Value

One feature commonly used for biomedical signals is the Mean Absolute Value (MAV). The MAV of a signal $x(t)$ is obtained by calculating the average of the absolute values of $x$ at all instants $t$. If the signal is discrete, then

$$MAV = \frac{1}{K}\sum_{k=1}^{K}|x(k)| \tag{3.6}$$

where $K$ is the number of samples that constitute a segment of $x$.

### Zero Crossing

Another time domain feature commonly used is the number of Zero Crossings (ZC), which represents how many transitions from positive to negative (or vice-versa) there are in a signal or a segment of a signal.

Figure 3.3 shows a typical surface electromyographic (sEMG) signal, which represents muscle activity. We will present many results using these bioelectrical signals in the next chapter. The figure shows the signal segmented into 3 parts, and it illustrates the MAV and ZC features.

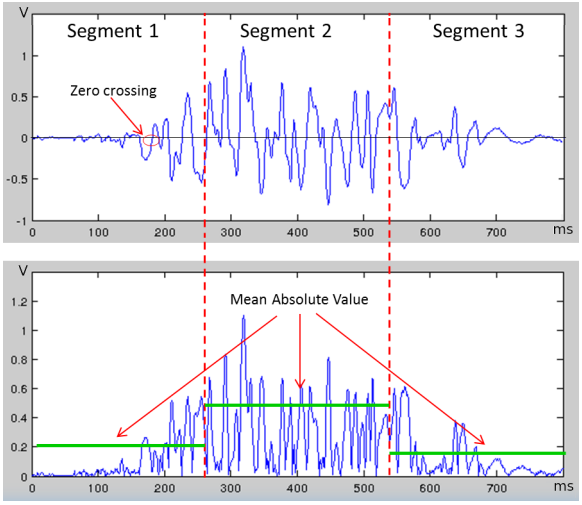



Figure 3.3: A typical sEMG signal segmented into 3 parts. The zero-crossings are indicated in the top figure. The rectified signal and the MAVs of the segments are shown in the bottom figure.

#### Compound Feature Vectors

After all of the features have been extracted, signal $x$ is represented by the following feature vector:

$$\vec{v} = [r_1, \cdots, r_n, m_1, \cdots, m_D, z_1, \cdots, z_D] \tag{3.7}$$

where $r_1, \ldots, r_n$ are the GUSSS ratios for each possible signature. The MAVs and ZCs for each segment of the signal are $m_k$ and $z_k$, respectively, for $k = 1, \ldots, D$. If other features are considered, they would be concatenated to the feature vector in a similar fashion.

### 3.2.2.3 Classification Scheme

Any number of classification methodologies could be used using the features described above. Our original goal was to demonstrate that GUSSS and the GUSSS ratio had a good discriminant power, even with simple classifiers. Here we describe one that we have already used in our preliminary work.

#### Class Signatures

Let us assume that there is a labeled training set with $n \times T$ signals – i.e. $T$ signals from each of the $n$ possible classes (different materials that need to be identified, or muscle patterns or gestures that need to be recognized, etc.). First, a signature for each class is obtained. The current approach is to do an averaging of the training signals grouped per class. That is, all $T$ training signals belonging to the same class $j$ are averaged creating a single signature: $s_j = \frac{1}{T}\left(\sum_{class\, j} x_l\right)$, where $x_l$ is the $l^{th}$ training signal of class $j$. In chapter 5 we discuss other strategies for learning signatures.

#### Distance Classifier

A classifier that we have employed is based on Mahalanobis distances. In addition to the class signatures, from the training signals we can obtain sets of feature vectors like

those in eq. 3.7, per class. Hence, we can calculate the mean vector and covariance matrix of each class, $\vec{\mu}_j$ and $\sum_j$, for $j = 1, \ldots, n$.

To classify a test signal $y$, its feature vector $\vec{v}$ is calculated. Then, Mahalanobis distances to the mean vectors $\vec{\mu}_j$ of the classes in consideration are obtained:

$$d_j = \sqrt{(\vec{v} - \vec{\mu}_j) \ \left(\sum_j\right)^{-1} \ (\vec{v} - \vec{\mu}_j)^T} \ , \ j = 1, \ldots, n \tag{3.8}$$

The test signal is then assigned to that class for which the distance is the closest to zero.

## 3.3 Hierarchical GUSSS

A hierarchical classifier using GUSSS has also been proposed to enhance the original classification approach described above. The proposed framework for the method is illustrated in Figure 3.4 and consists of a two-level hierarchical classifier: 1) a GUSSS-based (distance) classifier; and 2) a Multi-Class SVM.

As it can be seen in Figure 3.4, the first level in the hierarchy involves a number of GUSSS-based classifiers. Basically, these classifiers function as confidence generators, inputing feature vectors extracted from the raw signal and outputting $N$ confidence vectors $\overrightarrow{\lambda}$, where the elements of the vector indicate the confidence that a test signal contains one of the signatures in the tuples – a tuple is a group with an arbitrary number of signatures: e.g. doubles, triples, etc. All of the obtained confidence vectors are concatenated into a second feature vector, which is then input to the classifier at the second level of the hierarchy. The output of the second level classifier is the final class assigned to the observed signal. The following sub-sections describe in further detail the classifiers at each level, as well as their training process.

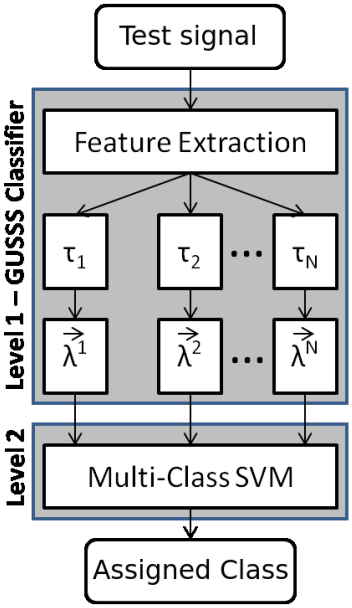



Figure 3.4: Proposed framework. There are two levels in the hierarchy. The first is a GUSSS-based classifier and confidence generator. The second level is a multi-class SVM classifier.

### 3.3.1 Optimal Choice of Tuples

Each GUSSS-based classifier is associated to a tuple of classes, where the sizes and members can be chosen arbitrarily depending on the gestures, user, muscle activity patterns, etc. The rationale behind the tuples is the following: when a large number of $C$ classes are considered at the same time, there might be much confusion between some of the classes. However, it is possible to find subsets of classes for which the confusion between such classes is minimized. So, the goal of the tuples is to allow similar classes to be separated. In a real-time system it is desirable to group as many classes as possible per tuple in order to reduce the complexity of the algorithm. One challenge is to find the right balance of number and size of the tuples to achieve the highest accuracy possible.

The optimal groupings of gestures can be automatically selected by iterating through all possible combinations and performing a modified version of a within class and between class analysis [51] – in the Euclidean space instead of the covariance space.

### 3.3.2 Level 1 Feature Vectors

As mentioned before, the input to each of the GUSSS-based classifiers is a feature vector extracted from the incoming signal. The features used and the way to obtain the signatures and feature vectors is as described in section 3.2.2. The procedure is followed for all $N$ tuples being considered. Let $\tau_i$ denote a particular tuple $i$. As explained in Section 3.2.2.1, the main idea of the GUSSS method is to identify particular signatures within a measured signal. For any given signal $x$, the GUSSS method seeks to identify the presence or not of each possible signature. This is done by iteratively injecting signatures and obtaining ratios for each one of them. For all $n_i = |\tau_i|$ classes in tuple $\tau_i$, the algorithm obtains the ratios $r_1, \ldots, r_{n_i}$. If signal $x$ contains a pattern in class $c$, ratio $r_c$ is expected to be smaller than all other ratios $r_j$, for $j \neq c$.

After all of the other features have been extracted, signal $x$ is represented by the following feature vector:

$$\vec{v}_i = [r_1, \cdots, r_{n_i}, m_1, \cdots, m_D, z_1, \cdots, z_D] \tag{3.9}$$

where $r_1, \ldots, r_{n_i}$ are the GUSSS ratios for each class in tuple $\tau_i$. $m_k$ and $z_k$, for $k = 1, \ldots, D$, are, respectively, the MAVs and ZCs for each segment of the signal. The above feature vectors are extracted for all $T$ training signals in each class and are used to form $\aleph\left(\vec{\mu}_j^i, \sum_j^i\right)$, representing the distribution of class $j$ in the tuple $\tau_i$, where $j = 1, \ldots, n_i$, and $i = 1, \ldots, N$.

### 3.3.3 Distance and Confidence Values

As it was mentioned before, the output of the first level in the hierarchy is a set of confidences that are concatenated to form a second feature vector. The confidences, which are based on Mahalanobis distances, are obtained from each one of the GUSSS-based classifiers.

An input signal $y$ is fed into each one of the optimal tuples described above. For each tuple $\tau_i$, a feature vector $\vec{v}_i$ (eq. 3.7) is calculated. Then, the GUSSS-based classifiers calculate Mahalanobis distances to the mean vectors $\vec{\mu}_j^i$ of the classes in the

tuple $\tau_i$:

$$d_j^i = \sqrt{\left(\vec{v}_i - \vec{\mu}_j^i\right) \ \left(\sum_j^i\right)^{-1} \ \left(\vec{v}_i - \vec{\mu}_j^i\right)^T} \ , \ j = 1, \ldots, n_i \tag{3.10}$$

If, for example, distance $d_j^i$ is small (close to zero), the confidence that signal $y$ belongs to class $j$ would be high.

To obtain the confidence values, the complementary error function is used: $erfc(x) = 1 - erf(x)$. For normal distributions, $erfc\left(\frac{d}{\sqrt{2}}\right)$, can be seen as the probability of a randomly selected sample to fall at a distance of $d$ standard deviations or more from the mean. For instance, $erfc\left(\frac{0}{\sqrt{2}}\right) = 1$, $erfc\left(\frac{1}{\sqrt{2}}\right) = 0.3173$ and $erfc\left(\frac{2}{\sqrt{2}}\right) = 0.0455$. Numerically, these values are appropriate as confidence values for Mahalanobis distances of 0, 1 and 2, respectively. The confidence value function is defined as:

$$\lambda(d) = erfc\left(\frac{d}{\sqrt{2}}\right), \ d \geq 0 \tag{3.11}$$

For the GUSSS-based classifier corresponding to tuple $\tau_i$, the confidence that signal $y$ belongs to class $j$ is given by $\lambda_j^i = \lambda(d_j^i)$. In the end, the classifier produces $n_i$ confidence values: $\vec{\lambda}^i = \left(\lambda_1^i, \ldots, \lambda_{n_i}^i\right)$.

### 3.3.4 Level 2 Feature Vector

After confidence values are obtained for all $N$ tuples, the second feature vector is created as follows:

$$\vec{u} = \left[\vec{\lambda}^1, \ \vec{\lambda}^2, \cdots, \vec{\lambda}^N\right] \tag{3.12}$$

### 3.3.5 Level 2 Classifier: Multi-Class SVM

The final classification method consists of a multi-class SVM. To train the SVM, the $\vec{u}$ feature vectors are computed for all training signals, for all classes. When it comes to classification, an incoming signal $y$ is fed through level 1 in the hierarchy to obtain the confidences and to create the $\vec{u}_y$ feature vector. The latter is fed to the multi-class SVM in order to generate the final class assignment.

# Chapter 4

# Preliminary Results

Over the past few years, we have applied the GUSSS methods described in the previous chapter to both synthetic and real data. We first performed experiments using a collection of sound signatures. Those experiments were designed to test the ability of GUSSS to determine the presence or not of a particular sound signature. Mixture signals were artificially created as linear combinations of the sound signatures. Many experiments were performed varying the intensities of the signatures, adding different noise levels, introducing delays to the signals, etc. The details of those initial experiments can be found in [1].

The results obtained using the artificial mixtures were very promising. Therefore, we decided to apply our method to real data. In [52], we first introduced the GUSSS technique and ratio (sec. 3.1) and we presented results in the context of pattern recognition of muscle activity. This proposed technique was employed in a classification framework for recognition of hand movements, using a single sEMG sensor. In [53], we further tested our method using data from more test subjects and signals generated by eyebrow activity, in addition to the hand gestures. We also presented a power wheelchair control system that relied on a single sEMG sensor and the GUSSS method. Our study on sEMG signals and the area of Assistive Technology (AT) led to the book chapter entitled "*Haptic and Gesture-Based Assistive Technologies for People with Motor Disabilities*" [54].

In [55], we presented and enhanced version of our GUSSS method, the HiGUSSS

(sec. 3.3). That allowed for the classification of a larger number of hand gestures while preserving a high classification accuracy, still relying on a single sensor. A multi channel version of the HiGUSSS method was later applied for pattern recognition of vocal gestures [56]. The goal of that study was to demonstrate that such a system had the potential to recognize and detect real vocal dysfunctions from multiple individuals with high accuracy under both intra- and inter-subject conditions.

We also applied our methods in the area of Terahertz (THz) technology. In [49], we proposed a framework for detecting the presence of chemicals and other materials using their THz signatures, relying on GUSSS. The method was tested using a public THz database, achieving high true positive and true negative percentages. Finally, the HiGUSSS method was proposed for investigating root growth and function of plants by analyzing and classifying THz data [57]. The results were promising for identifying organic materials from potting soil or sand using both THz transmitted and reflected signals.

In this chapter we present the details of our proposed methods in the different areas and applications, as well as the results that we have obtained. For each application, we present relevant background and related work.

## 4.1 Applications to Terahertz Technology

THz technology has been greatly developed over the past couple of decades. The THz electromagnetic radiation lies between light and radio waves ($0.1 - 0.3$ to 10 THz) [58–60]. It has attracted a lot of attention, especially because of its potential in innovative sensing systems, ultra-fast wireless communication systems, devices for medical examinations and detection of hazardous materials [61]. The main advantages of this radiation are: the ability to penetrate many common barrier materials enabling hidden objects to be seen; adequate spatial resolution for imaging or localization of threat objects due to short wavelengths; non-ionising properties; safe to use on people at modest intensities; etc. [62].

Common techniques usually applied to neighboring bands have not shown the same

success for THz radiation [60]. So, for many years there had been this "undeveloped gap" between the electronics of the microwave and the optics of infrared [61]. Some decades ago THz technologies were mainly used in astronomy for searching far-infrared radiation, in laser fusion for the diagnostic of plasmas [63] and in chemistry for spectral characterization of rotational and vibrational resonances and thermal-emission lines of simple molecules [60].

Currently, one of the main uses of Terahertz technology is in spectroscopy [63, 64], which allows investigating properties of materials as a function of frequency. Many methods have been developed for performing THz spectroscopy, including: Fourier transform spectroscopy (FTS); narrowband spectroscopy with tunable THZ source or detector; and THz time-domain spectroscopy (THz-TDS). The latter, a more recent technique, uses short pulses of broadband THz radiation typically generated using ultra-fast laser beams [60]. It has various advantages over FTS, including being able to calculate refractive indices and absorption coefficients from the phase and amplitude of the waveforms, and being able to use information directly from the time domain waveforms [61].

The first demonstration of THz wave time-domain spectroscopy was in the late 1980's, and after that there has been a series of significant advances thanks to improved materials leading to more powerful THz sources and more sensitive detectors [60, 63]. The development of Quantum Cascade Lasers and Terahertz-Range Quantum Well Photodetectors are of great impact [61]. All these advances have provided new opportunities for understanding the THz frequency range, allowing major technical developments, which in turn have greatly extended the potential of THz systems. The use of THz radiation is now open to fields such as physics, chemistry, biology, materials science and medicine [58, 60, 63, 65].

A major interest for THz technology is the detection of explosive devices such as improvised explosive devices (IED). The National Research Council has made recommendations regarding the application of this technology for explosive detection techniques and security screening. These recommendations are based on the fact that most explosives exhibit strong absorption and dispersion in the THz frequency range [63],

and they exhibit characteristic spectroscopic signatures at THz wavelengths which can be used to identify these explosives [59, 62, 66].

Several studies have reported the spectra of energetic compounds like RDX, PETN, HMX and TNT, as well as commercial explosives based on those compounds [59,67–69]. Practical implementation of detection and security systems would need to operate in reflection mode rather than in transmission mode because of the high absorption coefficients of the explosives [59, 66].

There is also an interest to detect and identify other threat materials such as chemical and biological warfare compounds, and illegal substances like drugs of abuse [66]. THz signatures of a number of drugs were identified in [70], and a THz-wave device for nondestructive detection of illicit drugs and hazardous substances hidden in sealed envelopes was proposed in [71]. Detection of hidden objects depends on the transmission of radiation through barrier materials. Envelopes and most of barrier materials such as cloth, paper, cardboard and plastics are semi-transparent to THz radiation [66].

The interest in identifying materials by means of their THz signatures has led to the development of spectra databases. These are valuable to enlarge the range of applications of THz spectroscopy by making it a popular measure of materials [61]. There are several open databases for THz spectroscopy, including the result of the European project "THz-Bridge: Tera-Hertz radiation in Biological Research, Investigation on Diagnostics and study of potential Genotoxic Effects" [72], and the public databases from Tera-photonics Laboratory, RIKEN Sendai and the National Institute of Information and Communications Technology (NICT) [73, 74] – which were used in this research.

### Pattern Recognition of THz Signatures

Most of the work done for identification and classification of materials using THz technology use individual samples of the materials under study. In those scenarios, a few pattern recognition and classification methods can be found in the literature. In [75] the authors proposed a system for identification and classification of four explosive and bio-chemical materials using THz spectroscopy. They used Principal Component Analysis (PCA) for feature dimensionality reduction, a minimum distance classifier

and a neural network based classifier. In [76] the author discusses the Mahalanobis distance classifier, the Euclidean discrimination matrix and Support Vector Machines (SVM) and presents results of case studies for biomedical specimen identification using those methods.

Unlike previous methods, the approach presented here aims to detect materials even if they are mixed together with other materials. That is achieved by using the GUSSS techniques developed in chapter 3, which can be used to separate the different signatures that may be present in a scene. In the following sections we present results in two different scenarios.

## 4.1.1 Results

### 4.1.1.1 Detecting THz Signatures - Data From THz Databases

In [49] we proposed a framework for detecting chemical compounds using their THz signatures. The framework relies on GUSSS to separate mixed signals and determine whether or not a particular chemical is present in a mixture. Our approach aims at detecting materials even if they are mixed together with other materials. Such a framework could be used for detection of Improvised Explosive Devices (IED), for detection of hazardous or illegal substances in airports, etc.

#### Proposed Framework

Figure 4.1 shows a block diagram illustrating the various parts of the system and the flow of the process. We start with a collection of THz signatures, which in our case come from public databases. In our approach we use time domain signals, so we need to transform the THz signatures from the databases (IFFT block in Figure 4.1). Time signatures were used for training the system, for creating test signals, and for obtaining the feature characterizing sensed signals, the GUSSS ratio. As we explained in section 3.2.1, the GUSSS ratio can be used as a criterion for determining whether a particular signature is present or not in the sensed signal. It is calculated for training signals in the system, and it is used to classify the testing signals using a Support Vector Machine

(SVM).

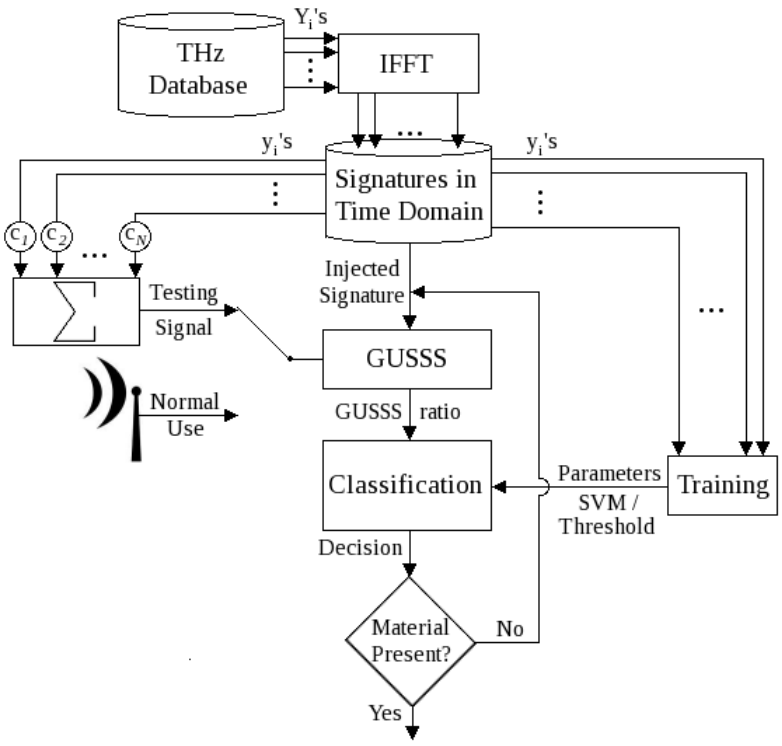



Figure 4.1: Proposed system.

### Training the System

Let $Y_1, \ldots, Y_N$ be a set of THz signatures, and let one of them, say $Y_p$, be the signature corresponding to a particular material or chemical that we are interested in detecting in a sensed signal $x$. Let $y_1, \ldots, y_p, \ldots, y_N$ be the corresponding time domain signatures. These signatures are inputs for the training module shown in Figure 4.1. The output of this module are classification parameters later necessary for testing.

The training module uses the time signatures to create training mixed signals as the following linear combinations:

$$x = c_1 y_1 + c_2 y_2 + \cdots + c_p y_p + \cdots + c_N y_N$$

The mixing coefficients $c_i$ represent the intensities that each individual signature $y_i$ has in the mixed signals. In other words, the training signals simulate sensed signals we would get from a real THz sensor. So, we must create two classes of training signals.

The first class corresponds to signals containing the particular signature $y_p$, i.e. class "present". To ensure that signature $y_p$ is present in the mixture, the corresponding coefficient $c_p$ has to be non-zero. We employed a lower limit ($ll$) and an upper limit ($ul$) that satisfy $0 < ll \leq ul \leq 1$. Then, the coefficient $c_p$ is randomly chosen from a uniform distribution defined over the interval [$ll$, $ul$]. The remaining coefficients are randomly chosen from a uniform distribution over the unit interval [0, 1] and then normalized so that $\sum_{i \neq p} c_i = 1 - c_p$. Note that $ll$ can be interpreted as a minimum % of signature $y_p$ present in the mixtures.

The second class of training signals corresponds to "not present", that is, signals not containing the particular signature $y_p$. Therefore, the corresponding coefficient $c_p$ is set to zero. The other coefficients are once again randomly chosen from the unit interval and normalized as before.

Next we apply the GUSSS algorithm described in section 3.1 to all training signals. We obtain GUSSS ratios labeled "present" and "not present". Finally, we obtain the training parameters – the support vectors – that will be used later to classify the test signal mixtures.

#### Testing the System (for Classification)

The goal of the system is to detect the presence of specific materials among sensed mixtures. So, after training the system for the particular signatures that we want to investigate, we use those same signatures for testing.

Next, given a test signal mixture, we want to determine if a certain material (i.e. a certain signature) is present in the mixture or not. As before, we assume that the test signal mixture is a linear combination of arbitrary time signatures. As Figure 4.1 shows, one of the inputs of the GUSSS module is the test signal mixture. The other input of this module is the signature of the material to be investigated. In other words, if we want to determine if the particular material $P_1$ with signature $y_{p_1}$ is present in the signal mixture, then we would inject a copy of $y_{p_1}$ in the GUSSS module.

The GUSSS ratio obtained from the test signal is then input to an SVM classification function along with the learned SVM parameters. This function determines whether

the given ratio (and therefore the corresponding signature) belongs to class "present" or class "not present". Despite the decision reached by the system for material $P_1$, different materials $P_i$ can be sequentially tested for their presence or absence in the mixture.

### Experiments and Results

The experiments that we conducted rely on THz signatures obtained from two public databases [73, 74]. We ran 18 tests, representing different combinations of materials from the databases. For each of the 18 tests, we selected one material to be the target material – i.e. the one to be detected. We used its signature and the signatures of 7 additional materials to train the system and obtain the training parameters. Then, we created testing mixtures using the same 8 THz signatures plus 2 additional ones. We introduced those two new THz signatures to simulate situations when the target material is mixed with materials for which no training had been provided – i.e. these 2 extra materials had never been seen by the system before the tests.

After selecting the materials for each of the 18 tests, we created 9 different sub-cases representing different potential concentrations of the target material in the total mixture. We did that by randomly selecting the concentration of that target material between a certain lower limit, $ll$, and an upper limit, $ul$. Finally, for each of the 9 sub-cases in each test, we created 500 mixture signals for training and another 1000 for testing. Half of the mixtures corresponded to cases when the target material is present, and the other half to cases when the material is not present. The mixtures were created as linear combinations of all 10 THz signatures in the test.

Figure 4.2 illustrates the behavior of the GUSSS ratios as the lower limit $ll$ varies from 0.1 to 0.9. The figure depicts in red and blue the distributions of the GUSSS ratios corresponding respectively to the cases when the particular material is present and when it is not present (log-scale). The left-hand figure shows the best material – i.e. the material for which the classification rate was the highest. The right-hand figure shows a more typical material. From the figures, we can notice that the GUSSS ratios for the cases when the material's signature is not present remain approximately

within the same range, independent of $ll$ – i.e. the averages of the logarithm of those GUSSS ratios are approximately constant throughout the plot. However, for the cases when the signature is present in the mixtures, the GUSSS ratios are smaller, and their values decrease further with the increase in the value of $ll$ – i.e. with increases in the concentration of the particular material in the mixture. Note how the separation between the set of "not present" cases and the set of "present" cases increases as $ll$ increases, as expected.

Table 4.1 shows the overall average results for all 18 tests. The complete set of tests and more details about the materials used can be found at:

http://vigir.missouri.edu/THz_GUSSS.htm

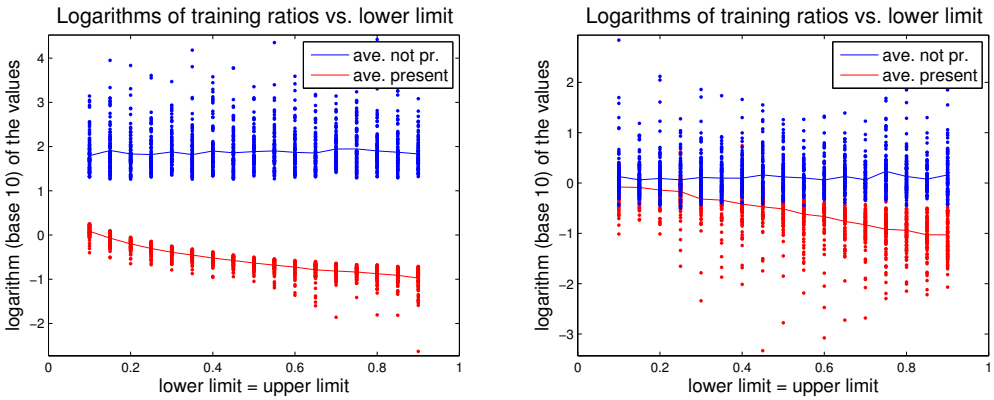



Figure 4.2: Logarithm of GUSSS ratios vs. lower limit ($ll$) of a particular target material concentration %. Red: ratios when the target material is present; Blue, when not present. The continuous lines go through the averages of the log-GUSSS ratios. Left: Best case. Right: Average case.

Table 4.1: Overall average percentages of True Positives, True Negatives and Correct Classification rates (18 experiments).

| $[ll, ul]$ | TP | TN | CC |
|---|---|---|---|
| [0.10, 0.20] | 67.1 | 71.9 | 69.5 |
| [0.20, 0.30] | 77.1 | 79.3 | 78.2 |
| [0.30, 0.40] | 85.3 | 85.1 | 85.2 |
| [0.40, 0.50] | 88.9 | 92.0 | 90.5 |
| [0.50, 0.60] | 93.5 | 94.4 | 93.9 |
| [0.60, 0.70] | 95.1 | 96.8 | 96.0 |
| [0.70, 0.80] | 96.4 | 97.8 | 97.1 |
| [0.80, 0.90] | 97.4 | 98.0 | 97.7 |
| [0.90, 0.95] | 97.5 | 98.3 | 97.9 |

#### 4.1.1.2 Application to Root Phenotyping Using THz Imaging

System Architecture (RSA) is the spatial representation of a plant root system. It plays a vital role in determining the life and growth of plants. Many researchers have long correlated root traits present in the various RSAs to physiological functions of the plant, such as as drought tolerance, carbon allocation, nutrient-acquisition capacity, etc. The key difficulty in measuring and classifying RSAs is the ability to measure root traits without destroying the plants. In order to facilitate this, some researchers opt to growing plants in hydroponic, aeroponic or translucent gel-based media. This approach allows for the inspection of plant roots while still inside a cylinder or solution [77], but they do not recapitulate the nature of RSA in soil while they restrict analyses to young plants. Others are using X-ray CT or MRI based approaches to non-destructively image roots in solid non-translucent media. However, the primary disadvantages of these approaches are the cost, the scanning times, and their limitation to plants grown in pots, where translation of these systems to field conditions does not seem likely. THz imaging is an emerging and significant nondestructive evaluation technique used for analyzing non-conducting materials [64]. THz signals can be captured at the emitter side (reflection) or on the opposite side of the object, at the detector side (transmission) [65, 78]. In both cases, THz signals interact with the different materials in the object

under investigation (e.g. soil, roots, rocks, etc) resulting in multiple reflections or transmissions that are captured by the sensor – this is usually referred to as signal "crosstalk". Those crosstalked signals are examples of mixed signals, and as such, are perfect candidates for our GUSSS based classification methods. Next we present the results from applying our HiGUSSS framework to Pulsed THz imaging of plant samples buried in sand and potting soil.

### Methods

As we have explained before, the idea behind GUSSS is to inject a sought-out signature – e.g. a previously learned signal which is typically reflected off or transmitted through a specific object (root, soil, etc.) – into the classifier and observe the response obtained in terms of the statistical independence of the original signal and the one created by the injected signature. If these two signals are statistically independent, this indicates that the sought out signature was not present (crosstalk) in the original signal. Otherwise, the signature was present and we can classify the signal as containing the sought-out object.

### Experiments and Results

Samples including carrots, sweet potatoes, turnip, rocks and wood pieces were buried in sand and in potting soil and imaged using both transmission and reflection signals. This allowed us to compare the two imaging methods in order to determine their effectiveness at acquiring the shape and size of different buried organic materials.

### Results from THz Transmission

For this experiment, a carrot of approximately $18cm$ in length was placed in a plastic container filled first with sand and then potting soil. The total depth of the material in both cases was approximately $5cm$. Figure 4.3(a) and (b) present photos of the samples investigated in this experiment with THz transmission imaging. Figure 4.3(c) and (d) show measured THz signals with the emitter (detector) located over (under) dry sand only, and over (under) the carrot buried in dry sand, respectively. Figure

4.3(e) shows a THz transmission time domain image for carrot buried in sand, and (f) shows the HiGUSSS classification results. Table 4.2 shows the classification results for the sand and the soil cases.

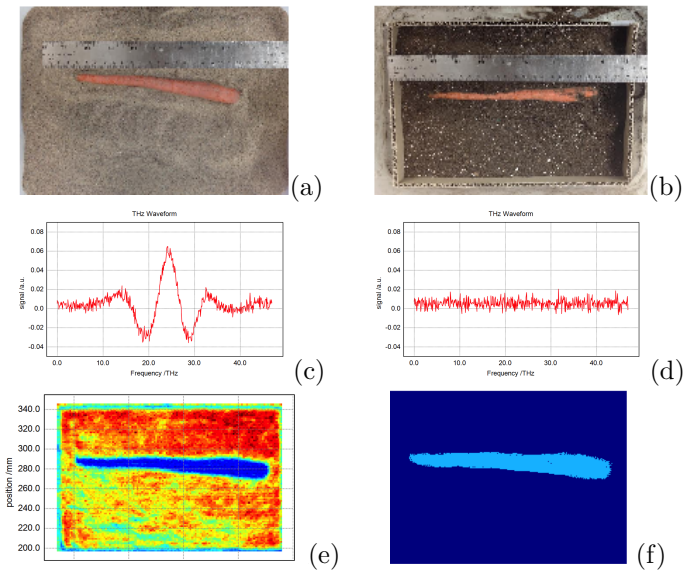



Figure 4.3: Carrot samples in (a) sand and (b) soil; Time domain THz signals transmitted (c) through sand and (d) through the carrot buried in sand; (e) THz transmission time domain images for carrot buried in sand; (f) Results of the HiGUSSS classification (sand).

Table 4.2: Classification Accuracy for the THz transmission experiments.

| Classification Accuracy % | |
|---|---|
| Sand + Carrot | 96.04 |
| Soil + Carrot | 96.06 |

### Results from THz Reflection

To investigate the reflection mode, the THz emitter and detector were both positioned above the sample at a 30-degree angle of incidence relative to the surface. The samples investigated in these experiments included a sweet potato, a turnip, a piece of dry tree

branch and four rocks of different sizes, all covered by $3 - 4mm$ of sand as shown in Figure 4.4(a). The time-domain window of the system was set such that both the sand surface reflection and the reflection from the buried objects could be measured. This is illustrated in Figure 4.4(b), which shows the measured reflected THz signal from both the sand and potato surfaces when emitter/detector are positioned above the potato. Similar to the transmission mode, an image can be formed from the reflected time-domain signals. This is shown in Figure 4.4(c), which illustrates the amplitude of the subsurface reflection. Figure 4.4(d) shows the results of the HiGUSSS framework applied to the THz reflection signals to identify the multiple classes of objects buried in the sand.

Comparison between Figure 4.4(a) and 4.4(d) demonstrate that the results are promising. The HiGUSSS achieved an average accuracy of 91.58% for all five types of objects (potato, turnip, rocks, tree branch and sand). Table 4.3 presents the classification percentages of each of the four types of objects (potato, turnip, rocks, and tree branch) against the sand.

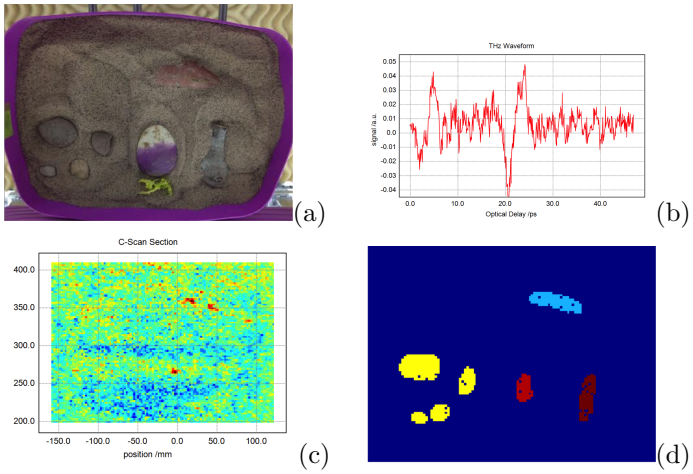



Figure 4.4: (a) Photo of the samples used for the THz reflection test including a sweet potato, a turnip, four rocks and a piece of tree branch (uncovered to show the objects); (b) The time domain reflected signal at a particular point in (c); (c) Time domain THz reflection image of the objects after being completely buried by dry sand; (d) Final classification using the HiGUSSS Framework.

Table 4.3: Classification Accuracy for the THz Reflection experiments.

| Classification Accuracy % | |
|---|---|
| Sand + Potato | 92.50 |
| Sand + Rock | 94.68 |
| Sand + Turnip | 91.43 |
| Sand + Wood | 87.73 |

## 4.2 Applications to Assistive Technology

Robotic Assistive Technology (RAT) is a field that addresses the development of systems to assist people with different levels of impairments in carrying out routine activities [79]. People who have lost limbs or who have suffered strokes or spinal cord injuries, patients engaged in physical therapy, elder people and people with any kind of physical disabilities may benefit from advances in this field [79–81]. The range of assistive devices that the technological advances have made possible is very wide. Research in this field is multi-disciplinary and very active. Even though a lot of advances have been made, there is constant need of finding better, faster, more reliable, more adaptable, user-friendlier and less expensive solutions.

Physical rehabilitation for people who have suffered an injury or have had surgery is essential to recover normal function for daily activities. And assistive technology can help in that process. For instance, [82] presents a knee orthopaedic device to illustrate how robotic technology can improve the outcome in knee rehabilitation. Unfortunately, a lot of people do not have the possibility to fully recover. For instance, people with degenerative diseases like progressive muscle dystrophy, Multiple Sclerosis (MS), Alzheimer's and Parkinson's diseases; elder people with weakened muscles or degenerative joint diseases; blind, deaf or amputees, etc. There is a need for devices and technologies to assist them. There are a lot of commercial devices available today, but research continues. To illustrate, in [83] an electric lifting chair meant to aid in standing up and sitting down is assessed; [84] presents a robot assisted rehabilitation protocol designed to treat cerebellar and motor symptoms in subjects with MS; [85] introduces an exoskeletal meal assistance system for progressive muscle dystrophy patients; and [86] studies haptic and auditory interfaces that can help blind people to

interact with computers.

Intelligent AT systems and devices rely on different kinds of signals or features coming from the environment or from the human users themselves. Temperature, velocity, acceleration, orientation, force, torque and many other quantities are detected and used. Audio and video signals are captured and analyzed, too. Some other signals such as electrical signals generated by brain, cardiac or muscle activity are also used for evaluating patients and for controlling AT devices. The methods presented in this work have been applied to those electrical signals, specifically the ones generated by muscles. In the next section we present a more specific background on them.

### 4.2.1 Background on EMG

Electromyographic signals collected at the surface of the skin (sEMG) have been used in many applications, including rehabilitation, prosthesis, computer interfacing, wheelchair control, etc. [87–89]. When it comes to rehabilitation, more specifically for power wheelchair control, EMG signals have been often used as on/off switches. In those cases, menu driven approaches [80], finite state machines [90], and combinations of multiple muscles and sensors [91] are common techniques employed to provide multiple dimensions in the operation to the interface.

Several sEMG-based systems that rely on more elaborated pattern recognition of EMG signatures have also been proposed, including for exoskeleton robotics [92], and they vary widely in terms of: the classification approach employed; the feature selection criteria; and the number of sensors used. But, again, they constantly require multiple sensors and much investment on the feature extraction and classification algorithms. Our goal in this work is instead to present a much simpler and yet effective technique using a single EMG sensor, allowing for other muscles and sensors to be used in other interfaces or to add modalities of operation to the interface.

As we mentioned above, sEMG-based systems vary widely and in terms of the classification algorithm. Artificial Neural Networks (ANN) [93–95], Fuzzy Logic and Fuzzy Control systems [32,94], are possibly the most common methods used to classify muscle activity – i.e. classify motor unit action potentials trains (MUAPT). The ability to

recognize MUAPT can be applied, for example, to hand gesture recognition, control of electro-mechanical prosthesis, computer mouse movement, etc. [89]. One such example can be found in [94], where an ANN was compared to a Fuzzy Inference System (FIS) for classification in a hand prosthesis control.

In another work presented in [95], several techniques for classification were employed in order to identify hand gestures using sEMG signals extracted from the forearm of human subjects. The authors reported good performance using ANN, Random Forest (RF), 1-Nearest-Neighbor (1NN), Support Vector Machine (SVM), Decision Tree (DT) and Decision Tree with Boosting (DT/B) as some of the different classification techniques used. In that case, the ANN approach presented a better performance than the other methods.

In terms of feature selection, the features can be extracted from the *time* or the *time-frequency* domains [89, 93, 94]. These features typically include: number of Zero Crossings (ZC), Mean Absolute Value (MAV), Slope Sign Changes (SSC), coefficients of Auto-regressive models (AR) [93,94]; Absolute Maximum/Minimum, Maximum minus Minimum, Median Value (Med), Variance, Waveform Length (WL) [89]; coefficients of the Short Time Fourier Transform (STFT) [89]; Wavelets Transform (WT) [88, 89], etc.

Given the wide range of features and their large dimensionality, many systems also employ dimensionality reduction techniques to the set of features. In those cases, Class Separability (CS), Principal Component Analysis (PCA), Analysis of Variance (ANOVA) or Multivariate ANOVA (MANOVA) are the techniques frequently used. In [94], for example, the authors developed a feature selection that employed CS and PCA for dimensionality reduction. In that system, as well as in [95] where ANOVA was the technique of choice, the main concern is always to reduce dimensionality without affecting the classification in a significant manner.

Finally, in terms of number of sensors used, most systems developed to date have made use of two or more sEMG signals derived from multiple sensors. For example, in [94], the authors reported using only two differential sEMG electrodes placed on the forearm of the test subjects. As we mentioned earlier, their system used multiple

features and a FIS+PCA classifier to achieve 83% accuracy. A better performance (93.3%) was obtained in [95], but with the cost of relying on more sensors – 5 to be more specific – and using ANN as the classification algorithm.

As it can be inferred from the literature review, the use of multiple electrodes and of sophisticated classification algorithms help coping with a major disadvantage of surface EMG: the occurrence of crosstalk from adjacent muscles [87]. It is exactly this crosstalk of MUAPTs that makes the use of a single sensor a quite challenging problem.

Several systems have been proposed to control wheelchairs using EMG signals. To illustrate, in [96], the authors developed a wheelchair controller for users with high-level spinal cord injury. They used Fuzzy Min-Max Neural Networks to classify forward, left and right movements, and rest. And they measured EMG signals from muscles in each side of the neck. In [91] the authors used EMG signals from the neck and the arm muscles to implement a "joystick-like" model to control a wheelchair. And in [90] the authors presented a hands-free control system based on EMG signals recorded from eyebrow muscle activity for directional control, and electro-oculography signals detected from eye movements for speed control.

Once again, systems like the one above either rely on very complex processing and classification algorithms, or use the EMG signals mainly as on/off switches. Moreover, they seek patterns based on the simple presence and the frequency of the activation. We have used our proposed GUSSS based frameworks described in chapter 3 for recognizing muscle movements or "gestures". We use GUSSS on the crosstalked MUAPTs signals from a single sEMG electrode. Unlike other methods based on ICA [32], our method relies on a single sEMG source.

As an example of a real life RAT system, we incorporated our gesture recognition method into a power wheelchair, allowing it to be driven through different muscle movements.

### 4.2.2 Results

We first introduced the GUSSS method and GUSSS ratio in [52]. The proposed classification framework was demonstrated for detecting specific hand gesture signatures using

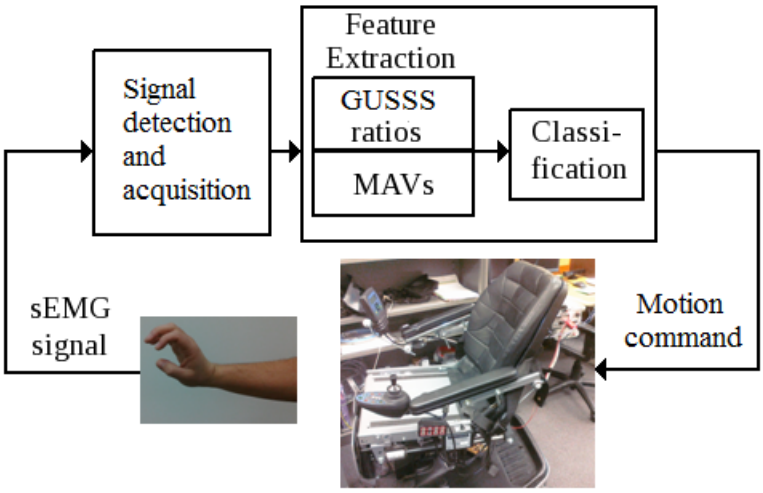



Figure 4.5: Original framework for the power wheelchair system.

a single sEMG source. Compared to other systems found in the literature which use multiple sEMG sources for classification, our proposed framework employed a much simpler classifier – a distance classifier (sec. 3.2.2.3) – using only two features, the GUSSS ratio and the MAVs. No segmentation of the signal was used. Yet, the classification accuracy obtained with our method, 85% for three different gestures, was quite comparable to previously reported methods. At the same time, it still required fewer sensors, fewer features, and a very straightforward classification algorithm. The tests in that first work were done offline.

In [53], further offline tests were performed, including more test subjects and signals generated by eyebrow movements, in addition to arm muscle activity. Furthermore, we showed how the technique can be employed to control of a power wheelchair. Online tests were performed using the wheelchair system. Figure 4.5 shows the original framework.

As we mentioned in the introduction of this chapter, due to our work in the area of AT, we wrote a book chapter entitled "*Haptic and Gesture-Based Assistive Technologies for People with Motor Disabilities*" [54], where we presented a survey on methods and algorithms used for haptic and gesture based technologies. We also presented our own results thus far, and we offered our particular views on the future of assistive

technologies in general.

In [55] we introduced the HiGUSSS method described in Section 3.3 and we presented the results of applying the method to a larger number of hand gestures. We aimed at improving the performance of our method both in accuracy and in the number of gestures to be recognized, while still using a single sEMG signal. We added an additional feature – the ZC – and we segmented the signals, keeping the distance classifier that we had used before. Next we present more details of this work, as well as the results obtained.

### Goals of the Experiments

The goals of the experiments presented in [55] were the following: 1) contrast with [53] for the same number of gestures (4) and with more gestures (5); 2) compare HiGUSSS with non-hierarchical methods; and 3) investigate how the accuracy varies as the number of gestures increases.

### Data Collection

Seven test subjects were asked to perform at least 100 repetitions of each of the five gestures shown in Figure 4.6. The sEMG signals of interest, i.e., the ones to be associated with each gesture, are those generated during the transition from a resting position to the actual hand gesture and back to the resting position. Each subject performed all gestures at the same level of effort and rested between gestures.

A single pair of sEMG electrodes was placed near the Extensor Carpi Radialis Longus muscle along the forearms of the human subjects to collect the performed gestures. A reference (ground) electrode was also placed on the wrist of the opposite arm of the subjects. The raw sEMG signals were amplified ($\times$2000) and low-pass filtered (1 kHz) using a GRASS amplifier (model 15$A$54) and sampled using a National Instruments (NI) digitizer at 4 kHz. The signals were divided into 3 segments (i.e. $D = 3$), and the size of the tuples was set to 2 for all tuples (i.e. $n_i = 2$).

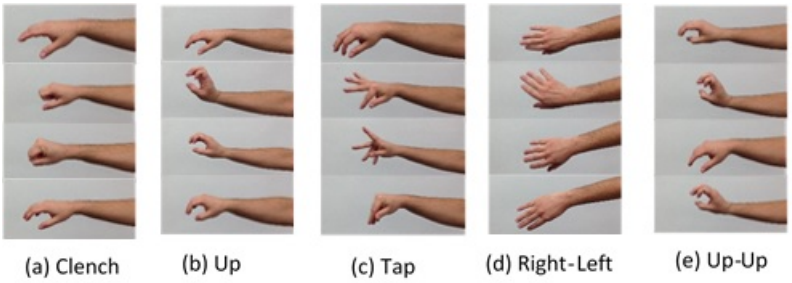



Figure 4.6: Hand gestures considered: a) "clench", b) "up", c) "tap" (finger tapping), d) "right-left", and e) "up-up" (done quickly). The figure shows the transition from the resting position to the gesture and back to the resting position.

### Results using HiGUSSS

To contrast with the work in [52, 53], we first tested the HiGUSSS method with 4 gestures and later with 5 gestures. For each experiment, a 10-fold cross validation was performed. Each time 90% of the signals of all the gestures were used for training. The remaining 10% of the signals were then classified as described in Section 3.3. Tables 4.4 a) – c) show the results using 4 gestures and Tables 4.5 a) – c) show the results using all 5 gestures. The results shown reflect the test subjects with the highest accuracy, with the lowest accuracy, and the average for all test subjects. The confusion matrices show the average percentages over the 10-fold tests. The average correct classification percentages are also presented at the bottom of each table.

Table 4.4: HiGUSSS: Confusion matrices for 4 gestures. The values are average percentages over a 10-fold cross validation.

| | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | clench | up | tap | up-up |
| True hand gest. | clench | 94.0 | 0.0 | 0.0 | 6.0 |
| | up | 0.0 | 96.0 | 0.0 | 4.0 |
| | tap | 0.0 | 0.0 | 99.0 | 1.0 |
| | up-up | 2.0 | 5.0 | 0.0 | 93.0 |
| Correct classification: 95.5% | | | | | |

a) Best case

| | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | clench | up | tap | up-up |
| True hand gest. | clench | 91.0 | 9.0 | 0.0 | 0.0 |
| | up | 6.0 | 93.0 | 0.0 | 1.0 |
| | tap | 3.0 | 6.0 | 88.0 | 3.0 |
| | up-up | 0.0 | 5.0 | 1.0 | 94.0 |
| Correct classification: 91.5% | | | | | |

b) Worst case

| | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | clench | up | tap | up-up |
| True hand gest. | clench | 88.8 | 6.1 | 2.5 | 2.7 |
| | up | 4.4 | 94.1 | 0.4 | 1.1 |
| | tap | 2.5 | 1.5 | 94.5 | 1.5 |
| | up-up | 3.7 | 2.6 | 0.8 | 92.9 |
| Correct classification: 92.6% | | | | | |

c) Average over 7 subjects

Table 4.5: HiGUSSS: Confusion matrices for 5 gestures. The values are average percentages over a 10-fold cross validation.

| | Assigned gesture | | | | |
|---|---|---|---|---|---|
| | clench | up | tap | up-up | right-left |
| clench | 94.5 | 0.9 | 0.0 | 0.0 | 4.5 |
| up | 2.7 | 96.4 | 0.9 | 0.0 | 0.0 |
| tap | 0.0 | 0.0 | 97.3 | 0.9 | 1.8 |
| up-up | 0.0 | 2.7 | 0.9 | 96.4 | 0.0 |
| right-left | 3.6 | 0.9 | 1.8 | 0.9 | 92.7 |
| Correct classification: 95.5% | | | | | |

a) Best case

| | Assigned gesture | | | | |
|---|---|---|---|---|---|
| | clench | up | tap | up-up | right-left |
| clench | 88.0 | 12.0 | 0.0 | 0.0 | 0.0 |
| up | 5.0 | 89.0 | 0.0 | 1.0 | 5.0 |
| tap | 3.0 | 3.0 | 86.0 | 0.0 | 8.0 |
| up-up | 0.0 | 2.0 | 1.0 | 88.0 | 9.0 |
| right-left | 1.0 | 8.0 | 6.0 | 3.0 | 82.0 |
| Correct classification: 86.6% | | | | | |

b) Worst case

| | Assigned gesture | | | | |
|---|---|---|---|---|---|
| | clench | up | tap | up-up | right-left |
| clench | 86.3 | 5.9 | 2.1 | 2.6 | 3.2 |
| up | 5.1 | 92.4 | 0.1 | 0.9 | 1.4 |
| tap | 3.6 | 0.6 | 93.4 | 0.6 | 1.8 |
| up-up | 3.4 | 1.7 | 0.8 | 91.2 | 2.9 |
| right-left | 5.0 | 2.3 | 1.6 | 2.6 | 88.5 |
| Correct classification: 90.4% | | | | | |

c) Average over 7 subjects

### Results - HiGUSSS vs. Non-Hierarchical Classifiers

The HiGUSSS was also compared to two other approaches for the same features except for the confidences, since the two approaches used were non-hierarchical. So, instead of using a pairwise approach, features were extracted from the training signals for all classes simultaneously. In other words, the feature vectors are similar to the level 1 vectors described in Section 3.3.2, with GUSSS ratio values for all $C$ classes – i.e. $C$

gestures.

For the distance classifier, the classification was obtained by selecting the smallest Mahalanobis distance from the input signal's feature vector with respect to the means of the class distributions.

Tables 4.6 a) and b) show the correct classification percentages for all 7 test subjects using the three classification methods. Despite the small differences, overall, both the distance classifier and the SVM are outperformed by the proposed hierarchical approach. As pointed out in [52,53], we attribute the good performance of all classifiers to the GUSSS ratio as a feature in all methods.

Table 4.6: Classification accuracies for 7 test subjects. The values are average percentages over a 10-fold cross validation (105 signals per gesture).

| | Distance | SVM | Hierarchical |
|---|---|---|---|
| Subject 1 | 93.5 | 94.0 | 95.5 |
| Subject 2 | 96.8 | 95.5 | 95.2 |
| Subject 3 | 86.8 | 86.3 | 91.5 |
| Subject 4 | 97.9 | 97.5 | 96.4 |
| Subject 5 | 89.7 | 88.4 | 90.0 |
| Subject 6 | 92.0 | 88.9 | 89.1 |
| Subject 7 | 89.5 | 90.0 | 90.5 |
| Overall | 92.3 | 91.5 | 92.6 |

a) Four gestures

| | Distance | SVM | Hierarchical |
|---|---|---|---|
| Subject 1 | 91.2 | 91.4 | 93.2 |
| Subject 2 | 91.3 | 93.1 | 95.3 |
| Subject 3 | 81.8 | 83.0 | 86.6 |
| Subject 4 | 94.2 | 93.1 | 95.5 |
| Subject 5 | 85.8 | 84.9 | 86.5 |
| Subject 6 | 87.8 | 88.7 | 87.6 |
| Subject 7 | 87.4 | 90.7 | 87.8 |
| Overall | 88.5 | 89.3 | 90.4 |

b) Five gestures

### Results - Accuracy vs. Increasing Number of Gestures

One last experiment was performed using up to 9 gestures to evaluate the effect on the classification accuracy when increasing the number of gestures. Data for 9 gestures were collected from one test subject by the same process described above. The gestures included the 5 shown in Figure 4.6 as well as four new gestures: "Down", "clench-clench", "open", and "rotate". To complete the "down" gesture the subject bends the wrist towards the ground. To complete the "clench-clench" gesture the subject quickly does the clench gesture shown in Figure 4.6 two times. To complete the "open" gesture the subject opens the hand extending all of the fingers. To complete the "rotate" gesture, the subject rotates the hand clockwise.

Table 4.7 shows the classification percentages for a test subject using the three classification methods. It's important to notice that as the number of gestures used increases, the gap in classification accuracy between the proposed method and the other two classifiers grows. This trend is illustrated in Figure 4.7 which shows the classification accuracy plotted against the number of gestures used. A $T$-test for statistical significance of this result was performed and the hierarchical classifier outperforms the SVM with $t = 4.11$ and $df = 6$ for $p < 0.05$, and the distance classifier with $t = 4.07$ and $df = 6$ for $p < 0.05$.

Table 4.7: Classification accuracy for a single test subject. The values are percentages over a 10-fold cross validation of 105 signals per gesture.

| | Distance | SVM | Hierarchical |
|---|---|---|---|
| 3 Gestures | 97.3 | 99.0 | 98.3 |
| 4 Gestures | 97.7 | 94.7 | 98.5 |
| 5 Gestures | 96.0 | 93.4 | 96.8 |
| 6 Gestures | 92.1 | 90.0 | 94.4 |
| 7 Gestures | 88.2 | 87.5 | 91.3 |
| 8 Gestures | 84.5 | 82.3 | 89.1 |
| 9 Gestures | 83.6 | 79.0 | 86.5 |

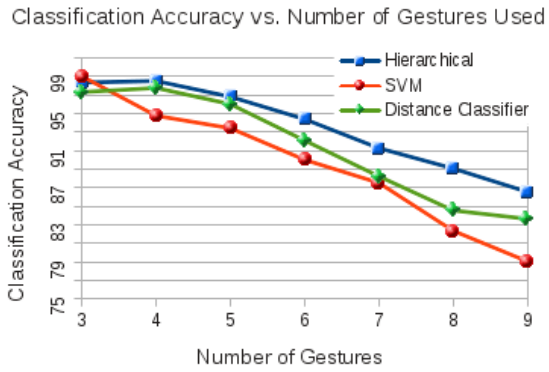



Figure 4.7: Classification accuracy vs. number of gestures for the proposed hierarchical approach, the SVM, and the distance classifier, tested with data from one subject.

The addition of ZC and segmenting the signals had a positive impact in the classification performance, as expected. The distance classifier achieved higher classification accuracies in the latest work compared to the earlier results – overall averages of 92.3% for 4 gestures and 88.5% for 5 gestures. During our tests, we noticed that certain gestures were very distinguishable from each other, and certain other gestures were very confused with each other. This observation motivated the tuples and the hierarchical method, which aimed at minimizing the confusion between those gestures. The results obtained with the hierarchical approach were, in fact, better than with the non-hierarchical classifiers – 92.6% for 4 and 90.4% for 5 gestures.

Even though the improvement gained by HiGUSSS was relatively small, it was interesting to notice that such improvement was comparatively higher with 5 gestures than with 4 gestures. This observation motivated the last experiment, where we examined the effect on the classification accuracy when increasing the number of gestures. As can be seen from Table 4.7 and Figure 4.7, HiGUSSS outperformed the other two methods, and the differences in accuracy were statistically significant, as confirmed by the T-tests results. As expected, as the number of gestures increased, the accuracies for all of the classifiers dropped. However, the classification accuracy for HiGUSSS decreased at a lower rate than for the other two classifiers. In the case of 3 gestures, the accuracy of HiGUSSS was about 1% higher than the accuracy of the distance classifier and less than 1% lower than that of the SVM – yet almost perfect. In the case of 8 gestures, HiGUSSS was almost 5% above the distance classifier and almost 7% above

the SVM. And with all 9 gestures, HiGUSSS was 7.5% above the SVM, though it was only about 3% above the distance classifier in this case.

The results presented here demonstrate the discriminant power of HiGUSSS. Its better performance compared to the other classifiers is because the hierarchical method employs tuples of gestures instead of comparing each gesture against every other gesture. This is more noticeable as the number of gestures increases.

## 4.3 Applications to Otolaryngology

Occupational voice users are at the highest risk for voice disorders largely due to the extraordinary vocal load placed on the laryngeal system while exercising their occupation [97]. Classic symptoms are hoarseness, vocal effort, and vocal fatigue, which are related to vocal hyperfunction [97]. In turn, vocal hyperfunction may lead to phonotrauma – e.g. vocal nodules – or to primary Muscle Tension Dysphonia (MTD) – i.e. excessive or dysregulated laryngeal muscular activity underlying the vocal changes [98]. So, the ability to differentiate normal and abnormal patterns of laryngeal muscular activity in daily life could improve our ability to detect and understand the pathophysiological processes leading to MTD, and thereby improving the diagnosis of this voice disorder.

In parallel to that, it is well known that as muscles contract, they undergo changes in electrical potentials, which can be monitored by electromyographic (EMG) devices. When studying a single muscle, the optimal signal to noise ratio is typically obtained when the electrodes are placed inside the muscle – a technique available in the healthcare or laboratory setting, but with limited use in people's everyday lives. A less invasive strategy is to place surface EMG (sEMG) electrodes on the skin near the muscle(s) of interest. Recently, the interest in many areas such as human-computer interfacing [99], prosthesis [100] and even voice pathology [101, 102] has fomented development of devices that can monitor muscle activity, and systems like the Delsys' Trigno, Great Lakes Neurotechnologies' BioRadio, and the Shimmer's Shimmer3 wearable sensors are just a few of the many examples in the market today.

Given the benefits of detecting MTD and the proliferation of sEMG devices in the context of other muscle-related dysfunctions (ALS, Cerebral Palsy, etc. [54]), it stands to reason that sEMG can be a powerful, non-invasive and well-suited tool also in the study of MTD. In that sense, even though reliable detection of sEMG signals in extralaryngeal muscular activity that can be associated to high risk of voice disorders is a non-trivial and scarcely investigated area, it should also be the foundation to study differences between normal and maladaptive muscular activity during voice production for speech.

In the past decade, research intensified the development of devices suitable for ambulatory monitoring of daily voice use. Commercial systems were made available to monitor vocal duration, voice intensity, and voice fundamental frequency ($f_0$) using accelerometers, microphones and frequency transforms [103–105]. Some of these non-invasive monitoring system even provide biofeedback capabilities and have been ported to smartphone platforms [106]. However, the richness and complexity of vocal patterns during speech go well beyond what can be captured by microphones and inertial sensors. In that sense, and with limited use to extralaryngeal activity, one of the latest developments was a smartphone-based Vocal Health Monitor in which collection of frequency and inertial data was calibrated to aerodynamic parameters, in particular glottal air flow [106–109]. On the other hand, vocal effort is thought to be partially the result of compensatory extralaryngeal activity to produce stronger or more consistent voice during vocal fatigue [110]. Hence, even though the detection of vocal effort and fatigue as early signs of MTD is often elusive in the isolated screening or clinical setting [107], classifying sEMG signals can help to answer unique research questions about magnitude and pattern of that same extralaryngeal activity and their correlation with MTD.

Finally, while we agree with recent statements that sEMG has not reached its full potential for application to clinical and basic research in voice, speech, and swallowing [102], ambulatory monitoring of voice using sEMG on extralaryngeal muscles can be an innovative approach to advance our understanding of vocal hyperfunction and ultimately to monitor its occurrence in heavy voice users. Indeed, the broader and

more pertinent issue at hand is to further research on how to best process and analyze data from any voice ambulatory monitoring system and to determine the data's clinical utility [105, 107], which is also the focus of this research, in the context of sEMG signals.

As alluded to earlier, sEMG is a less invasive strategy than its non-surface counterpart, with more real-world practicality. However, it comes at the expense of noisier signals and exacerbated occurrences of crosstalk between adjacent electrodes. That is because the biologic functions that are subserved by muscular activity do not result from the action of a single muscle, but from the activity of several muscles working in a coordinated fashion. Moreover, when it comes to recognizing the crosstalk patterns of muscle activity in a reliable, accurate and robust manner, much remains to be done. In [111], an ambulatory sEMG device named ECHO (EMG multi-Channel Hardware for Otolaryngology) was proposed to log sEMG data from multiple differential sEMG sensor channels. The system was connected to the anterior neck of the subject since many complex physiological motor functions underlying voice, speech, and swallowing occur within the neck. Also, the muscles in this area are located relatively close to the skin and are quite appropriate for sEMG. The goal in [111] was to demonstrate that: 1) an ambulatory sEMG device can help to build our understanding of complex laryngeal patterns underlying voice for speech and non-speech vocal behaviors through sEMG signals; and 2) the neck offers an excellent location from which to capture such signals. The HiGUSSS was then tested for a single test subject and it achieved a classification accuracy of over 90% for six gestures.

So, in [56], two new research questions were raised: 1) whether sEMG devices can reliably associate a larger number of patterns of extralaryngeal muscle activity with voice tasks underlying speech and non-speech behaviors; and 2) whether they can differentiate between multiple vowel sounds produced in a normal manner compared with a pressed (low air flow) manner for intra- and inter-subject testing. The answer to these questions can lead to a method applicable clinically to the *detection* of normal and maladaptive extralaryngeal patterns associated with voice problems.

In order to answer these questions, we drastically expanded on the testing and

validation of the system proposed in [111] by: 1) more than doubling the number of test subjects (ten) with different ages and genders; 2) adding new groups of different gestures with both similar and distinct patterns representing normal and simulated dysfunctional conditions; 3) testing a larger number (ten) of vocal gestures; and 4) creating different test scenarios involving intra- and inter-subject cases. The results presented in [56] show that despite the complexity of the muscle groups on the neck, meaningful detection of vocal dysfunctions through the recognition of sEMG signals is possible, at high levels of accuracy.

### 4.3.1 Multi-Channel Hierarchical GUSSS

For the enhanced version of the Hierarchical GUSSS with multiple channels used in this research, the steps detailed in 3.3 were replicated for each channel, leading to a set of vectors $\overrightarrow{u}$. These channel vectors were then averaged in order to form a single confidence feature vector to serve as the input to the Multi-Class SVM (MC-SVM).

### 4.3.2 Experiments

Expanding on the six gestures collected in [111] for a single subject, for the experiments reported here, a total of ten vocal gestures and one resting condition were collected for ten subjects. The full set of gestures collected for each subject includes: */a/, /a/ pressed, /u/, /u/ pressed, /i/, /i/ pressed, /t/, /s/, cough, and throat clear*. A pressed gesture indicates a simulated vocal dysfunction by using a pressed voice for the corresponding gesture. Vocally healthy subjects were trained on how to simulate these dysfunctional gestures by using a pressed voice, or restricting their air flow during a vocal gesture. This training is described in Section 4.3.2.3.

These ten gestures were then grouped into the following four different test sets. The first test set includes the six original vocal gestures */u/, /i/, /t/, /s/, cough, and throat clear* also found in [111], but this time for ten subjects instead of one. The second test set consists of all ten gestures */a/, /a/ pressed, /u/, /u/ pressed, /i/, /i/ pressed, /t/, /s/, cough, and throat clear*. This set was used to test the ability of the hierarchical approach to classify a large gesture set with high accuracy. The next set

was formed by the gestures */a/, /a/ pressed, /u/, /u/ pressed, /i/, and /i/ pressed*, and it was used to test the ability of the system to classify unique normal and simulated dysfunctional gestures. In other words, to identify a vocal gesture with or without simulated dysfunction, as well as to classify the occurrence of specific vocal gestures. Finally, the gestures */a/, /u/, and /i/* were grouped together into a *Normal* class, while */a/ pressed, /u/ pressed, and /i/ pressed* into a *Pressed* class. In order to further validate the proposed hierarchical approach, a comparison was performed between the proposed method, a distance classifier, and a single-layer MC-SVM classifier.

#### 4.3.2.1 Data Collection

The main goal of all experiments was to validate the claim that meaningful classification can be achieved from extralarygneal sEMG signals of the anterior neck – not only for normal voice production, but also simulated disordered voice production. Therefore, sEMG signals were collected under well-controlled laboratory conditions. The subjects were six males and four females in good health who denied the presence of any voice disorder. Four pairs of sEMG electrodes and a ground electrode were placed as explained in Section 4.3.2.2 and seen in Figure 4.8. Data were collected in an IAC Acoustics audiology booth (New York, New York) in the Department of Communication Science and Disorders at the University of Missouri.

For the sEMG data collection, the test subjects were asked to perform 55 repetitions of each of the ten selected gestures in the following order: */a/, /a/ pressed, /u/, /u/ pressed, /i/, /i/ pressed, /t/, /s/, cough, and throat clear*. Each subject performed all of the gestures of a given type within a 2 second interval per each repetition of a gesture and with a 1 second break between repetitions. After the data for a single gesture were collected, the subject rested for several seconds and drank water as needed before completing the data collection for the next gesture.

During the rest period between repetitions of a gesture, the subject was asked to be as relaxed as possible, and try to minimize any motion in the throat or mouth area. The sEMG signals of interest, i.e., the ones to be associated with each gesture, are those generated during the transition from the resting condition to the actual vocal

gesture and back to resting.

For the experiments presented here, data were collected using a Tektronix MSO 4054 digital oscilloscope with a sample rate of 5KHz. The signals were treated by both digital and analog bandpass filters at 30Hz and 1KHz and were divided into 3 segments (i.e. $D = 3$), as described in Section 3.2.2.2.

#### 4.3.2.2 Electrode Placement

As seen in Figure 4.8, surface electrodes were placed according to established guidelines for sEMG recordings [112] with special consideration of recommendations proposed for voice, speech, and swallowing research [102]. Disposable 10mm Ag/AgCl surface electrodes were placed in bipolar configurations for single differential recordings from the anterior neck musculature. Two identical electrode pairs were placed on the left and right side of the neck to capture suprahyoid (submental) and infrahyoid muscular activity corresponding to elevations and depressions of the larynx during voice for speech, respectively [113]. The first electrode for the submental muscle site was placed approximately 1 cm from midline in the submandibular area superior to the hyoid bone [114–117]. The second electrode of the submental pair was placed in line with the fibers of the muscle and with an interelectrode distance of approximately 1.5 cm [101, 102, 112, 118]. The submental location captures muscle activity from the anterior belly of the digastric, mylohyoid, and geniohyoid muscles.

For the infrahyoid muscle site, the first electrode was centered over the thyroid cartilage just below the thyroid notch and approximately 1 cm off midline [101, 102, 114, 115, 119]. The infrahyoid location captures muscle activity from the sternohyoid and omohyoid muscles with additional activity captured from the thin muscle sheath called platysma overlying most of the neck [102, 116]. Due to the small sizes of the individual muscles making up the submental and infrahyoid musculature as well as the multilayered structure of the muscles, sEMG can only capture muscle group activity and not activity from individual muscles. Moreover, it is not realistic to record activity from deeper muscles such as the thyrohyoid and cricothyroid [102]. The ground electrode was placed on the superior bony prominence of the shoulder (acromion). For voice and

speech sEMG recordings, a placement of the ground electrode close to the electrodes is preferred [102]. A net bandage was placed over the electrodes in order to help keep the cables from moving around during data collection as seen in Figure 4.8.

The quality of electrode placement was confirmed with tasks that produce target activations such as a swallow (submental and infrahyoid activity) and production of a front vowel (/i/, submental) and back vowel (/u/, infrahyoid).

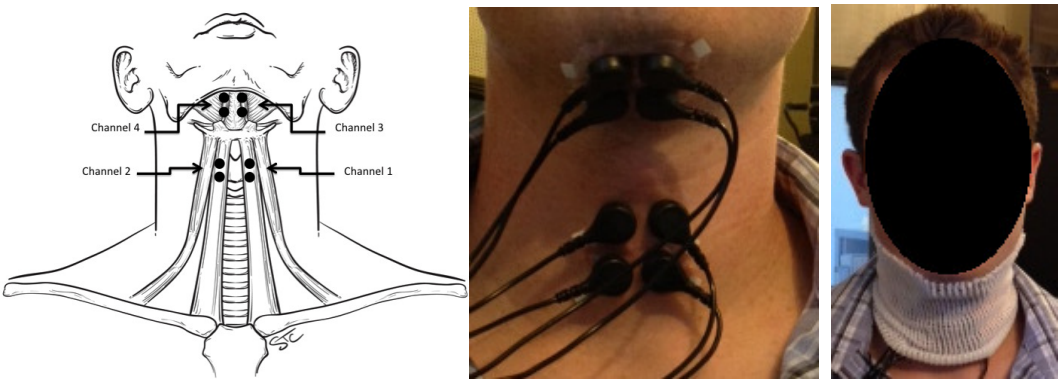



Figure 4.8: Muscle groups on the human neck: diagram, actual view of electrode placement, and actual view with bandage applied.

#### 4.3.2.3 Pressed Vocal Gesture Training

Since the participants were all vocally healthy, prior to data collection, a training program was implemented by a certified speech-language pathologist with experience in voice disorders to simulate "pressed" voice productions to be completed by each subject. The training consisted of verbal description and demonstrations of pressed voice based on the protocol by [120]: i.e. "an extremely high-effort phonation mode, with the perception of an almost completely closed airway, as if pushing". Next, subjects listened to selected audio samples of sustained /a/ productions by males and females with severe vocal hyperfunction chosen from the KayPentax Disordered Voice Database. Finally, the participants practiced normal and pressed productions of /a/, /u/, and /i/ before proceeding to vowel productions with concurrent air-flow feedback using the KayPentax Phonatory Aerodynamic System. Target levels for air flow were primarily

based on normative data for the PAS "Comfortable Sustained Phonation Protocol" (sustained /a/) and consisted of a mean of 0.13 L/s (SD = 0.08) (for both male and female, ages 18-39 years) and a mean of 0.11 L/s (SD = 0.05) (for males, 40-59 years) for normal /a/ production [121]. Reported normative data for average air-flow rates for the vowel /i/ are 0.14 L/s for males and 0.18 L/s for females [122]. Normative data for /u/ are not readily available, but they are expected to fall within a similar range. Each subject was able to produce normal and pressed vowel productions inside and outside the norm range, respectively, during training, and contrasts were perceptually distinct. Five repetitions of each vowel gesture were recorded with concurrent air-flow visual feedback data prior to full sEMG data collection. Full sEMG data were collected without concurrent air-flow feedback to avoid additional muscular neck activity from holding the PAS face mask against the face. During data collection, all participants were perceptually monitored for contrasts between normal and pressed productions. Participants were encouraged and received feedback to maintain pressed phonations as necessary.

### 4.3.3 Results

For all of the results presented here, a 10-fold cross validation was performed. Each time 90% of the signals from all collected gestures were used for training and the remaining 10% were used for classification.

#### 4.3.3.1 Distinct Gestures

In the first test, a set of six distinct gestures containing speech and non-speech behaviors (e.g. vowel, consonant, and throat sounds) was used. These are the same gestures used in [123], that is, */u/, /i/, /t/, /s/, cough, throat clear*. However, the results here expand upon those tests by using data from ten subjects – four female and six males – as opposed to a single subject. Classification was completed using the improved HiGUSSS algorithm described in Section 3.3. The overall average in classification accuracy – i.e. over all gestures and over all subjects – was approximately 85%. Figure 4.9 shows the classification accuracies per gesture, averaged over all ten subjects, while Figure 4.10

shows the classification accuracies per subject, averaged over all six gestures. Both the means and the corresponding standard deviations are depicted in these same figures. Finally, Table 4.8 presents the average confusion matrix computed over all ten subjects.

The high accuracy achieved in this test should positively address part of our first research question: whether sEMG devices can reliably associate a larger number of patterns of extralaryngeal muscle activity with voice tasks underlying speech and non-speech behaviors.

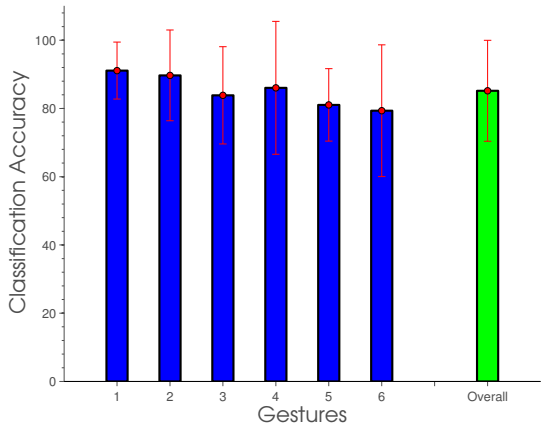



Figure 4.9: Means and standard deviations of the classification accuracies per gesture, over all ten subjects. Six gestures considered: */u/*, */i/*, */t/*, */s/*, *cough*, and *throat clear*.

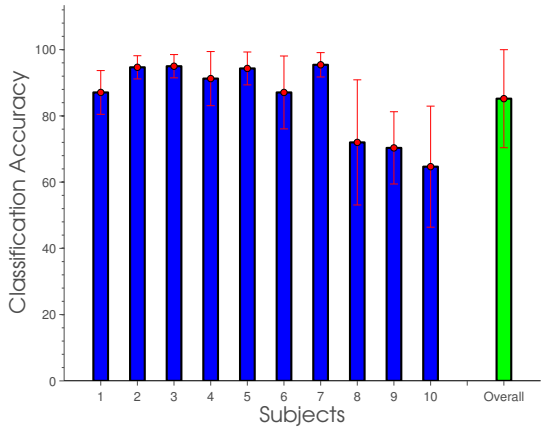



Figure 4.10: Means and standard deviations of the classification accuracies per subject, over six gestures. Six gestures considered: */u/*, */i/*, */t/*, */s/*, *cough*, and *throat clear*.

Table 4.8: Confusion matrix for six gestures averaged over all ten subjects.

| | | Classified | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1) */u/* | 2) */i/* | 3) */t/* | 4) */s/* | 5) *Cough* | 6) *T. clear* |
| Actual | 1) */u/* | 91.10 | 6.30 | 0.25 | 0.70 | 0.70 | 0.95 |
| | 2) */i/* | 7.70 | 89.70 | 0.20 | 0.20 | 0.80 | 1.40 |
| | 3) */t/* | 1.60 | 0.20 | 83.85 | 8.90 | 3.80 | 1.65 |
| | 4) */s/* | 0.25 | 0.40 | 8.05 | 86.05 | 2.65 | 2.60 |
| | 5) *Cough* | 0.70 | 0.60 | 3.95 | 1.45 | 81.05 | 12.25 |
| | 6) *T. Clear* | 0.25 | 0.60 | 1.25 | 2.65 | 15.90 | 79.35 |

#### 4.3.3.2 Large Gesture Set

In order to further address our first research question, we selected a large number of gestures (ten). The selected gestures were, */a/, /a/ pressed, /u/, /u/ pressed, /i/, /i/ pressed, /t/, /s/, cough, throat clear*. As before, classification was completed using the HiGUSSS algorithm. An overall average classification accuracy of 74% was achieved – i.e. averaged over all gestures and over all subjects. As it can be observed in the next figures, it is important to notice that one of the subjects (#6) presented a much lower average, bringing down the overall average to 74%. In spite of that, these results still shows that the system is robust to large gesture sets (research question one), but also to sets containing similar gestures (research question two). Figures 4.11 and 4.12 show, respectively, the classification accuracies per gesture, over all ten subjects, and per subject, over all ten gestures. Table 4.9 shows the average confusion matrix.

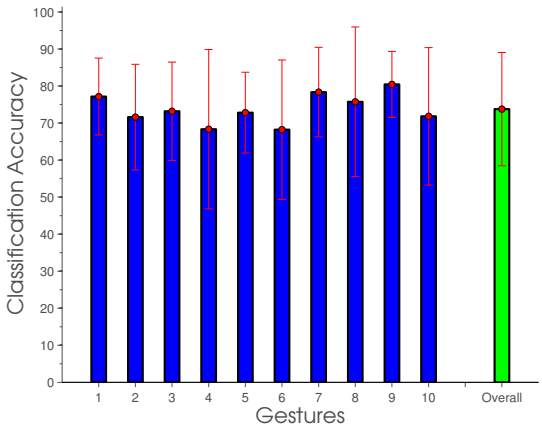



Figure 4.11: Means and standard deviations of the classification accuracies per gesture, over all ten subjects. Ten gestures considered: */a/*, */a/ pressed*, */u/*, */u/ pressed*, */i/*, */i/ pressed*, */t/*, */s/*, *cough*, and *throat clear*.

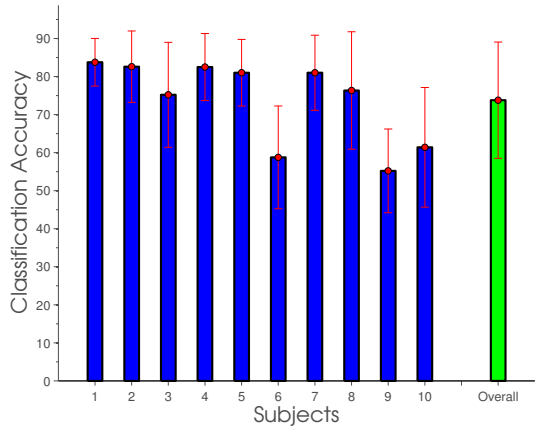



Figure 4.12: Means and standard deviations of the classification accuracies per subject, over all ten gestures. Ten gestures considered: */a/*, */a/ pressed*, */u/*, */u/ pressed*, */i/*, */i/ pressed*, */t/*, */s/*, *cough*, and *throat clear*.

#### 4.3.3.3 Normal vs. Pressed Gestures

In order to show that not only can the system achieve high accuracy with similar gestures, but it can also distinguish specific gestures within the normal and pressed

Table 4.9: Confusion matrix for ten gestures averaged over all ten subjects.

| | | Classified | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1) /a/ | 2) /a/ *pressed* | 3) /u/ | 4) /u/ *pressed* | 5) /i/ | 6) /i/ *pressed* | 7) /t/ | 8) /s/ | 9) *Cough* | 10) *T. clear* |
| Actual | 1) /a/ | 77.17 | 7.90 | 3.08 | 2.20 | 1.90 | 4.70 | 0.40 | 0.25 | 0.75 | 1.65 |
| | 2) /a/ *pressed* | 5.78 | 71.62 | 0.58 | 5.55 | 1.98 | 10.08 | 0.25 | 1.55 | 2.00 | 0.60 |
| | 3) /u/ | 4.10 | 2.65 | 73.20 | 8.27 | 6.45 | 4.88 | 0.00 | 0.00 | 0.45 | 0.00 |
| | 4) /u/ *pressed* | 1.45 | 6.65 | 4.70 | 68.35 | 2.30 | 13.25 | 0.20 | 0.25 | 1.85 | 1.00 |
| | 5) /i/ | 4.20 | 4.95 | 6.25 | 5.55 | 72.82 | 3.68 | 0.40 | 0.20 | 1.05 | 0.90 |
| | 6) /i/ *pressed* | 2.00 | 10.10 | 2.28 | 14.05 | 1.35 | 68.22 | 0.20 | 0.25 | 1.10 | 0.45 |
| | 7) /t/ | 0.00 | 3.30 | 1.00 | 1.65 | 0.40 | 1.32 | 78.37 | 5.77 | 6.00 | 2.20 |
| | 8) /s/ | 0.00 | 6.10 | 0.50 | 3.45 | 0.00 | 0.83 | 7.77 | 75.75 | 3.00 | 2.60 |
| | 9) *Cough* | 0.65 | 2.80 | 0.50 | 1.50 | 0.00 | 0.33 | 2.40 | 0.53 | 80.43 | 10.85 |
| | 10) *T. clear* | 1.60 | 3.25 | 1.00 | 2.90 | 0.20 | 0.45 | 0.85 | 1.40 | 16.53 | 71.82 |

classes, tests were completed using three normal and three pressed gestures. This partially addresses our second research question or whether the system can differentiate between multiple vowel sounds produced in a normal manner compared with a pressed (low air flow) manner.

The gestures used here were */a/, /a/ pressed, /u/, /u/ pressed, /i/,* and */i/ pressed*, and they were collected as described in Section 4.3.2.3. In this case, an overall average classification accuracy of 78% was achieved. As before, classification accuracies per gesture and per subject are presented before the average confusion matrix: Figure 4.13, Figure 4.14, and Table 4.10, respectively.

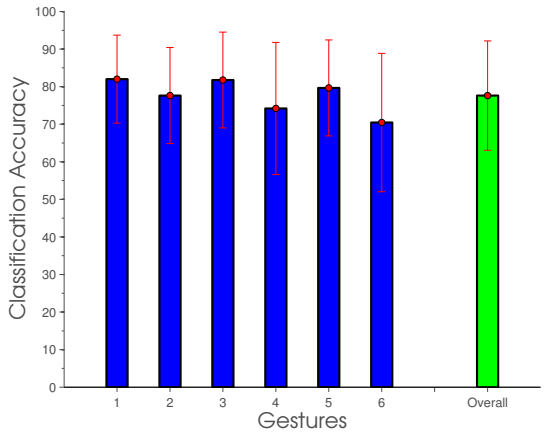



Figure 4.13: Means and standard deviations of the classification accuracies per gesture, over all subjects. Six gestures considered: */a/, /a/ pressed, /u/, /u/ pressed, /i/,* and */i/ pressed.*

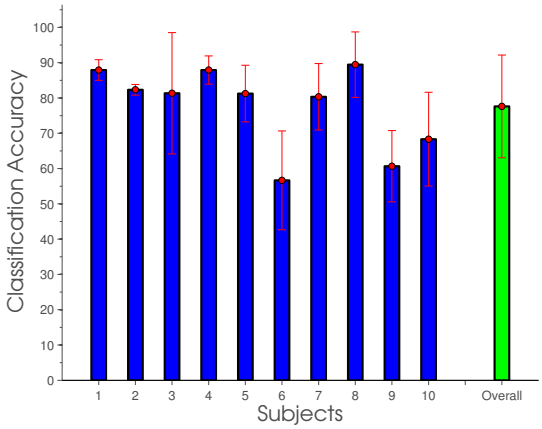



Figure 4.14: Means and standard deviations of the classification accuracies per subject, over six gestures. Six gestures considered: */a/*, */a/ pressed*, */u/*, */u/ pressed*, */i/*, and */i/ pressed*.

Table 4.10: Confusion matrix for the second set of six gestures averaged over all ten subjects.

| | | Classified | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1) */a/* | 2) */a/ pressed* | 3) */u/* | 4) */u/ pressed* | 5) */i/* | 6) */i/ pressed* |
| Actual | 1) */a/* | 82.00 | 6.38 | 3.33 | 2.45 | 2.65 | 3.18 |
| | 2) */a/ pressed* | 6.15 | 77.63 | 1.30 | 5.40 | 2.00 | 7.52 |
| | 3) */u/* | 1.83 | 1.70 | 81.77 | 6.33 | 6.30 | 2.07 |
| | 4) */u/ pressed* | 0.65 | 4.75 | 4.90 | 74.20 | 3.65 | 11.85 |
| | 5) */i/* | 3.30 | 2.90 | 6.70 | 2.65 | 79.67 | 4.78 |
| | 6) */i/ pressed* | 2.10 | 8.65 | 1.00 | 14.20 | 3.60 | 70.45 |

#### 4.3.3.4 Intra- and Inter-Subject Testing

The remainder of research question two was addressed by completing intra- and inter-subject testing. Once again, together with the previous test, the results of this test can justify our method as a potential solution to the detection of normal and maladaptive extralaryngeal patterns associated with voice problems.

### Intra-Subject

Intra-subject testing was carried out for each of the ten subjects separately using three normal and three pressed gestures, that is, */a/, /a/ pressed, /u/, /u/ pressed, /i/,* and */i/ pressed*. These gestures were divided into two sub-classes: the gestures */a/, /u/, and /i/* were treated as the *Normal* class, and gestures */a/ pressed, /u/ pressed, and /i/ pressed* were treated as the *Pressed* class. The classification was then completed as a two class problem using HiGUSSS. An overall average in classification accuracy of 95% was achieved. This clearly supports the potential use of an sEMG device for detecting vocal dysfunctions. Figure 4.15 shows the average classification accuracy over all subjects for each of the two sub-classes (Normal vs. Pressed), and Figure 4.16 shows the results per user. Table 4.11 shows the average confusion matrix averaged over all ten subjects.

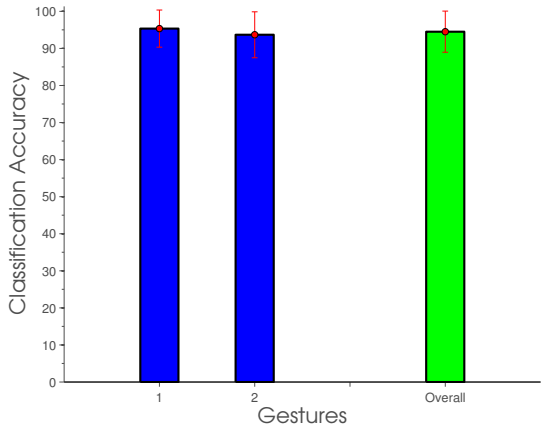



Figure 4.15: Means and standard deviations of the classification accuracies over all subjects using normal vs. pressed gestures – i.e. simulated dysfunction.

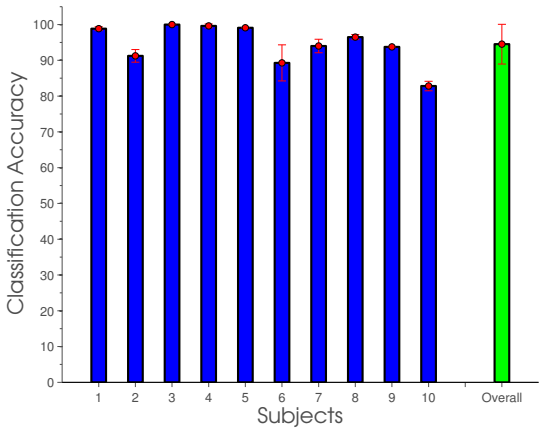



Figure 4.16: Means and standard deviations of the classification accuracies per subject using normal vs. pressed gestures – i.e. simulated dysfunction.

Table 4.11: Confusion matrix for intra-subject accuracy in detection of simulated dysfunction averaged over all ten subjects.

| | | Classified | |
|---|---|---|---|
| | | 1) Normal | 2) Pressed |
| Actual | 1) Normal | 95.35 | 4.65 |
| | 2) Pressed | 6.32 | 93.68 |

### Inter-Subject

Inter-subject tests were also completed for all ten subjects combined and using the same three normal and three pressed gestures. These gestures were once again grouped into two sub-classes, *Normal* and *Pressed*. However, this time, the test was performed in a *leave-one-out* fashion. That is, for each subject, the training was completed using data from all the other subjects – i.e. all data except for the subject being tested. Classification was then completed as a two class problem using the data from that subject.

As before, Figures 4.17 and 4.18, and Table 4.12 show the results of this test. As the reader will notice, the HiGUSSS performed very poorly for the inter-subject case. So, to further investigate the reason for such low performance, we run the same test set

using a MC-SVM. The figures and tables above also include the result for the MC-SVM, which performed even worse than the HiGUSSS. We attribute this poor performance by both classifiers to data overfitting: i.e. obviously, both classifiers can learn very well each subject's patterns, but they fail to generalize across subjects. This conclusion is supported by the excellent result in the intra-subject test on one hand, and the poor result in inter-subject test on the other hand. The actual reason for this overfitting needs to be further investigated, but adding more diversity to the data by adding more subjects or more gestures could alleviate this problem.

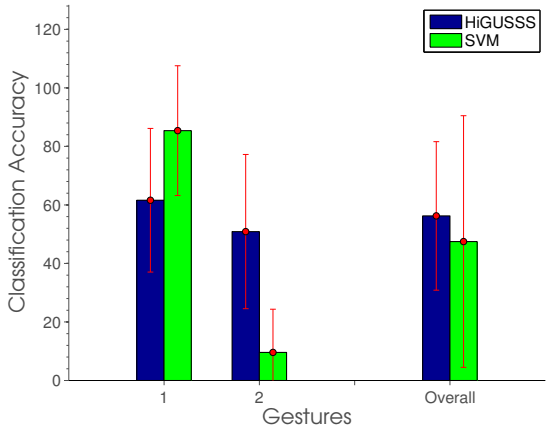



Figure 4.17: Means and standard deviations of the classification accuracies per gesture, for normal vs. pressed gestures – using the Leave-one-out approach for the HiGUSSS and the SVM Classifier

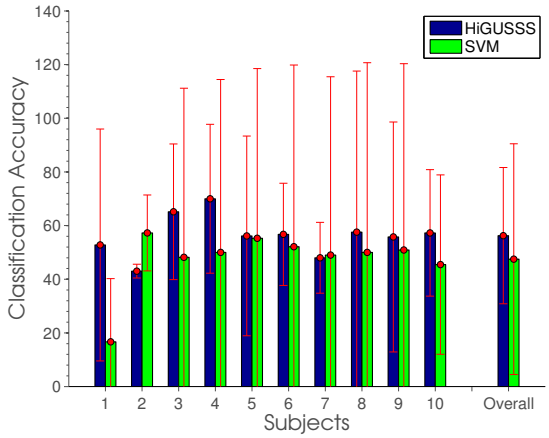



Figure 4.18: Means and standard deviations of the classification accuracies per subject, for normal vs. pressed gestures – using the Leave-one-out approach for the HiGUSSS and the SVM Classifier

Table 4.12: Confusion matrix for inter-subject accuracy in detection of simulated dysfunction averaged over all ten subjects for the HiGUSSS and the SVM classifier.

| HiGUSSS | | Classified | |
|---|---|---|---|
| | | 1) Normal | 2) Pressed |
| Actual | 1) Normal | 61.60 | 38.40 |
| | 2) Pressed | 49.12 | 50.88 |

| SVM | | Classified | |
|---|---|---|---|
| | | 1) Normal | 2) Pressed |
| Actual | 1) Normal | 85.40 | 14.60 |
| | 2) Pressed | 90.42 | 9.58 |

#### 4.3.3.5 Classifier Comparison

Finally, a comparison between the HiGUSSS algorithm and two other classifiers was also completed. The goal was to illustrate the value of the HiGUSSS system as opposed to other more traditional classifiers. For comparison purposes, tests were run on all three groups of gestures: i.e. the six distinct gestures used in Section 4.3.3.1; the large gesture set used in Section 4.3.3.2; and the three pressed and three normal gestures used in Section 4.3.3.3. These three groups of gestures were classified using both a simple

distance classifier and a single layer Multi-Class SVM, which were then compared to the results for the HiGUSSS presented earlier. Figures 4.19, 4.20 and 4.21 show the corresponding classification results per subject, and overall. Note that the HiGUSSS classifier outperforms the distance and SVM classifiers in almost all the cases, and most notably for the cases with large number of gestures.

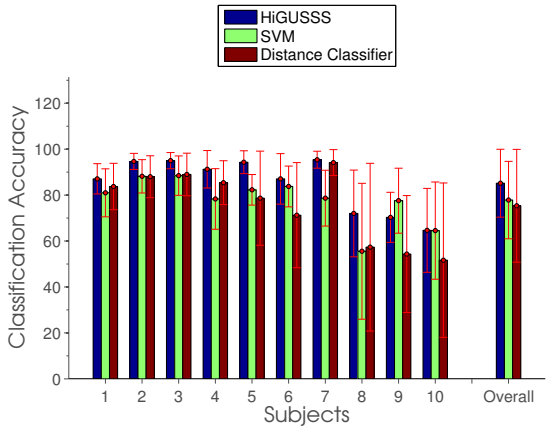



Figure 4.19: Comparison between the average classification accuracies per subject, over six gestures, for each of the three classifier (HiGUSSS, MC-SVM, and Distance). First set of six gestures considered: */u/*, */i/*, */t/*, */s/*, *cough,* and *throat clear.*

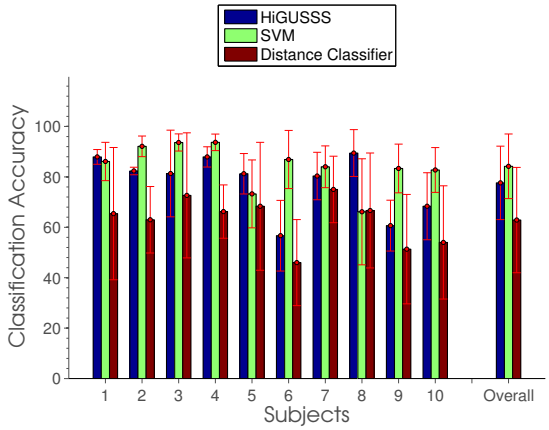



Figure 4.20: Comparison between the average classification accuracies per subject, over six gestures, for each of the three classifier (HiGUSSS, MC-SVM, and Distance). Second set of six gestures considered: */a/*, */a/ pressed*, */u/*, */u/ pressed*, */i/*, and */i/ pressed.*

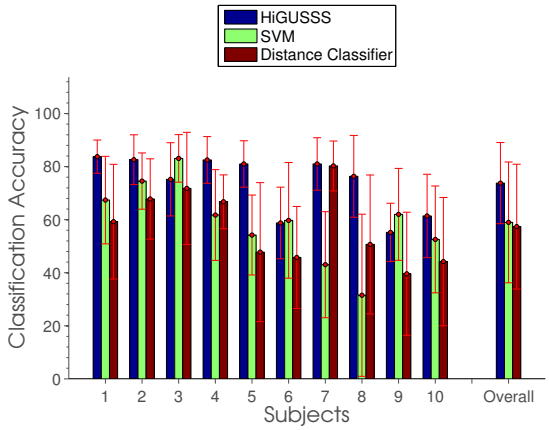



Figure 4.21: Comparison between the average classification accuracies per subject, over all ten gestures, for each of the three classifier (HiGUSSS, MC-SVM, and Distance). Set of all ten gestures considered: */a/, /a/ pressed, /u/, /u/ pressed, /i/, /i/ pressed, /t/, /s/, cough,* and *throat clear.*

### 4.3.4 Conclusions

The results presented for the distinct gestures further reinforce conclusions drawn from previous work [111], and address our first research question: that meaningful classification can be drawn from sEMG signals collected at the anterior neck. In [111], data were collected and tested for only a single subject, while in this work data were collected and tested for ten subjects (four females and six males). As seen in Section 4.3.3.1, average classification accuracy for ten subjects performing six gestures (*/u/, /i/, /t/, /s/, cough, throat clear*) was 85%, which is consistent with the 90% classification accuracy achieved in [111] for one single subject.

The classification accuracy achieved using the HiGUSSS system was compared with the classification accuracies from a single-layer MC-SVM and a simple distance classifier in Section 4.3.3.5. Two meaningful trends were discovered during these tests. First, it can be noticed that as the number of gestures increased, the advantage of the HiGUSSS method over the other two classifiers became clearer. Second, as the gestures in the set became more similar, once again the HiGUSSS system outperformed the other two methods. This shows the validity of the HiGUSSS system and the stronger case for its

application in the detection of vocal dysfunctions for similar gestures and large vocal gesture sets.

# Chapter 5

# Formal Validation of the Method

As shown in the previous chapter, the GUSSS method for pattern recognition has been successfully applied in various areas. However, there are some important aspects of the method that needed to be investigated. In this chapter, we pose a series of research questions to formally validate the proposed method. We also describe the corresponding experiments that were designed to further investigate aspects such as: the effect of different parameters, effects of the ICA algorithm used, independence and other assumptions, the learning of the signatures, and comparison with other methods. The tests were performed using both synthetic and real data, the latter collected over the past few years or taken from public databases.

## 5.1 Research Questions

The frameworks that have been developed and tested rely on the GUSSS method described in sec. 3.1. Several research questions have been raised, related to fundamental aspects about the method.

1. Is the method sensitive to the choice of parameters and ICA algorithms? Does the method behave as expected? Is it affected by noise?

2. Is it valid to assume independence of a set of signals?

3. Is it possible to learn the signatures?

4. How does the method compare to other approaches?

5. Other considerations.

In order to answer these questions and further validate the proposed method, we ran extensive experiments, which are described in the following sections. In this chapter we present representative and illustrative results. The complete set of results are presented in the Appendices.

## 5.2 Synthetic Dataset

For many of the experiments below, we considered 8 different sound sources or signatures, taken from the ICA Research Center at Helsinki University of Technology [124]. Let $s_1$, $s_2$, ..., $s_8$ be those signatures, which are shown in Figure 5.1. Assume there is only $M = 1$ microphone (sensor) that can capture a sound signal. This signal is a combination or mixture of the 8 source signatures, and it can be modeled as:

$$x = c_1 s_1 + c_2 s_2 + \cdots + c_8 s_8 \tag{5.1}$$

The mixing coefficients $c_i$ represent the intensities that each individual signature $s_i$ has in the mixed signal, due to the different distances between the sources and the microphone. Note that if a particular signature, say $s_p$, contributed to the mixture $x$, the corresponding coefficient $c_p$ is non-zero. If source $p$ could not be detected when the mixture was obtained, then $c_p = 0$.

As we have explained in the previous chapters, given a sensed mixture $x$, the GUSSS method can be used to determine if signature $s_p$ was present or not within $x$ by estimating $c_p$ and thus the GUSSS ratio $r_p$.

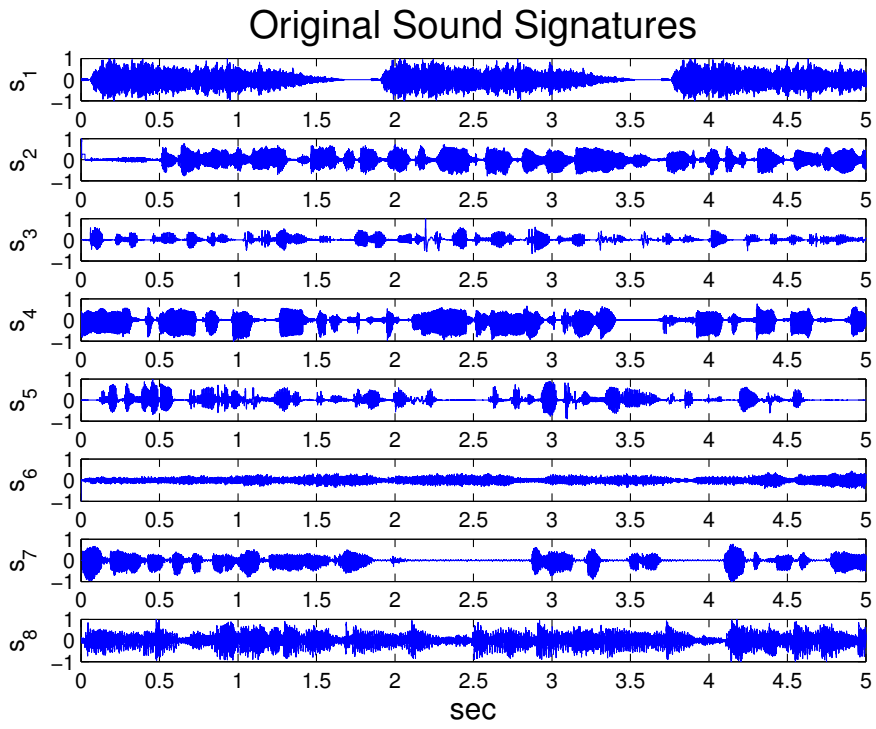



Figure 5.1: Eight sound signatures used for various tests. They correspond to: a police car's siren ($s_1$); a man speaking English ($s_2$); an old man speaking a foreign language ($s_3$); a woman speaking English ($s_4$); another man speaking a foreign language ($s_5$); an opera singer ($s_6$); a woman speaking a foreign language ($s_7$); a band playing pop music ($s_8$).

We have run many experiments using several different combinations of the signatures described above, as will be explained in the following sections.

## 5.3 Sensitivity of the Method

The choice of parameters, the initialization of algorithms, or even the use of so-called "magic numbers" are fundamental factors in many methods found in the literature, across many different areas. A poor choice may result in sup-par performances or even failure of the algorithms. On occasions, the choices become *ad-hoc* for the specific applications. The results presented in chapter 4 demonstrate the effectiveness of the GUSSS method in various applications. All those results were obtained using the same

parameters, which include weight values and the choice of ICA algorithm. We are interested in determining if the method's success or applicability depends upon the choice of those parameters.

We are also interested in determining if our method is well-behaved. That is, for controlled conditions such as the amount of target signature present in a mixture, we expect output values – i.e. GUSSS ratios – to be consistent with those conditions. Finally, we want to examine how the method is affected by noise. Next we describe the experiments designed to address research question 1, and we present the corresponding results.

### 5.3.1 Effects of Weights

In the GUSSS algorithm, a second synthesized signal $x_p$ is created by injecting a weighted copy of the sought-after signature $s_p$ into the sensed signal $x$. Recall eq. 3.2, $x_p = w_1 x + w_p s_p$. In all the preliminary results, both weights, $w_1$ and $w_p$, have been arbitrarily set to 1.

To test how the GUSSS ratios are affected when varying the weight values, we chose different combinations of $w_1$ and $w_p$ and we calculated the corresponding ratios.

Mixtures containing the sound signatures discussed above were created as follows:

$$x = \sum_{i \neq p} s_i + c_p s_p$$

For class "Present", $c_p$ was set to 1. For class "Not Present", $c_p$ was set to 0. A total of 400 synthesized signals $x_p$ were created using $w_1$ and $w_p$ values taken from the interval [0.1, 10]. We observed that if the weights were smaller than 0.1 (approaching zero), the ICA algorithm would fail. This makes sense, given eq. 3.2. If $w_1 = 0$, then the synthetic signals would not contain the test signal $x$. If $w_p = 0$, the synthetic signal would not contain the sought-after signature. We also tried values greater than 10, but no further insights were gained by doing so. That is why we kept the [0.1, 10] interval.

Figures 5.2 and 5.3 show ratios obtained using signatures $s_1$ and $s_5$, respectively. $p = 1, 5$ were chosen arbitrarily here. The same tests were performed for all signatures.

For illustration purposes, the ratios (as function of $w_p$ and $w_1$) are shown as a $3D$ surface plot, and as a $2D$ gray scale image.

These results were obtained using the FastICA algorithm with its default settings, which we had used in our initial preliminary work. Those settings consist of a randomly chosen initial estimate for the mixing matrix $A_{init}$, and a Gaussian non-linearity [41,42].

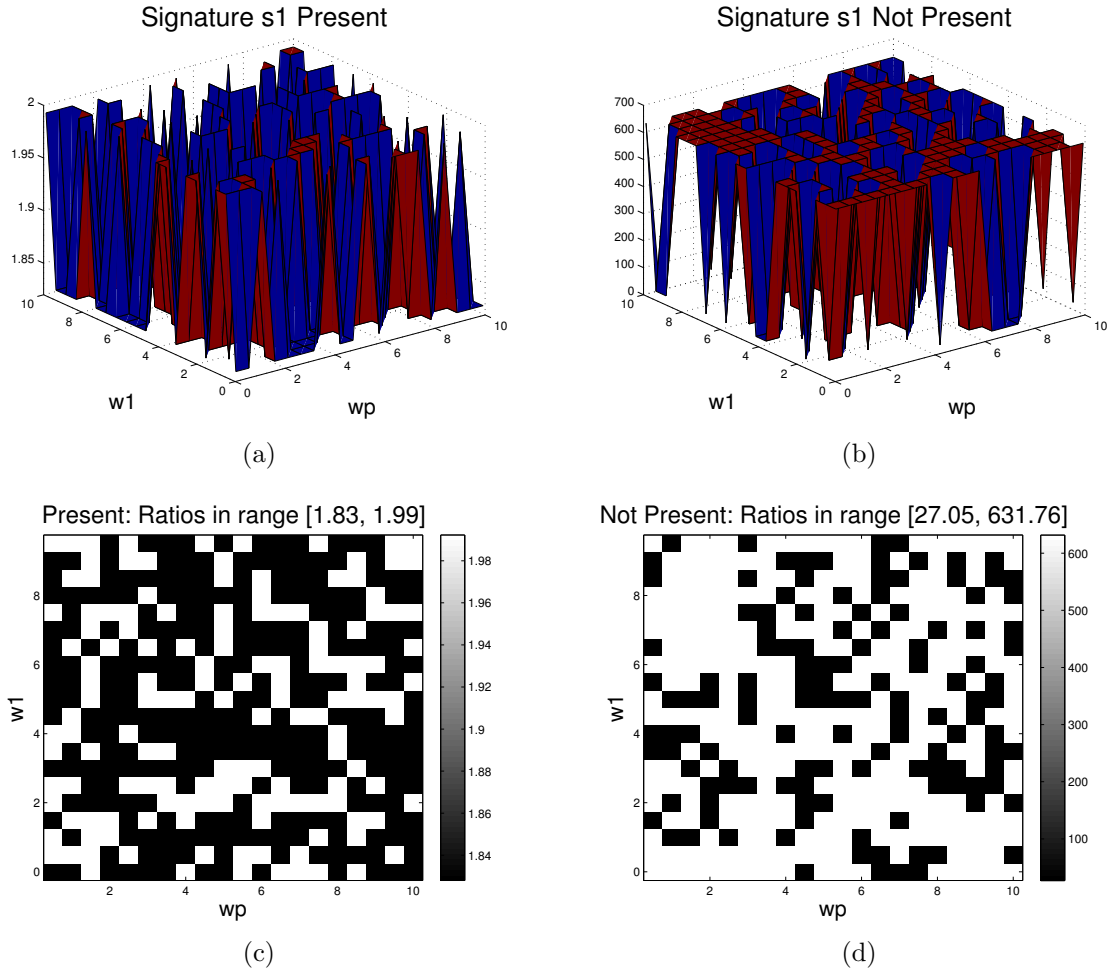



Figure 5.2: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. Sought-after signature $s_1$. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

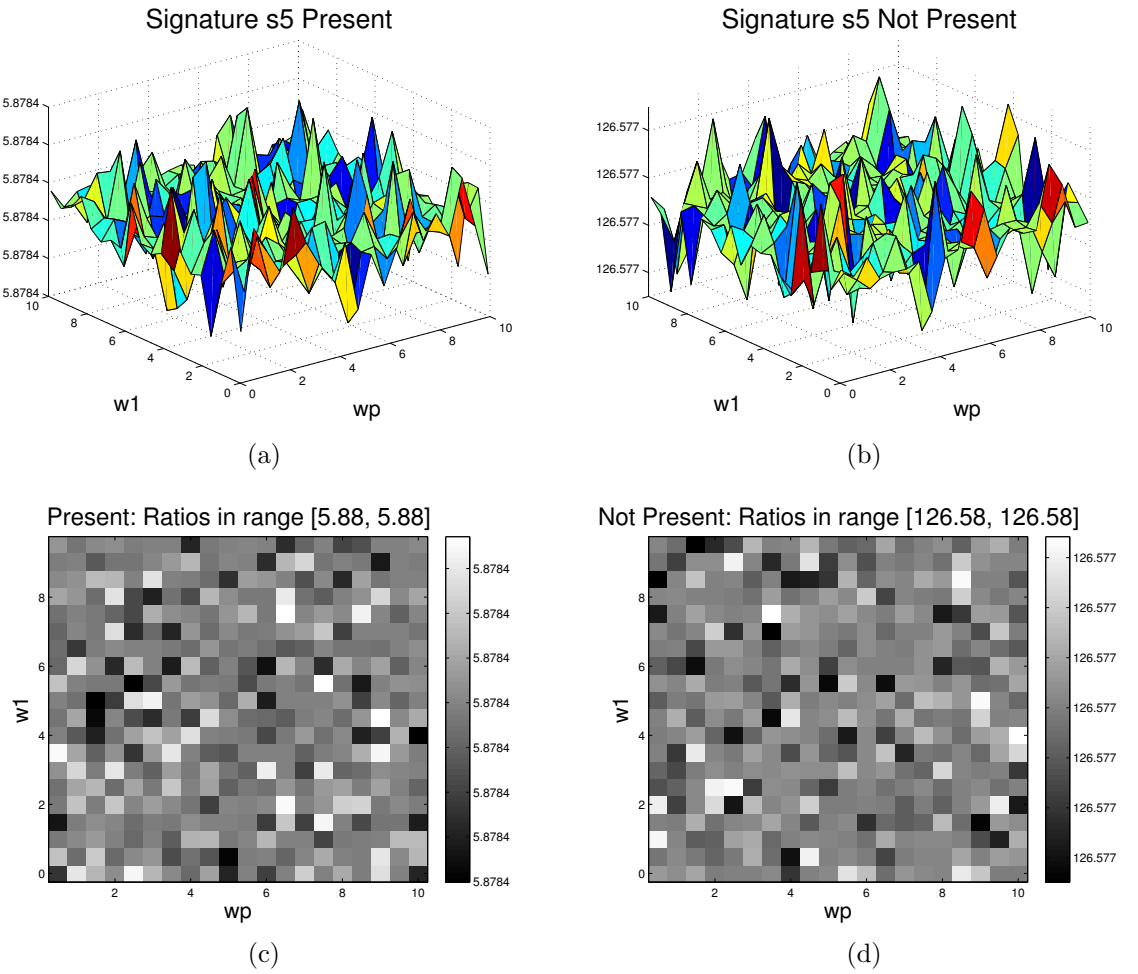



Figure 5.3: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. Sought-after signature $s_5$. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

In the case of $s_1$, note how the ratio values vary considerably, as opposed to the case of $s_5$, where the ratios are practically the same. Nevertheless, classes "Present" and "Not Present" are well separated in both cases. Note that there does not seem to be a significant effect of the weights in the GUSSS ratios. The randomness of the initial estimate $A_{init}$ seems to cause the variations. The ICA algorithm seemingly finds local maxima when determining the independent components.

### 5.3.2 Effects of ICA Initialization

As we have mentioned before, we have used the FastICA algorithm for all our preliminary results. That implementation allows to provide an initialization $A_{init}$ for the mixing matrix $A$ that ICA estimates (sec. 2.5.1). After some initial work using the default setting of a random $A_{init}$, we set the initialization to the identity matrix, i.e. $A_{init} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. This was motivated by the fact that we wouldn't always get the same GUSSS ratios for the same mixtures being analyzed. The variation was attributed to the randomness of $A_{init}$. This is consistent with the results from the previous section. After we set a fixed $A_{init}$, we would get the same ratio for any given mixture we would test. We would always use weights equal to one for the synthesized signal $x_p$, as explained before.

We ran tests similar to those presented in section 5.3.1, only this time we used different initial estimates $A_{init}$. The goal was to determine the effect of those estimates, combined with the weight values.

Figure 5.4 shows ratios obtained using $s_1$ as the sought-after signature, and using $A_{init} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Figure 5.5 shows ratios obtained using $s_1$ as well, but using $A_{init} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. We ran tests with many other initial estimates and for all the signatures. Those results can be found in Appendix A.

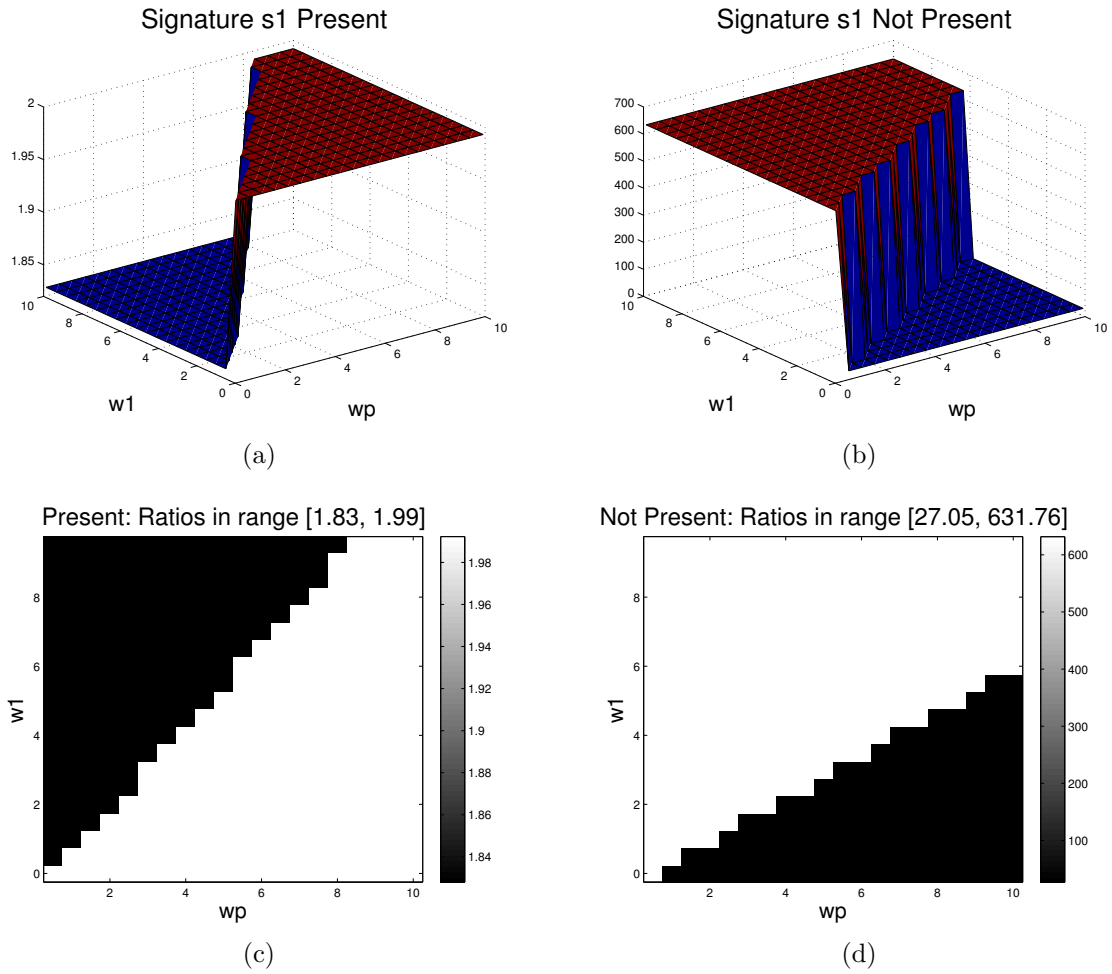



Figure 5.4: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. $A_{init} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Sought-after signature $s_1$. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

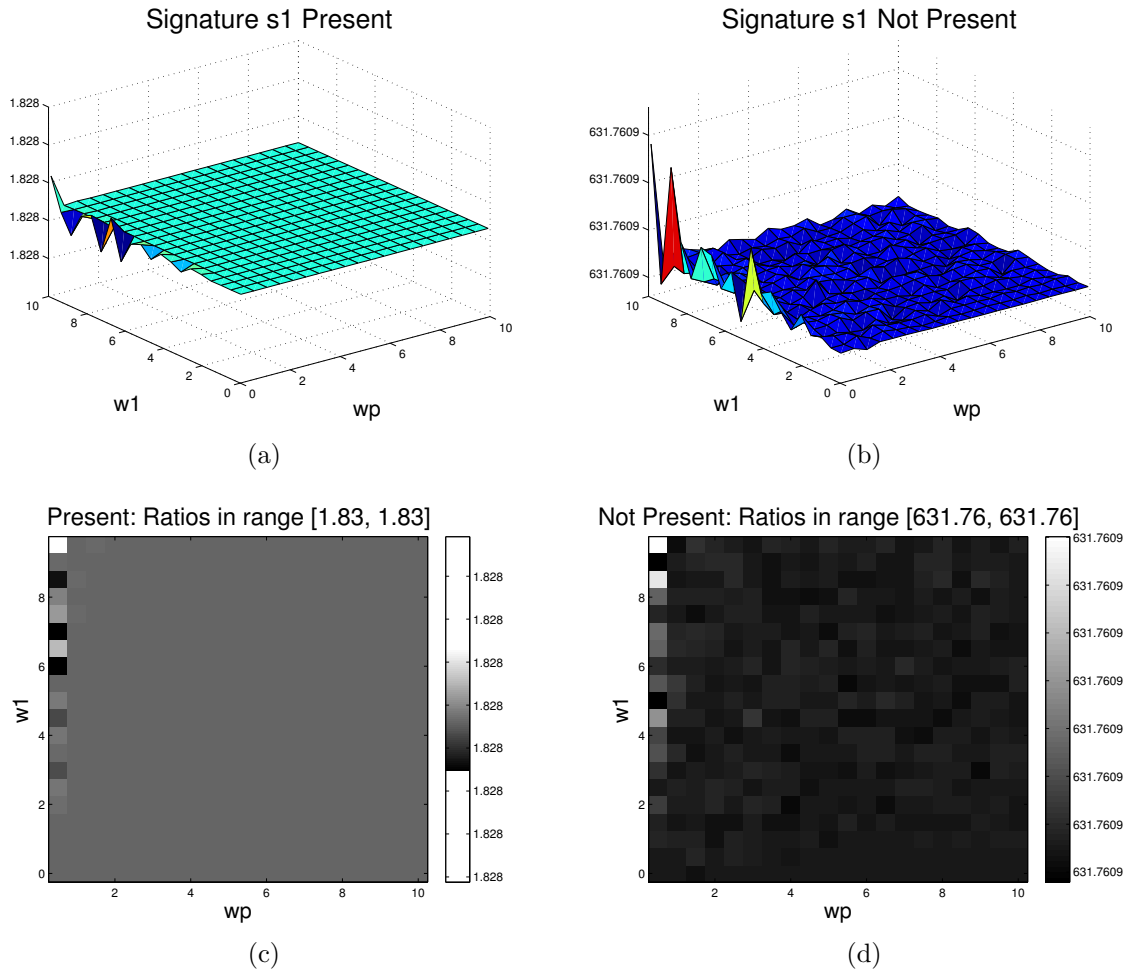



Figure 5.5: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. $A_{init} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Sought-after signature $s_1$. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

For the case $A_{init} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, we observe that the ratio values are in the same ranges as in the case of random initialization (Fig. 5.2). However, here we notice two distinct regions, seemingly separated by a line in the $w_p - w_1$ space. That is, with the given initialization, there are different combinations of weights that lead to the ICA algorithm to reach a different maxima, and thus different mixing matrices $A$.

On the other hand, for the case $A_{init} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, we observe that the results are more consistent. The ratios are practically the same for each class: 1.83 for class

"Present" and 631.76 for the class "Not Present". Note that the results seem to start becoming unstable for $w_p$ close to zero, as expected.

Further cases presented in Appendix A show similar behaviors. Anti-diagonal initial estimates result in more stable results. For other initialization matrices, the choice of weights may have an effect on the ratio values. Either way, the "Present" and "Not Present" cases are clearly distinct.

### 5.3.3 Effects of ICA Algorithm

As we mentioned above, we have been using the FastICA algorithm thus far. We have performed similar experiments than those presented in sections 5.3.1 and 5.3.2, using different non-linearity functions for the FastICA algorithm, as well as using other ICA algorithms described in section 2.5.8.

Figure 5.6 shows ratios obtained using $s_5$ as the sought-after signature. The FastICA algorithm was used, but we set the non-linearity function to be $tanh()$ instead of a Gaussian. We used $A_{init} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Figures 5.7 and 5.8 show results for signature $s_1$ using JADE and a Kernel based algorithm (KDICA), respectively.

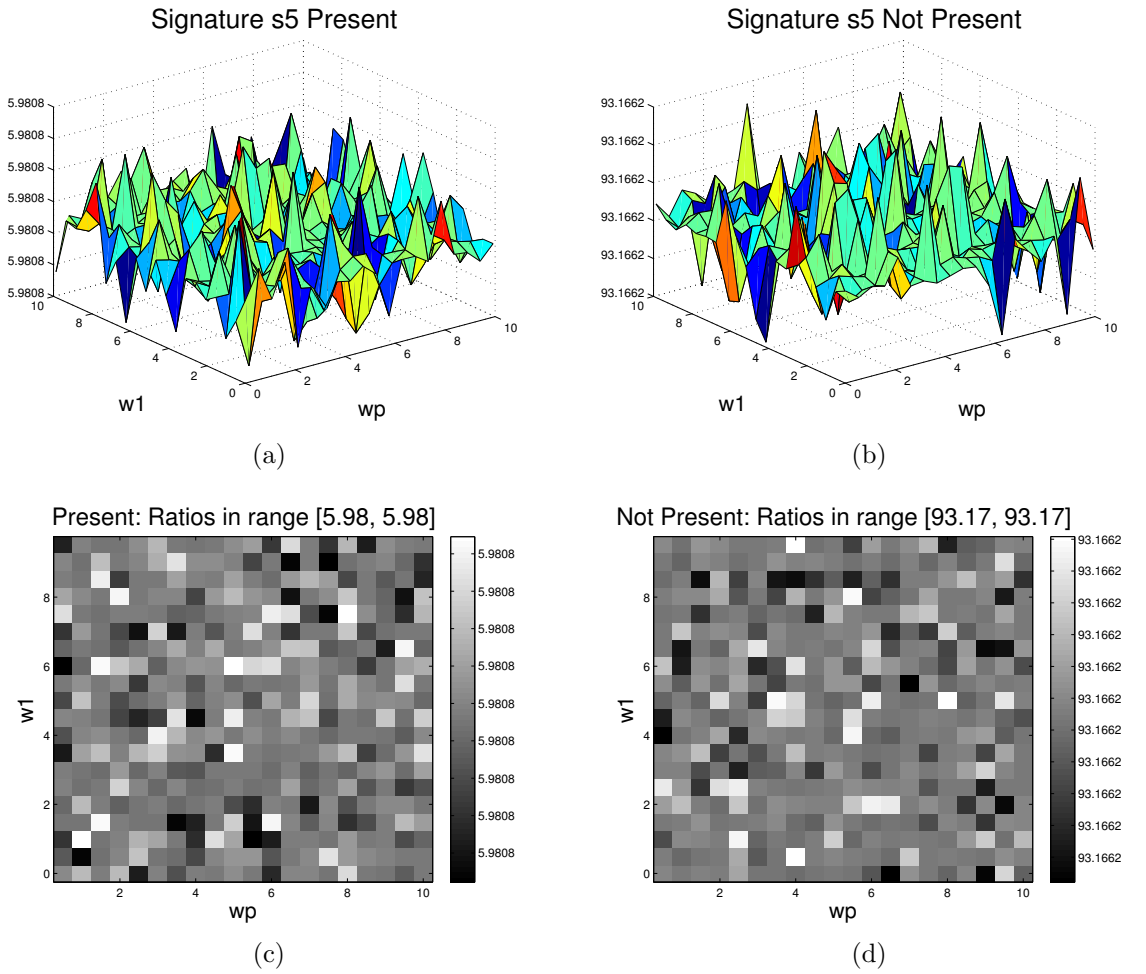



Figure 5.6: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. $A_{init} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ Sought-after signature $s_5$. FastICA algorithm with $tanh()$ nonlinearity function. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

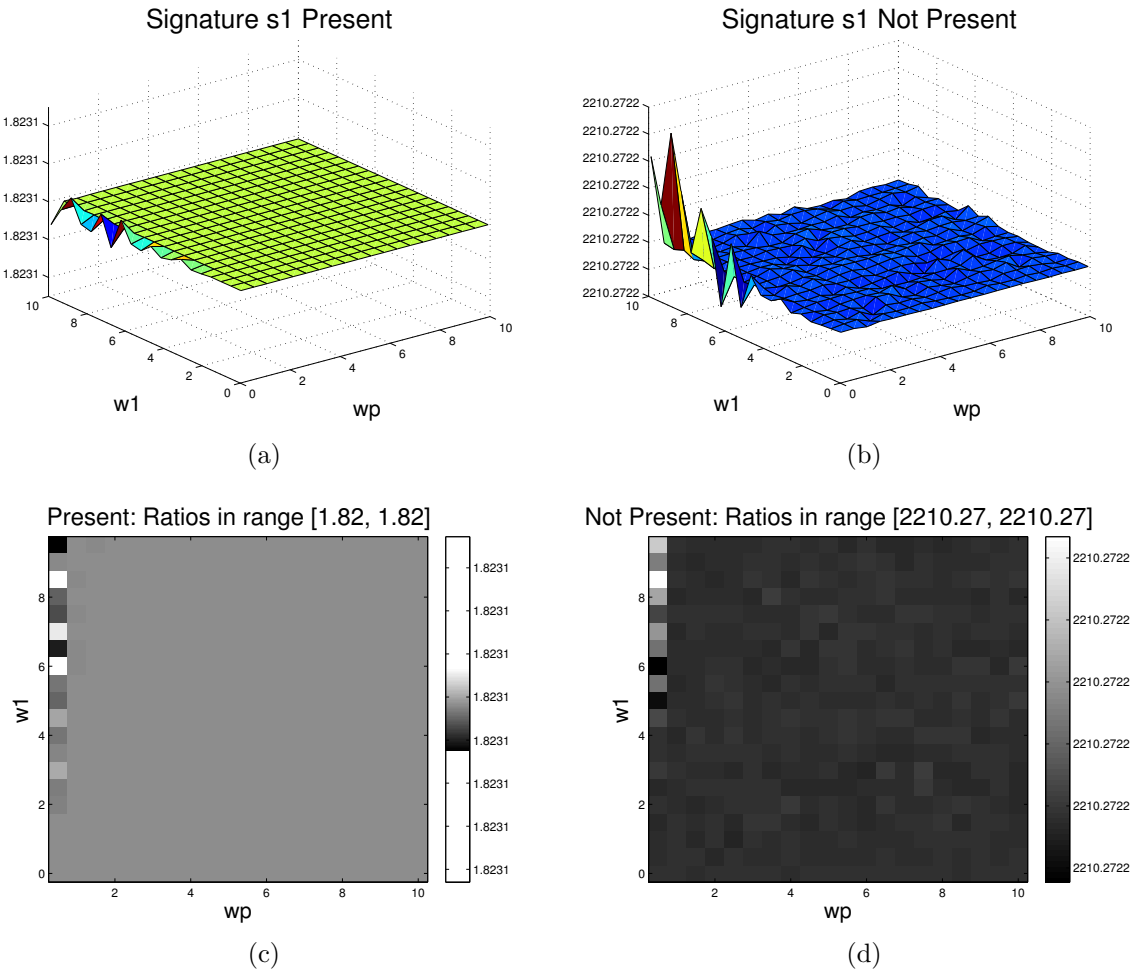



Figure 5.7: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. Sought-after signature $s_1$. JADE algorithm. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

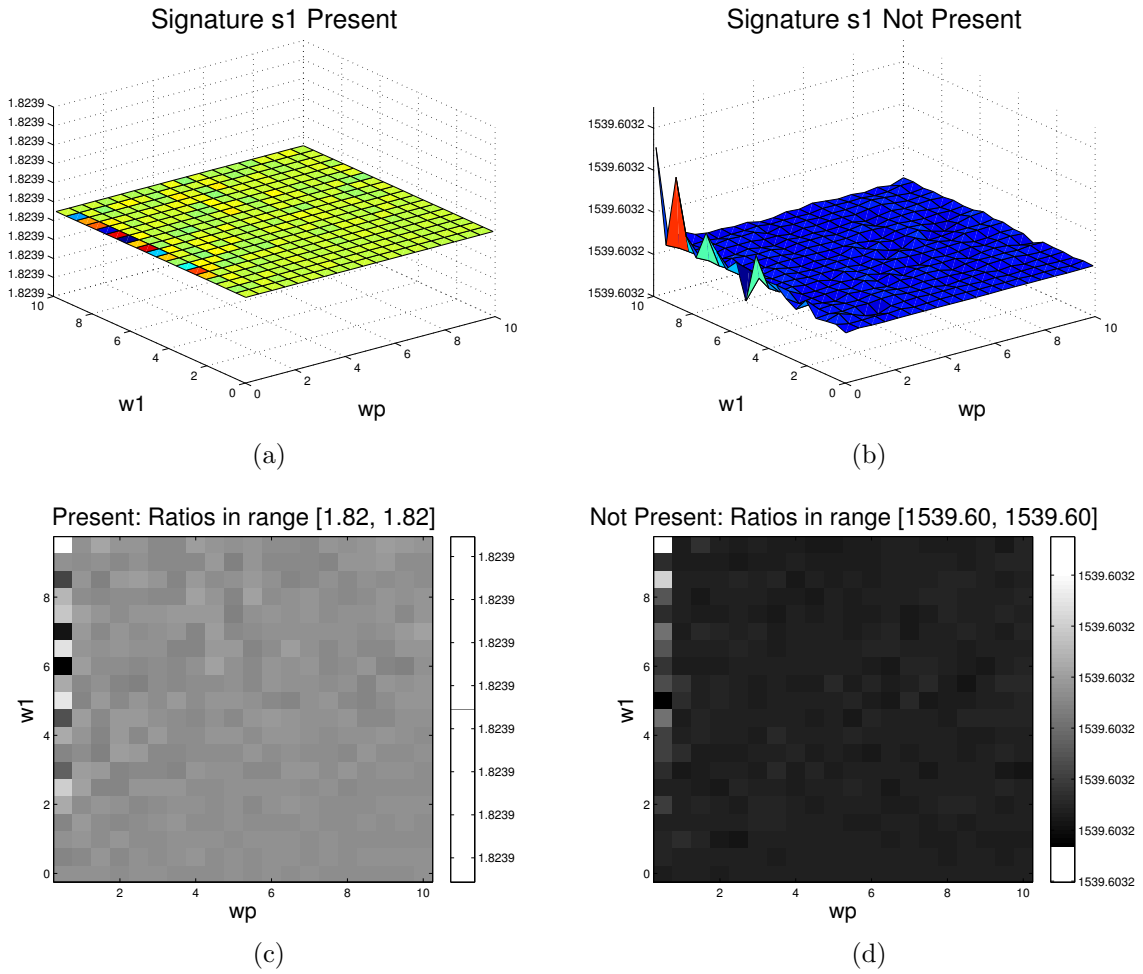



Figure 5.8: GUSSS ratios obtained for different combinations of weights $w_1$ and $w_p$. Sought-after signature $s_1$. KDICA algorithm. a) and c): signature present in mixtures; b) and d): signature not present in mixtures.

Comparing Figures 5.3 and 5.6, we observe that there is practically no difference in the ratio values for class "Present" – 5.88 vs. 5.98. However, there is a difference for class "Not Present" – 126.58 vs. 93.17. The nonlinearity used seems to have some effect when estimating the mixing matrix $A$, and thus, on the GUSSS ratios. However, classes "Present" and "Not Present" are clearly distinguishable in both cases.

In Figures 5.5, 5.7 and 5.8 we observe similarities when using the different ICA algorithms, as well. The ratios for class "Present" are practically the same, for all algorithms, for all weight combinations. The ratios for class "Not Present", on the

other hand, differ from algorithm to algorithm – 631.76 vs. 2210.27 vs. 1539.60 – although there seems to be no effect form the weights in either case. Once again, both classes are clearly separable.

More results using the different signatures and some other ICA algorithms are shown in Appendix A.

### 5.3.4 GUSSS Ratio vs. % of Signature Present in Mixtures

We have stated that in the case when the sought-after signal $s_p$ is not present in the mixture signal $x$, the mixing coefficient $c_p$ should be in theory zero. On the other hand, if $s_p$ is present in the mixture $x$, that coefficient must be different than zero. In practice, the coefficient $c_p$ estimated by the ICA algorithms is never zero, and thus, the GUSSS ratio is not infinite, as seen in the results above. However, it should be very small (close to zero) whenever the particular signature is not present in $x$. Consequently, the GUSSS ratio in those cases is expected to be "large". As the contribution of $s_p$ to the mixture signal grows, that is, as $|c_p|$ increases, the corresponding ratios should show a decreasing tendency. Next we present experiments to show if the expected tendency holds true.

We created mixtures containing the sound signatures discussed above as follows:

$$x = \sum_{i \neq p} s_i + c_p s_p + n$$

where $n$ is a Gaussian noise vector ($\mu = 0$, $\sigma$). We varied $c_p$ from $-0.01$ to 1, creating a different mixture for each $c_p$ value. We calculated the corresponding ratios. We ran the experiments for all signatures, using different noise levels. We also used a few different weight combinations, and we used FastICA with different $A_{init}$ matrices and nonlinearities, as well as other ICA algorithms. Here we show some cases using signature $s_1$. Other cases are shown in Appendix A.

Figure 5.9 shows plots for four combinations of weights: $w_1 = 1$, $w_p = 1$; $w_1 = 1$, $w_p = 10$;$w_1 = 10$, $w_p = 1$; and $w_1 = 10$, $w_p = 10$. For each weight combination,

we calculated ratios using FastICA with 3 different $A_{init}$ matrices: $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ (FICA1); $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ (FICA2); and $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ (FICA3); and we used both Gaussian (G) and $tanh()$ (T) nonlinearities. Ratios corresponding to $c_p \in [-0.01, 0.1]$ are shown in logarithmic scale (to the left), due to the large ratio values observed for $c_p$ close to zero. Ratios corresponding to $c_p \in [0.1, 1]$ are shown in linear scale (to the right). No noise ($\sigma = 0$) is considered for the cases in Figure 5.9. Figures 5.11 and 5.13 also show results using FastICA, for noise levels $\sigma = 0.01$, and $\sigma = 0.1$, respectively.

Figures 5.10, 5.12 and 5.14 show similar plots to those described above, but obtained using other ICA implementations – ITEFICA [125], JADE [40], KDICA [46] and EFICA [48]. Noise levels $\sigma = 0$, $\sigma = 0.01$, and $\sigma = 0.1$ were used, respectively.

In general, the plots show the expected tendency. The GUSSS ratios are highest for $c_p$ values close to zero, and they show a decreasing tendency as $|c_p|$ increases. This is still the case in the noisy cases, although there is clearly an effect of the noise, especially for $c_p$ close to zero. More notably, we observe that some ICA algorithms and FastICA settings lead to different results. For some of those cases, the choice of weights affect, as well. Consider Figure 5.9. FICA3 cases show a peak close to $c_p = 0.01$, as opposed to $c_p = 0$. That is true for the first two weight combinations. However, that anomaly is not true for the other weight combinations, nor for the other FastICA configurations. Similar effects are observed in the noisy cases (Figures 5.11 and 5.13). The ITEFICA

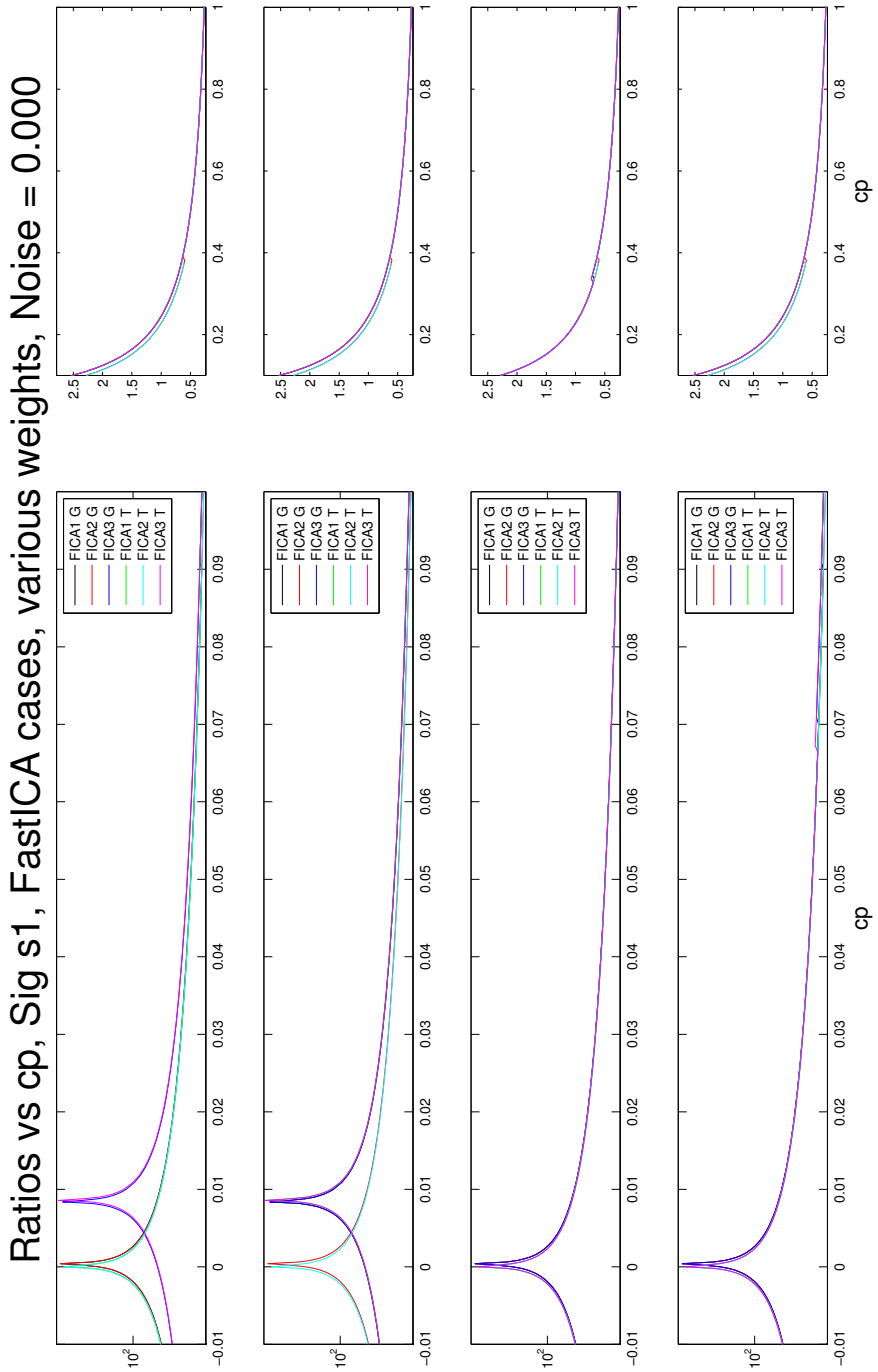



Figure 5.9: Ratios vs. $c_p$. FastICA cases. Different weight values ($w_1$, $w_p$). To the left, $c_p$ values from $-0.01$ to 0.1; plots shown in log scale. To the right, $c_p$ values from 0.1 to 1; plots in linear scale. No noise added to the mixtures.

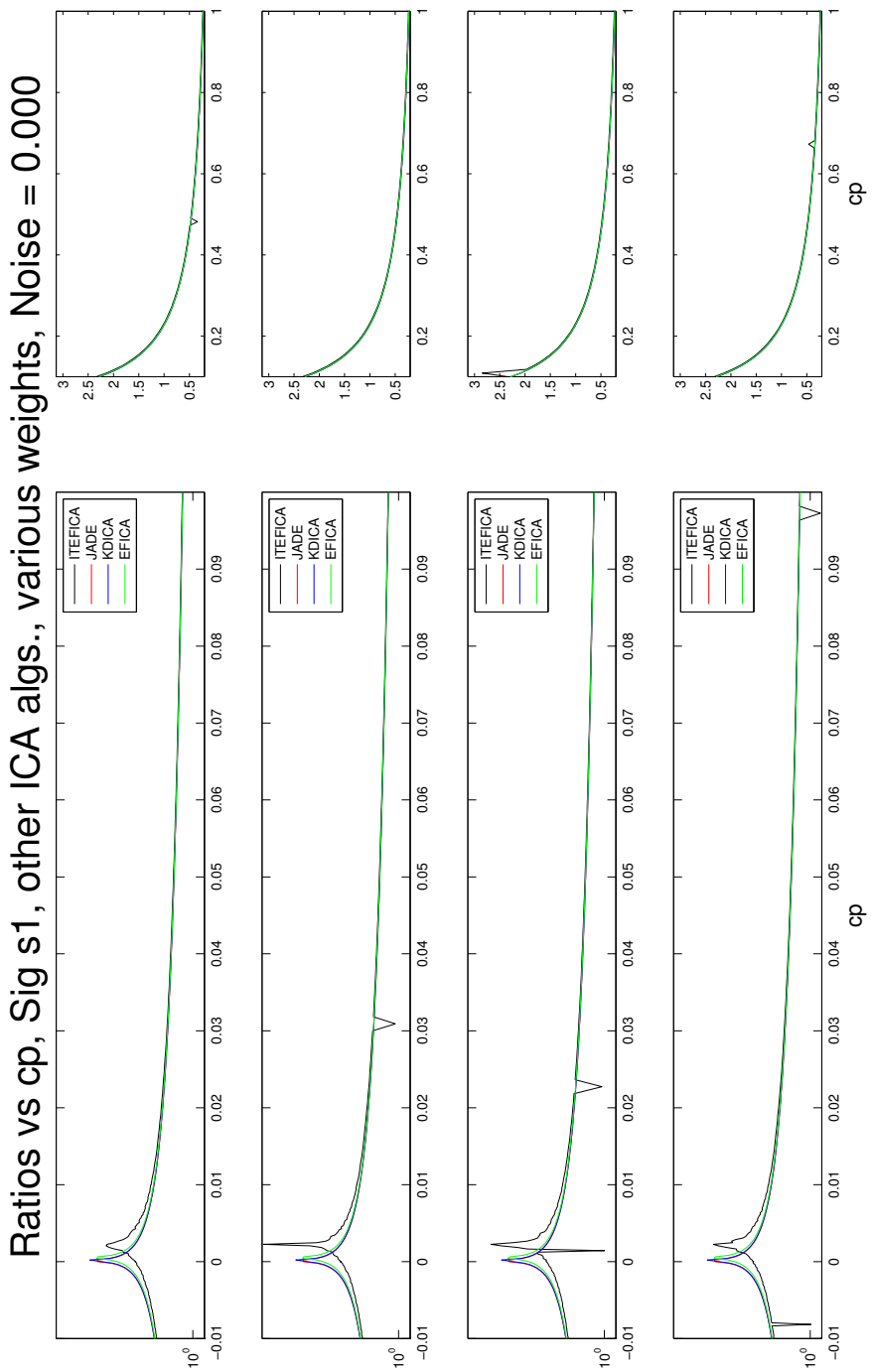



Figure 5.10: Ratios vs. $c_p$. Other ICA algorithms. Different weight values $(w_1,\ w_p)$. To the left, $c_p$ values from $-0.01$ to 0.1; plots shown in log scale. To the right, $c_p$ values from 0.1 to 1; plots in linear scale. No noise added to the mixtures.

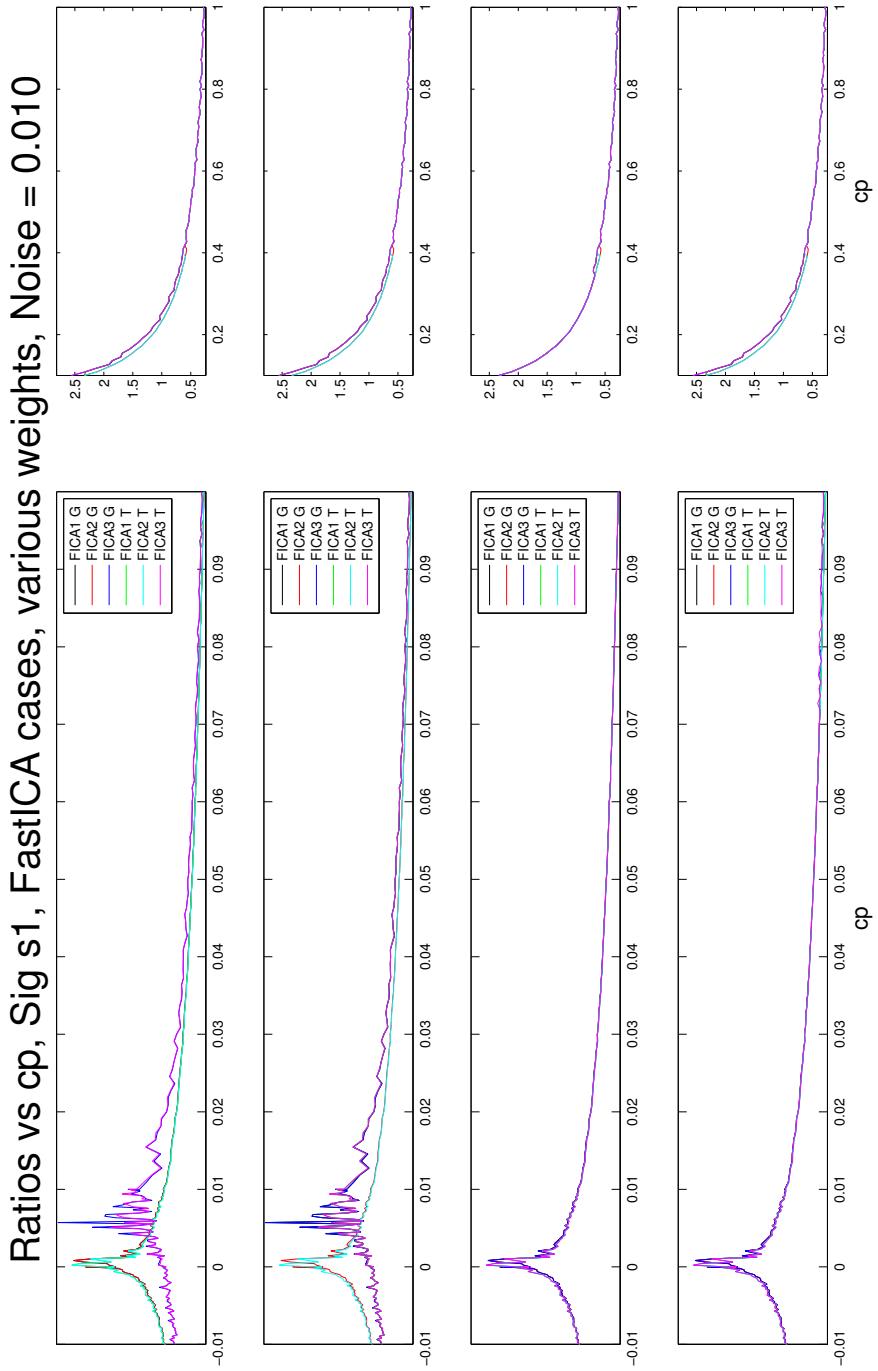



Figure 5.11: Ratios vs. $c_p$. FastICA cases. Different weight values $(w_1, w_p)$. To the left, $c_p$ values from $-0.01$ to 0.1; plots shown in log scale. To the right, $c_p$ values from 0.1 to 1; plots in linear scale. Noise added to the mixtures ($\sigma = 0.01$).

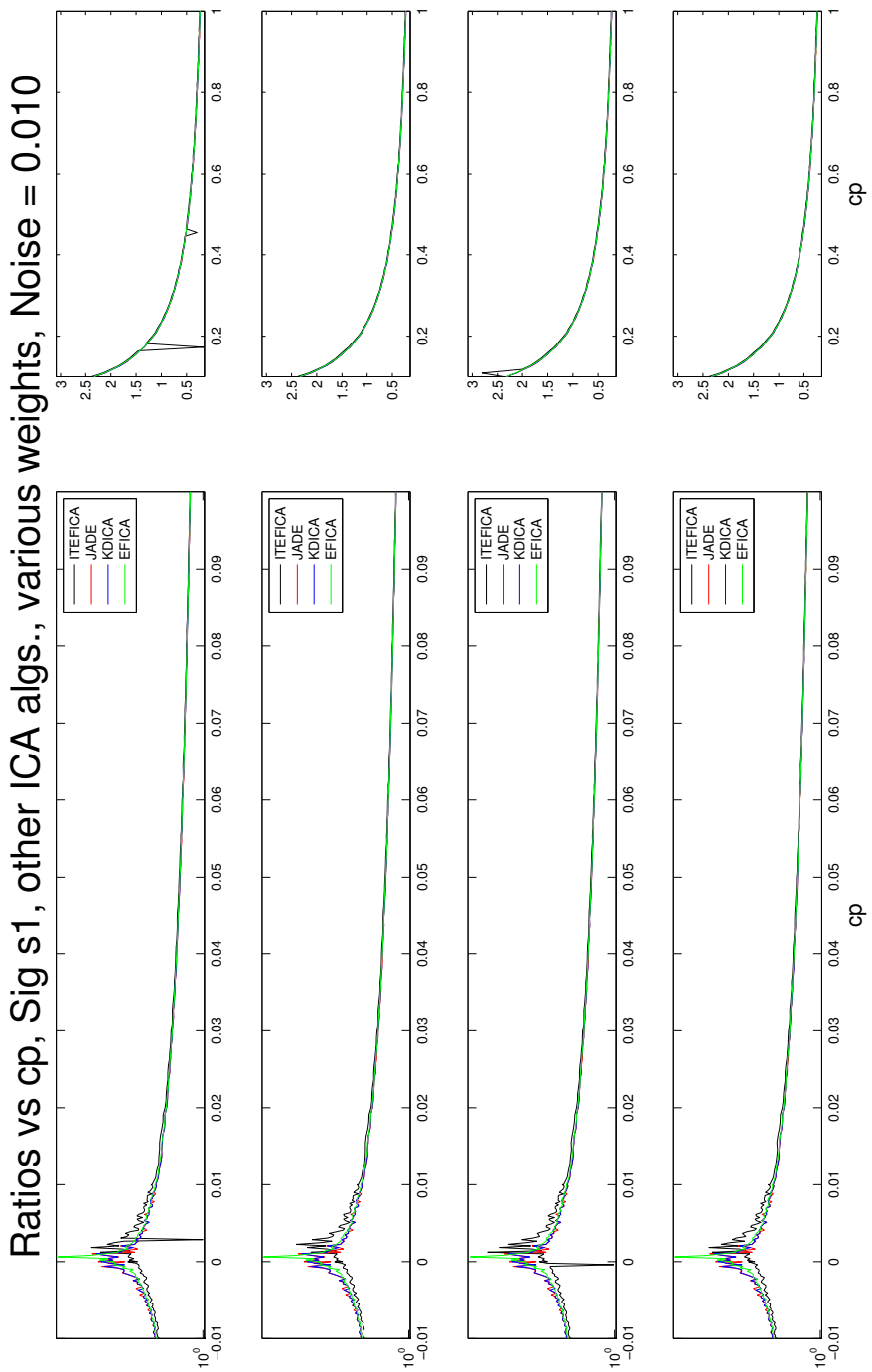



Figure 5.12: Ratios vs. $c_p$. Other ICA algorithms. Different weight values $(w_1,\ w_p)$. To the left, $c_p$ values from $-0.01$ to 0.1; plots shown in log scale. To the right, $c_p$ values from 0.1 to 1; plots in linear scale. Noise added to the mixtures ($\sigma = 0.01$).

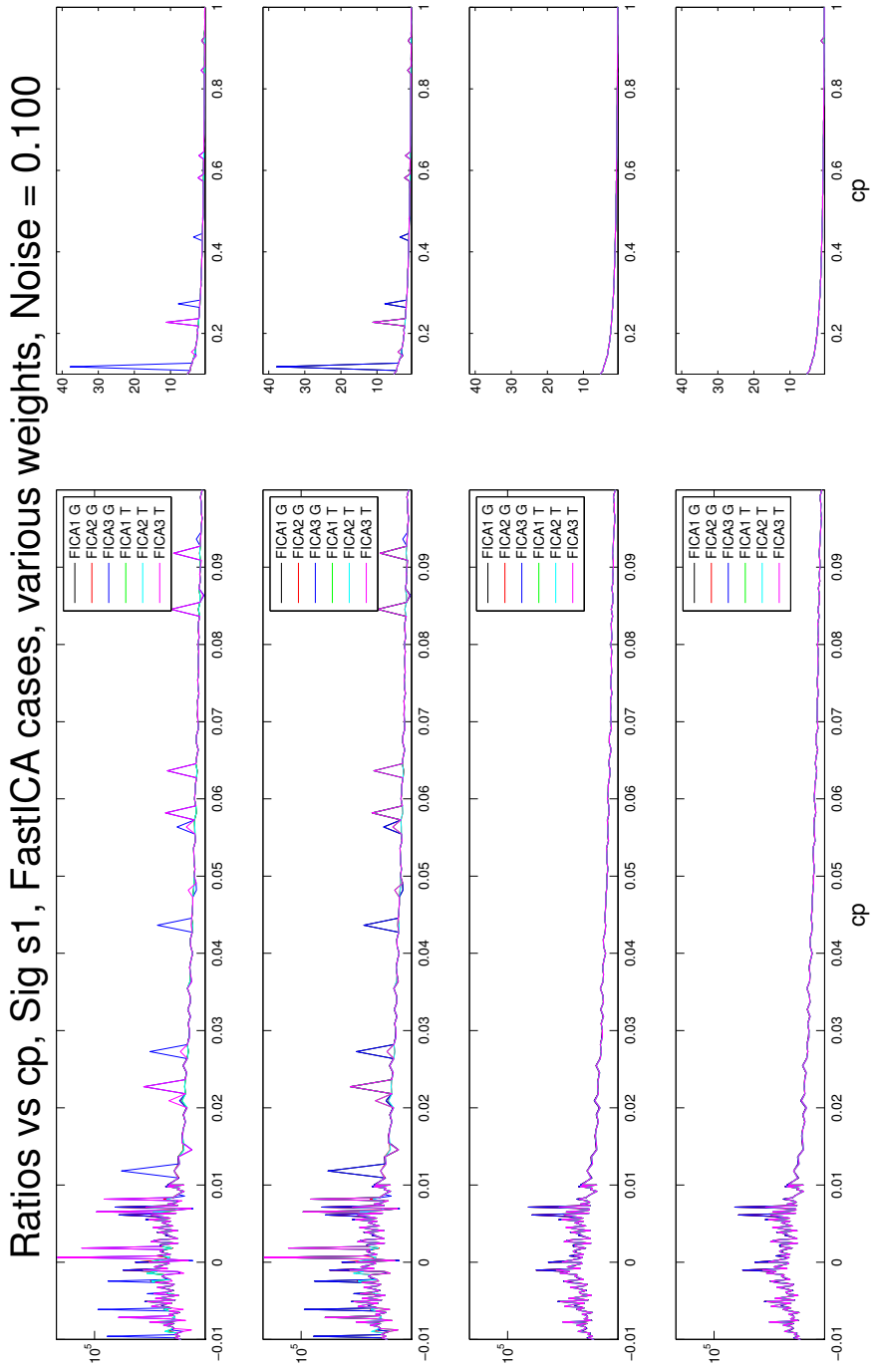



Figure 5.13: Ratios vs. $c_p$. FastICA cases. Different weight values $(w_1, w_p)$. To the left, $c_p$ values from $-0.01$ to 0.1; plots shown in log scale. To the right, $c_p$ values from 0.1 to 1; plots in linear scale. Noise added to the mixtures ($\sigma = 0.1$).

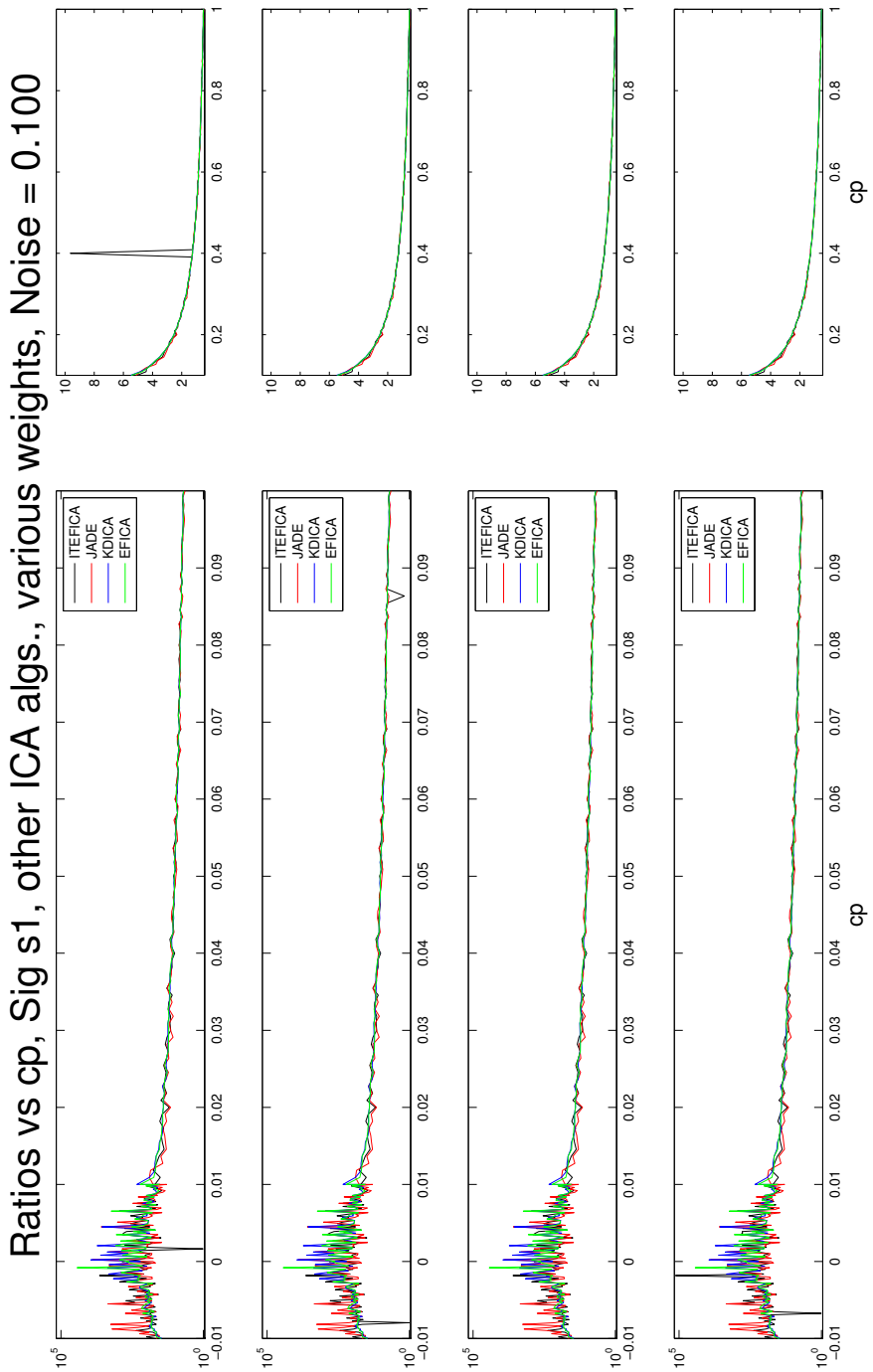



Figure 5.14: Ratios vs. $c_p$. Other ICA algorithms. Different weight values ($w_1$, $w_p$). To the left, $c_p$ values from $-0.01$ to 0.1; plots shown in log scale. To the right, $c_p$ values from 0.1 to 1; plots in linear scale. Noise added to the mixtures ($\sigma = 0.1$).

algorithm (Figures 5.10, 5.12 and 5.14) produces such variations, as well, although the peaks are closer to $c_p = 0$ with respect to the FICA3 cases. ITEFICA also produces some strange values. Those can be seen as peaks and valleys that deviate from the expected tendency of the ratio values. Those other anomalies are also observed for the FICA3 cases, but again, only for the first two weight combinations.

The anomalies can be explained by the algorithms converging to local maxima, or not converging at all. As we have observed in this and the previous sections, the choice of $A_{init}$, and algorithm implementation may have an effect on the final ICA output, and thus, the GUSSS method. After analyzing the effects of the different parameters of the ICA and GUSSS method, we selected the best ones in terms of stability, separability and resilience to noise. We used the best settings for all the experiments described next, regardless of the application. There is no further need of adjusting the parameters for each case.

The noted tendency had been observed in our results using THz signatures. The following table reproduces Table 4.1. The true positive, true negative and correct classification percentages are directly proportional to the contribution of the target materials in the mixtures ($c_p$).

Table 5.1: Overall average percentages of True Positives, True Negatives and Correct Classification rates (18 experiments).

| $c_p$ | TP (%) | TN (%) | CC (%) |
|---|---|---|---|
| [0.10, 0.20] | 67.1 | 71.9 | 69.5 |
| [0.20, 0.30] | 77.1 | 79.3 | 78.2 |
| [0.30, 0.40] | 85.3 | 85.1 | 85.2 |
| [0.40, 0.50] | 88.9 | 92.0 | 90.5 |
| [0.50, 0.60] | 93.5 | 94.4 | 93.9 |
| [0.60, 0.70] | 95.1 | 96.8 | 96.0 |
| [0.70, 0.80] | 96.4 | 97.8 | 97.1 |
| [0.80, 0.90] | 97.4 | 98.0 | 97.7 |
| [0.90, 0.95] | 97.5 | 98.3 | 97.9 |

## 5.4 Independence Assumption

A fundamental assumption of our method is the independence of source signals that may be present in observed mixtures. We are interested in determining the validity of such assumption, and how varying degrees of dependence may affect the performance of our method. In this section we present several experiments conducted to address research question 2. For many of the experiments we used the sound dataset described in Section 5.2. We also conducted experiments using real sEMG data.

### 5.4.1 Measures of Independence

We tested the following measures of independence: Mutual Information (MI), Distance Correlation (DCOR) and Kernel Generalized Variance (KGV) (Sec. 2.5.5 - 2.5.7). To get an initial idea of the behavior of these measures, we ran tests similar to those presented in sec. 5.3.4, calculating the dependence as a function of the coefficient $c_p$.

Mixtures were created using the sound signals described above. As before, a mixture is given by

$$x = \sum_{i \neq p} c_i s_i + c_p s_p + n$$

Given a particular $c_p$ value, the remaining coefficients $c_i$, $i \neq p$, are randomly chosen so that $\sum c_i = 1$. As before, $c_p$ represents a percentage of the signature in the mixture. For example, $c_p = 0.3$ means that the mixture is created adding 0.3 times the normalized signature $s_p$ and 0.7 times a normalized combination of the remaining signatures. Different noise levels $n$ ($\sigma$) were added.

For each noise level, and for every $c_p$ value, 50 cases were run, and averages were obtained. To illustrate the results, Figures 5.15 and 5.16 show the plots for signatures $s_1$ and $s_3$, respectively. Similar results were obtained for the other signatures. For instance, consider Figure 5.15. The top subplot corresponds to results using MI. At $c_p = 0.9$, the green dot represents the average of the MI values obtained between signature $s_1$ and 50 different mixtures $x$ constructed as explained above, with an added Gaussian noise ($\mu = 0$, $\sigma = 0.2$).

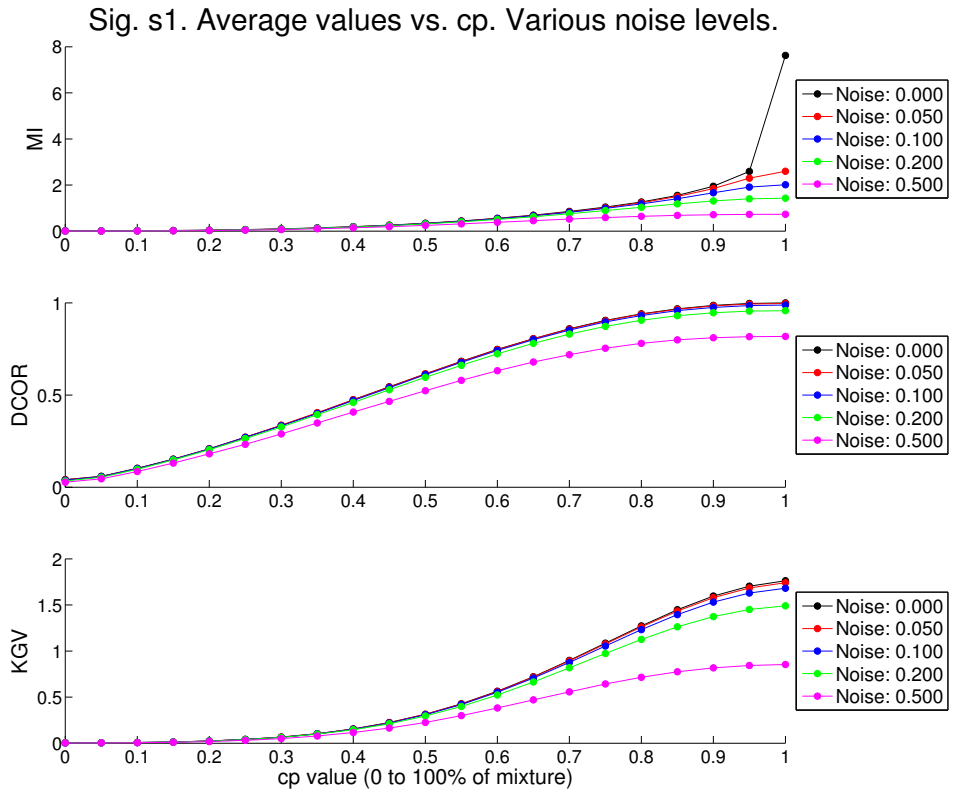



Figure 5.15: MI, DCOR and KGV average values vs. $c_p$. Signature $s_1$. Various noise levels added.

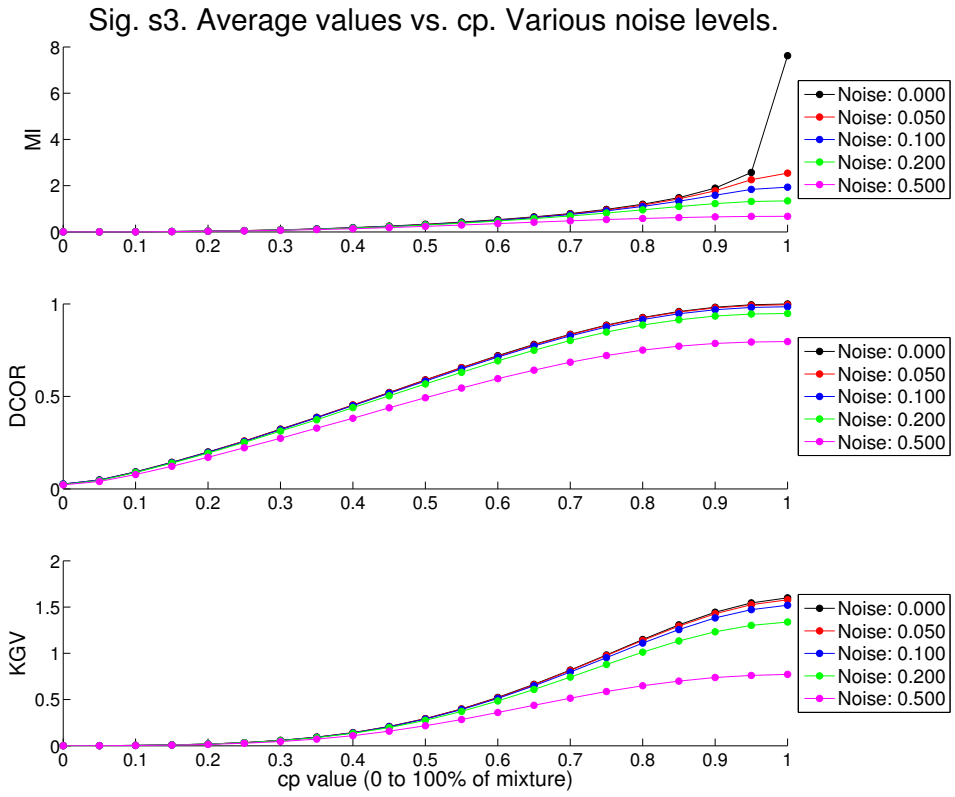



Figure 5.16: MI, DCOR and KGV average values vs. $c_p$. Signature $s_3$. Various noise levels added.

The results above show a clear correlation between the $c_p$ coefficients and the dependence measurement values. Those values are high for mixtures containing a large fraction ($c_p$ close to 1) of the signature of interest. On the contrary, the dependence values decrease as $c_p$ decreases. The noise level added to the mixtures also affect the dependence values. As expected, higher noise levels result in mixtures being less dependent to the particular signature considered. These tendencies can be seen very clearly in the figures above. Similar behaviors were observed for all three independence measures, and for all signatures.

### 5.4.2 Performance as a Function of Dependency

The main assumption of the ICA algorithms is that of the statistical independence of the components that constitute a mixture. Research question 2 is raised to determine how valid this assumption is, and how the GUSSS method would be affected by various degrees of dependence of the components.

#### 5.4.2.1 Tests on Synthetic Dataset

Once again, we ran some tests using the sound data set. For each signature $s_p$, 200 mixtures were created: 100 including $s_p$, and 100 without $s_p$. MI, DCOR, KGV and GUSSS ratios were calculated between the signature and each mixture. Averages were calculated. This was repeated for different $c_p$ values, and for different noise levels. The results are shown in figures 5.17 - 5.19. For instance, Figure 5.17 shows bar diagrams representing the average MI values (first column), DCOR (second) and KGV (third) and inverse of GUSSS ratios (fourth), for mixtures with the target signatures accounting for 90% ($c_p = 0.9$) of the mixtures – "Present" bars (P). The "Not Present" bars (NP) represent the average values obtained for the mixtures not containing the target signatures. Each row of diagrams corresponds to different noise levels used. The results are averages over all eight sound signatures.

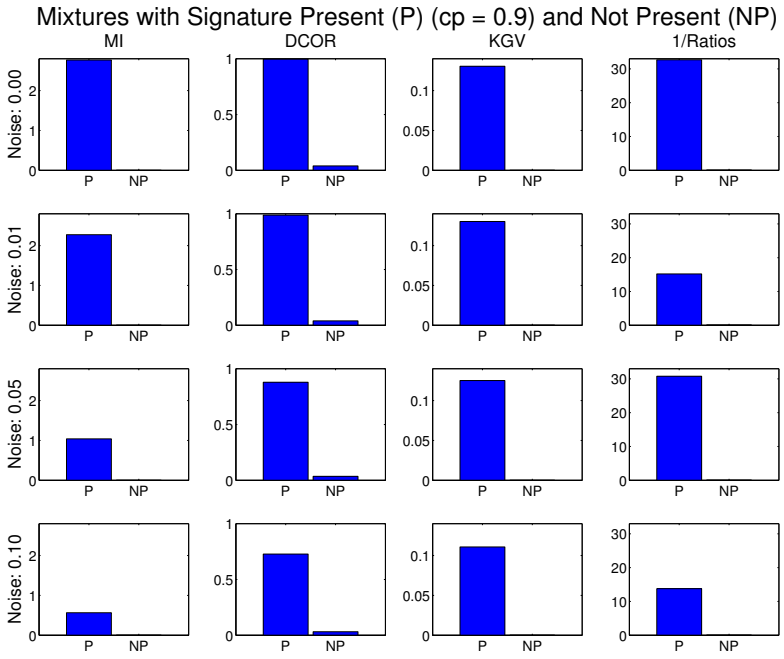



Figure 5.17: Average MI, DCOR, KGV and inverse GUSSS ratio values. $c_p = 0.9$. Noise: $\sigma = 0.00,\ 0.01,\ 0.05,\ 0.10$.

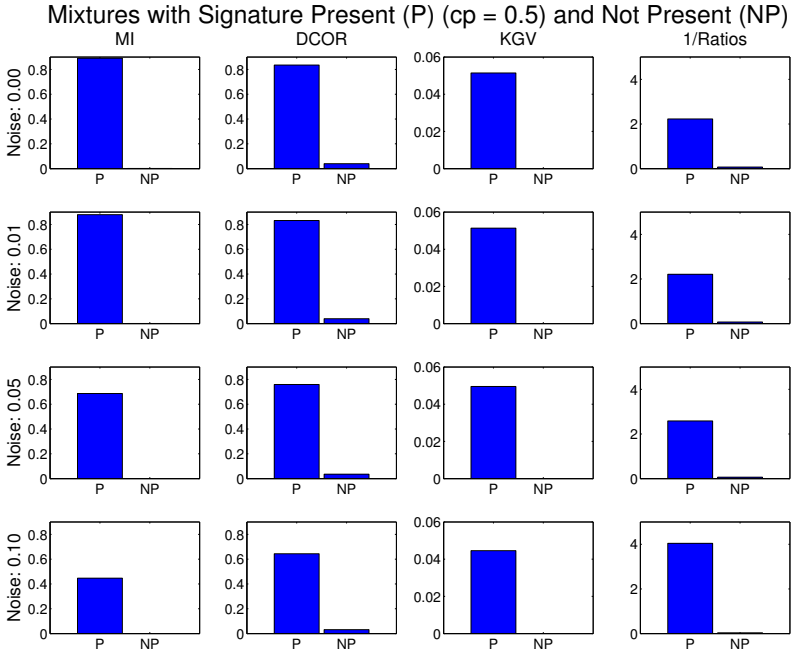



Figure 5.18: Average MI, DCOR, KGV and inverse GUSSS ratio values. $c_p = 0.5$. Noise: $\sigma = 0.00,\ 0.01,\ 0.05,\ 0.10$.

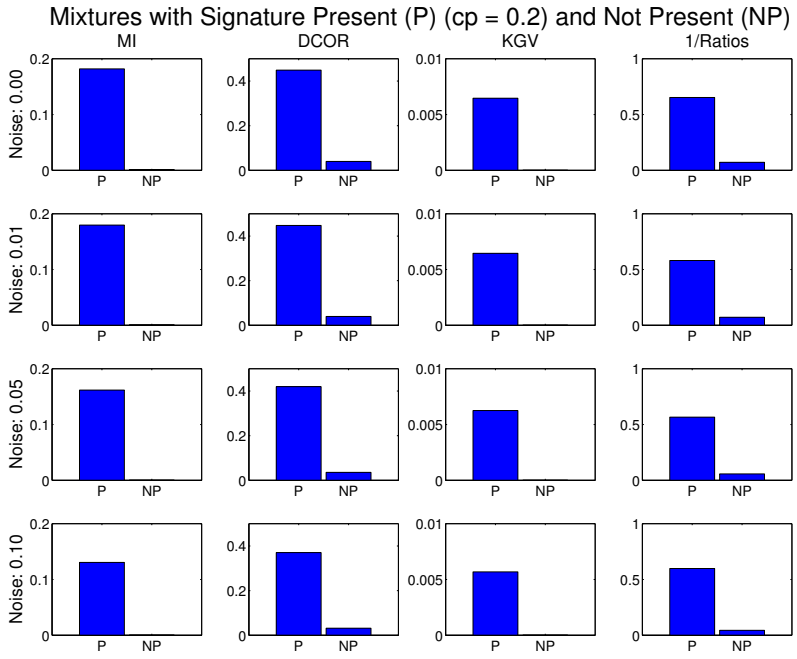



Figure 5.19: Average MI, DCOR, KGV and inverse GUSSS ratio values. $c_p = 0.2$. Noise: $\sigma = 0.00,\ 0.01,\ 0.05,\ 0.10$.

The results above once again show how mixtures containing the target signatures are more dependent to the those signatures, as compared to mixtures without the target signatures. The same tendencies observed in the previous section are observed here. The higher the $c_p$ coefficient and the lower the noise, the more separation between the "Present" and "Not Present" cases.

Recall that the GUSSS ratio is the inverse of the estimated $c_p$ coefficient. In these experiments we show results of the inverse of the ratios instead of the ratios themselves, so the comparison with the dependence measures is direct. As expected, those inverse ratios show clear separability between the "Present" and "Not Present" cases. The smaller the contribution of the particular signature to the mixture (and thus the more independent the signature and the mixtures are), the smaller the inverse of the GUSSS ratios. The separation between the "Present" and "Not Present" cases is greater for higher $c_p$ coefficients.

#### 5.4.2.2 Tests on sEMG Dataset

As mentioned before, in addition to the tests using artificial mixtures of the sound signatures, we performed several tests using real sEMG data. The goal was to determine how similar (in terms of dependence) the different gestures are. If there is a high degree of dependence between two gestures, it is expected that there will be confusion between them in classification tasks.

We took sEMG sample signals from the dataset used in [55] and described in Sec. 4.2.2. We calculated MI and DCOR values between (1) individual sample signals; (2) individual sample signals and mean-based signatures; and (3) mean-based signatures. In all three cases, calculations were done both within same gestures and between different gestures. We left KGV out, given that, as seen in the previous subsection, the result tendencies were similar to those observed for MI and KGV – though the numerical scales are different.

Figures 5.20 and 5.21 show the results for two of the test subjects. The complete set of results is shown in Appendix B. To illustrate, consider Figure 5.20. The top left bar plot shows averages of MI and DCOR values obtained for gesture 1. The first set of three bars (#1 on the $x$-axis) represents averages of measurements within gesture 1. The second set of three bars (#2) represents averages of measurements between gesture 1 and gesture 2. The fifth set (#5) is for measurements between gesture 1 and gesture 5. The dark green bars represent measurements between individual samples. The light green bar represents measurements between samples from gesture 1 and learned signatures from the different gestures. And the yellow bars represent measurements between signatures learned for gesture 1 and signatures for other gestures.

The top right bar plot represents measurements between gesture 2 and all gestures. Similar plots are shown for gestures 3, 4 and 5. The bottom right image is a graphical representation of the confusion matrix for classification of the 5 gestures (dataset described in Sec. 4.2.2). The values are average percentages over a 5-fold cross validation. It is important to highlight that the classification was done using GUSSS ratios only (Sec. 3.2.2.1). No other features were included. A simple distance classifier was used (Sec. 3.2.2.3).

The reason behind not using the full GUSSS or HiGUSSS method was to correlate the independence measures and a ratio-based classification performance alone.

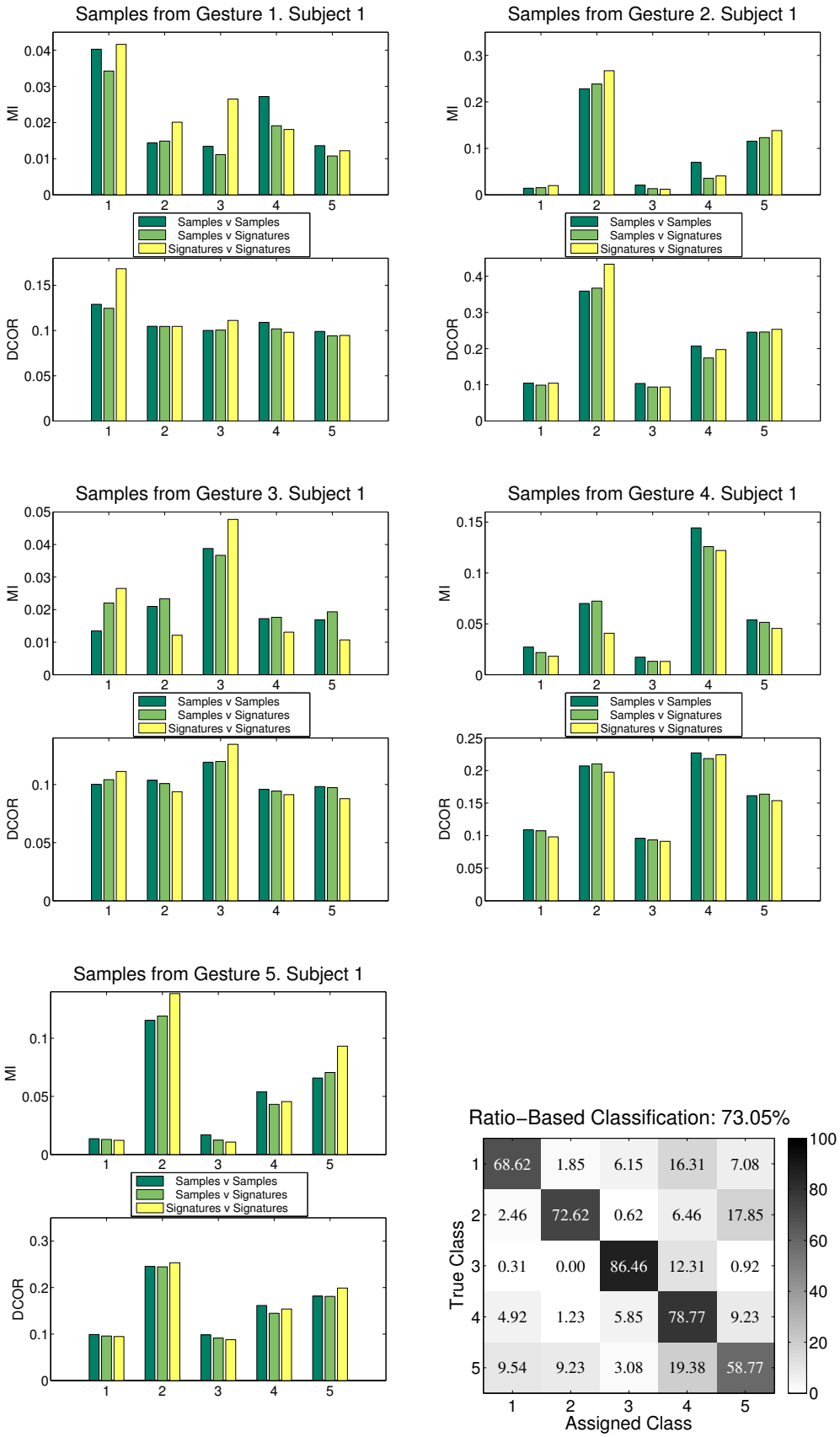



Figure 5.20: MI and DCOR average values. 5 different gestures. Test subject 1.

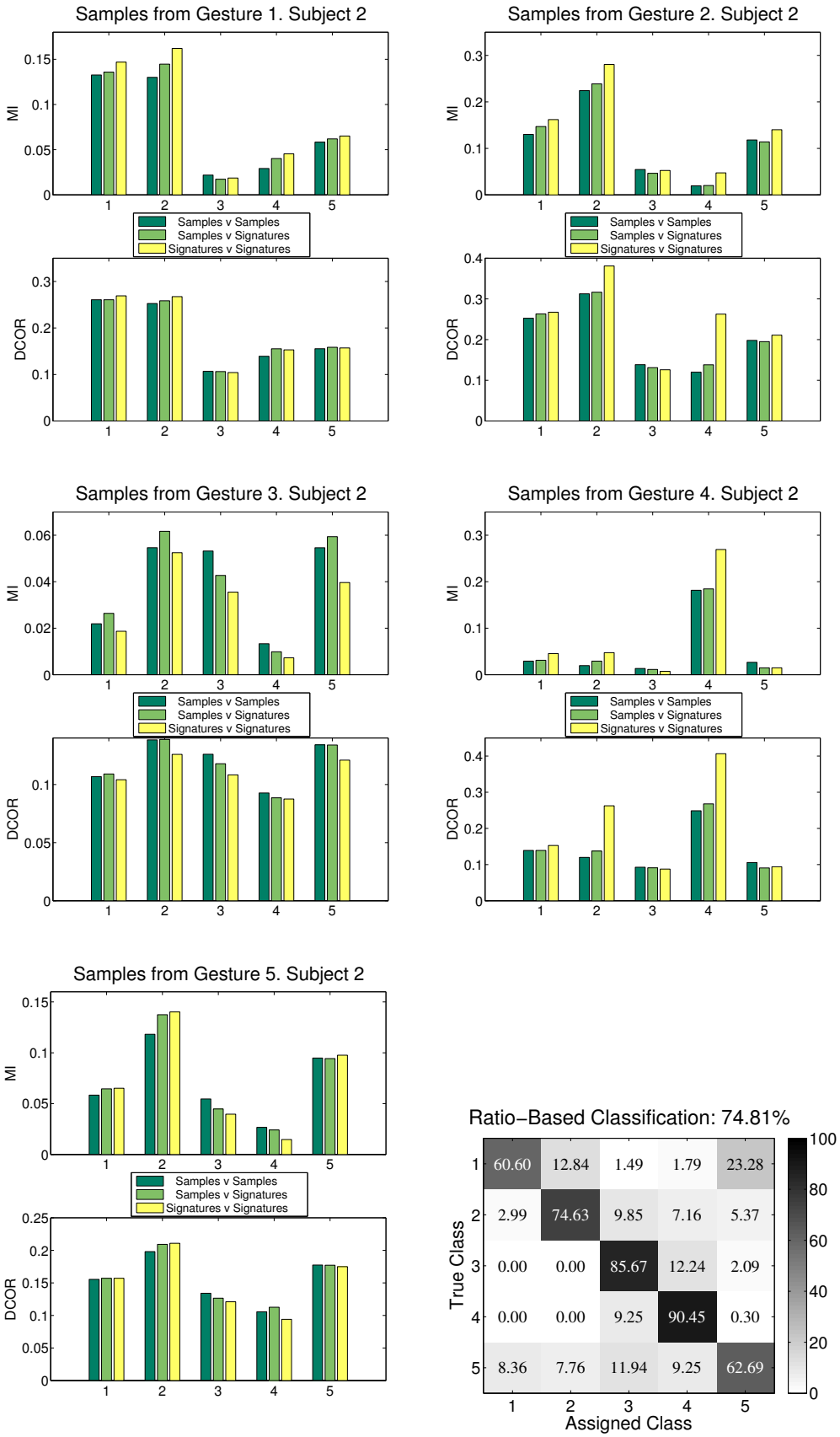



Figure 5.21: MI and DCOR average values. 5 different gestures. Test subject 2.

The results above show that some gestures are very dependent with the other gestures, while other gestures show little dependence (e.g. gesture 4 of subject 2). As opposed to the sound examples, sEMG signals generated by the same set of muscles have many common components, which explain the high degree of dependence. Still, on average, for the most part, samples from one gesture is more similar to samples and signatures from the same gesture, as expected.

The classification was good for gestures showing low dependence values with other gestures. On the other hand, confusion and a lower classification performance was obtained in cases where there was a high degree of dependence between the gestures. Nevertheless, the complete GUSSS and HiGUSSS methods have shown very good classification performance on sEMG and other signals, taking advantage of the GUSSS ratios and a few additional features. Further demonstration of this good performance is presented in the coming sections.

## 5.5 Learning Signatures

As seen in some of the applications such as Assistive Technology, the signatures needed in our method may not be available. In such cases, there is a need for learning those signatures from training data. It is fair to ask if the approach used thus far for learning the signatures is appropriate, or even valid, and if there are better approaches for learning.

To address research question 3, we investigated and compared several learning approaches. We looked into using ICA on all the available training signals to separate the main components and used them to construct the signatures for each of the classes or gestures. We also looked into clustering approaches. Next we describe the tests.

For all cases, it is assumed that there is a labeled training set with $K \times M$ signals – i.e. $M$ signals from each of the $K$ different classes (different materials that need to be identified, or muscle patterns or gestures that need to be recognized, etc.). The goal is to learn signatures for each class, so they can be used in the GUSSS method.

### 5.5.1 Mean-based Approach

The results presented in Chapter 4 were obtained using signatures obtained as described in Sec. 3.2.2.3. That is, the signatures are calculated by simply averaging the training signals grouped per class: all $M$ training signals belonging to the same class $k$ are averaged creating a single signature: $sig_k = \frac{1}{M}\left(\sum_{class\,k} x_m\right)$, where $x_m$ is the $m^{th}$ training signal of class $k$.

### 5.5.2 Cluster-based Approach

A second approach we investigated is learning signatures through clustering of the entire training set. In particular, we used a simple $K$-means algorithm, and we used two different distance measures: euclidean distance and correlation distance.

When using euclidean distance, each centroid is the mean of the points in that cluster. The difference between this approach and the mean-based approach is that there is no guarantee that all samples from a particular class will end up in the same cluster. If that were the case, then the signatures would be identical to the mean-based ones.

When using correlation, the distance value is given by one minus the sample correlation between points (treated as sequences of values). Each centroid is the component-wise mean of the points in that cluster, after centering and normalizing those points to zero mean and unit standard deviation. The idea behind using correlation distance comes from the fact that dependence implies correlation, so this clustering may provide better signatures for our ICA based method.

### 5.5.3 ICA Decomposition Approach

A further approach was to apply ICA to the training signals to obtain the basic components that constitute the different classes/gestures. Then, the signatures are created by selecting the components that optimize certain criteria, for each class.

The motivation behind this approach came from the study of sEMG signals. Recall that in the assistive technology and otolaryngology applications we perform classifi-

cation using those sEMG signals. Learning the different signatures representing the gestures is necessary, given that no true signatures are known. Consider, for instance, a set of sEMG training signals collected from different hand gestures. The sEMG signals are constituted by muscle activation patterns, many of which may be present for more than one gesture – though with different intensities. The idea is then to find those basic components that best describe each particular gesture – according to certain criteria – and reject those components which may be present in all or most of the gestures. Signatures can then be constructed using the better (most discriminative) components.

This ICA decomposition approach was also inspired by the Bag-of-Words (BoW) model. This model has been used for natural language processing and information/text retrieval, where text is represented as a bag (multiset) of its words. Those ideas have led to methods for document classification (e.g. spam filtering) [126]. BoW ideas have also been used in computer vision and speech analysis. In [127], a method for object localization in video using Bag-of-Visual-Words (BoVW) was presented. A method for visual categorization was described in [128]. The concept of Bag-of-Audio-Words (BoAW) was used in [129] for detection of negative emotions in speech signals.

In our case, the basic independent components in the signals may be considered "words", and the different gestures may be considered collections (bags) of the independent components. Next we formalize the decomposition process, the selection criteria, and the final signature construction.

#### 5.5.3.1 ICA Decomposition

Recall the ICA model. A set of observed signals $X = \{x_1, \ldots, x_M\}$ are related to a set of independent sources $S = \{s_1, \ldots, s_N\}$ through the linear model $\mathbf{X} = A\mathbf{S}$, where $\mathbf{X} = [x_1, \ldots, x_M]^T$ and $\mathbf{S} = [s_1, \ldots, s_N]^T$ are vector representations of the sets $X$ and $S$, and $A$ is an $M \times N$ mixing matrix.

Let $X_k$ be the set of training signals from class $k$, i.e. $X_k = \left\{x_1^k, \ldots, x_{M_k}^k\right\}$. Applying ICA to $X_k$ results in the set of sources $S_k = \left\{s_1^k, \ldots, s_{N_k}^k\right\}$, where $M_k \geq N_k$. The sets $S_k$, $k = 1, \ldots, K$ represent the basic components of the different $K$ classes.

For each class, we want a signature constructed out of those components that

maximize the dependence with samples of the same class, but minimize the dependence with samples of other classes. For that, we follow a two-step approach: 1) ranking the components and building signature candidates; and 2) selection of the final signature through a validation set. We describe the two steps next.

#### 5.5.3.2 Signature Candidates

The goal here is to find those components with high dependence to the class in question, and low dependence to the other classes.

Define the function

$$F(s,k) = \frac{\frac{1}{M_k}\sum_{m_k=1}^{M_k} MI\left(s, x_{m_k}^k\right)}{\frac{1}{K-1}\sum_{j\neq k}\left[\frac{1}{M_j}\sum_{m_j=1}^{M_j} MI\left(s, x_{m_j}^j\right)\right]} \tag{5.2}$$

where $MI(s,x)$ is the mutual information value between component $s$ and sample $x$. Note that other independence measures could have been used. We selected MI over DCOR or KGV due to its computational efficiency (results using either measure were similar).

The numerator of Function 5.2 is the average of the MI values between component $s$ and training signals from class $k$. The denominator is the average over all other classes (different from class $k$) of the averages of the MI values between component $s$ and training signals from those other classes. This ratio will be high when component $s$ is very related (dependent) to class $k$, but at the same time, is not very related to the other classes. Note that function $F$ is analogous to the $J$ measures for class separability using scatter matrices [51]. Numerically, those $J$ measures are larger when the between-class scatter is large, or when the within-class scatter is small.

For each class, we sort the components $\left\{s_1^1,\ldots,s_{N_1}^1,s_1^2,\ldots,s_{N_2}^2,\ldots,s_1^K,\ldots,s_{N_k}^K\right\}$ in descending order, according to their $F$ value. Let $\tilde{S}_k$ be the sorted set of components for class $k$: $\tilde{S}_k = \left\{\tilde{s}_1^k,\ldots,s_{N_A}^k\right\}$, where $F\left(\tilde{s}_1^k,k\right) \geq F\left(\tilde{s}_2^k,k\right) \geq \ldots \geq F\left(\tilde{s}_{N_A}^k,k\right)$, and $N_A = \sum_{k=1}^{K} N_k$.

We then create a set of candidate signatures for class $k$ using weighted combinations

of the sorted components: $C_k = \left\{c_1^k, \ldots, c_{N_A}^k\right\}$, where $c_n^k = \sum_{j=1}^{n} \bar{a}_j^k \tilde{s}_j^k$. The weight $\bar{a}_j^k$ is a normalized average of the mixing coefficients obtained through the ICA decomposition for component $\tilde{s}_j^k$.

#### 5.5.3.3 Final Signatures

The final signature for class $k$ is chosen out of the set of candidates $C_k$ described above. We use validation sets of signals from each class: $Y_k = \left\{y_1^k, \ldots, y_V^k\right\}$, $k = 1, \ldots, K$. $V$ is the number of validation signals, which is chosen to be the same for all $k$. Signature $sig_k$ is selected follows:

$$sig_k = \underset{c_n^k \in C_k}{arg\,max} \left\{ \frac{\frac{1}{V}\sum_{i=1}^{V} MI\left(c_n^k, y_i^k\right)}{\frac{1}{K-1}\sum_{p \neq k}\left[\frac{1}{V}\sum_{j=1}^{V} MI\left(c_n^k, y_j^p\right)\right]} \right\} \tag{5.3}$$

Eq. 5.3 is very similar to Eq. 5.2. The numerator is the average of the MI values between component candidate $c_n^k$ and validation signals from class $k$. The denominator is the average over all other classes (different from class $k$) of the averages of the MI values between candidate $c_n^k$ and validation signals from those other classes. This ratio will be high when candidate $c_n^k$ is very related (dependent) to class $k$, but not very related to the other classes. The final signature is the candidate which gives the maximal ratio.

#### 5.5.3.4 All Class vs. Per Class Training Signals

The first step described above consists of taking training samples and applying ICA to get a set of basic components, which will then be used to generate candidate signatures for each class. We can choose to combine all training signals from all classes and run ICA on that combined set to get the basic components. Or we can run ICA on the different training sets separately, get the basic components per class, and use those to find the candidates for the corresponding class. We tested both approaches.

### 5.5.4 Results using the Different Learning Approaches

We applied the GUSSS method for classification of the sEMG signals described in Sec. 4.2.2, this time using the different approaches for learning signatures described above: Mean-based, Cluster-based using Euclidean distance, Cluster-based using Correlation distance, ICA-decomposition-based using all training signals combined, and ICA-decomposition-based using the training signals per class separately.

We tested using 3, 4 and 5 gestures. For each experiment, a 5-fold cross validation was performed. Tables 5.2 - 5.4 show the results, which consist of confusion matrices with average percentages over the 5-fold tests. Specifically, we show the average over all test subjects. The average correct classification percentages are also presented at the bottom of each table. The individual results for all the test subjects are presented in Appendix C.

Table 5.2: Confusion matrices for 3 gestures. The values are average percentages over a 5-fold cross validation, over all test subjects.

| Mean-based | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 91.91 | 5.34 | 2.75 |
| | 2 | 3.15 | 94.38 | 2.47 |
| gesture | 3 | 1.15 | 0.58 | 98.27 |
| Correct classification: 94.86% | | | | |

| Cluster-based Euclidean | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 91.55 | 5.74 | 2.71 |
| | 2 | 2.68 | 95.22 | 2.11 |
| gesture | 3 | 1.11 | 0.81 | 98.08 |
| Correct classification: 94.95% | | | | |

| Cluster-based Correlation | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 92.31 | 5.09 | 2.61 |
| | 2 | 3.58 | 93.59 | 2.83 |
| gesture | 3 | 1.19 | 0.58 | 98.23 |
| Correct classification: 94.71% | | | | |

| ICA-based All classes | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 91.81 | 5.18 | 3.01 |
| | 2 | 3.35 | 94.28 | 2.37 |
| gesture | 3 | 1.15 | 0.67 | 98.17 |
| Correct classification: 94.76% | | | | |

| ICA-based Per class | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 92.66 | 4.86 | 2.48 |
| | 2 | 3.35 | 94.18 | 2.46 |
| gesture | 3 | 1.06 | 0.68 | 98.26 |
| Correct classification: 95.03% | | | | |

Table 5.3: Confusion matrices for 4 gestures. The values are average percentages over a 5-fold cross validation, over all test subjects.

| Mean-based | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| True gesture | 1 | 89.85 | 5.53 | 2.44 | 2.18 |
| | 2 | 4.25 | 91.27 | 1.32 | 3.16 |
| | 3 | 0.63 | 0.95 | 98.29 | 0.13 |
| | 4 | 1.14 | 0.79 | 0.77 | 97.30 |
| Correct classification: 94.18% | | | | | |

| Cluster-based Euclidean | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| True gesture | 1 | 89.16 | 5.42 | 2.27 | 3.15 |
| | 2 | 3.72 | 92.13 | 0.90 | 3.25 |
| | 3 | 1.02 | 0.92 | 97.64 | 0.43 |
| | 4 | 0.98 | 0.70 | 0.76 | 97.56 |
| Correct classification: 94.12% | | | | | |

| Cluster-based Correlation | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| True gesture | 1 | 89.56 | 4.73 | 2.50 | 3.21 |
| | 2 | 3.83 | 91.15 | 1.79 | 3.24 |
| | 3 | 1.14 | 0.86 | 97.69 | 0.30 |
| | 4 | 1.10 | 0.70 | 0.85 | 97.35 |
| Correct classification: 93.34% | | | | | |

| ICA-based All classes | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| True gesture | 1 | 89.37 | 5.60 | 2.44 | 2.59 |
| | 2 | 2.99 | 92.75 | 1.36 | 2.89 |
| | 3 | 0.72 | 0.86 | 98.10 | 0.31 |
| | 4 | 1.22 | 0.61 | 0.90 | 97.26 |
| Correct classification: 94.37% | | | | | |

| ICA-based Per class | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| True gesture | 1 | 90.04 | 5.35 | 2.05 | 2.55 |
| | 2 | 3.35 | 92.25 | 1.27 | 3.12 |
| | 3 | 1.16 | 0.86 | 97.76 | 0.23 |
| | 4 | 1.02 | 0.65 | 0.90 | 97.43 |
| Correct classification: 94.37% | | | | | |

Table 5.4: Confusion matrices for 5 gestures. The values are average percentages over a 5-fold cross validation, over all test subjects.

| Mean-based | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| True gesture | 1 | 84.10 | 4.41 | 1.98 | 2.72 | 6.78 |
| | 2 | 2.70 | 90.35 | 0.96 | 0.47 | 5.53 |
| | 3 | 1.35 | 0.32 | 96.93 | 0.44 | 0.97 |
| | 4 | 0.86 | 0.38 | 0.66 | 95.71 | 2.38 |
| | 5 | 4.05 | 0.35 | 3.77 | 2.32 | 89.51 |
| Correct classification: 91.32% | | | | | | |

| Cluster-based Euclidean | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| True gesture | 1 | 83.90 | 5.01 | 2.20 | 3.33 | 5.56 |
| | 2 | 2.30 | 92.74 | 0.45 | 0.48 | 4.03 |
| | 3 | 1.15 | 0.41 | 97.30 | 0.18 | 0.96 |
| | 4 | 0.82 | 0.66 | 0.61 | 95.86 | 2.05 |
| | 5 | 3.73 | 1.38 | 3.14 | 2.85 | 88.91 |
| Correct classification: 91.74% | | | | | | |

| Cluster-based Correlation | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| True gesture | 1 | 84.89 | 3.80 | 2.66 | 2.68 | 5.98 |
| | 2 | 2.52 | 91.36 | 1.42 | 0.48 | 4.22 |
| | 3 | 1.56 | 0.26 | 96.99 | 0.17 | 1.01 |
| | 4 | 1.07 | 0.43 | 0.81 | 95.87 | 1.82 |
| | 5 | 4.44 | 0.73 | 4.02 | 2.89 | 87.92 |
| Correct classification: 91.41% | | | | | | |

| ICA-based All classes | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| True gesture | 1 | 83.64 | 4.62 | 2.47 | 3.43 | 5.83 |
| | 2 | 2.56 | 90.06 | 0.93 | 0.52 | 5.93 |
| | 3 | 1.35 | 0.36 | 97.16 | 0.22 | 0.92 |
| | 4 | 0.78 | 0.35 | 0.72 | 96.59 | 1.56 |
| | 5 | 4.14 | 0.62 | 3.81 | 2.92 | 88.51 |
| Correct classification: 91.19% | | | | | | |

| ICA-based Per class | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| True gesture | 1 | 84.93 | 4.47 | 2.60 | 2.85 | 5.15 |
| | 2 | 2.47 | 91.40 | 0.81 | 0.46 | 4.86 |
| | 3 | 1.31 | 0.45 | 97.19 | 0.22 | 0.83 |
| | 4 | 0.98 | 0.48 | 0.66 | 96.27 | 1.61 |
| | 5 | 4.49 | 1.05 | 3.69 | 1.84 | 88.93 |
| Correct classification: 91.75% | | | | | | |

The results above are very good, in line of what was reported in Sec. 4.2.2. Even though the difference is not substantial among the different signature approaches, the ICA-decomposition-based signature approach, using the training signals per class separately, outperforms the other learning approaches. This is relevant, because a better way of learning signatures was found, compared to the original, mean-based approach that had been used thus far. However, it is important to highlight that the mean-based approach was already a very good way of learning signatures for the GUSSS method.

We also compared the results obtained in [55] for 3 to 9 gestures, using the new learning approaches. Table 5.5 shows the average classification results using Mean-based signatures and using the ICA-decomposition-based signature (per class). The latter gave the best overall results among the different learning approaches.

Table 5.5: Classification accuracy for a single test subject. Comparison between Mean-based signatures and ICA-decomposition-based signature (Per class).

| | Mean-based | ICA-based |
|---|---|---|
| 3 Gestures | 98.3 | 99.6 |
| 4 Gestures | 98.5 | 99.0 |
| 5 Gestures | 96.8 | 99.0 |
| 6 Gestures | 94.4 | 98.3 |
| 7 Gestures | 91.3 | 94.3 |
| 8 Gestures | 89.1 | 92.6 |
| 9 Gestures | 86.5 | 90.9 |

## 5.6 Comparison with Other Methods

Another important aspect to investigate is how methods that have been applied in other domains compare to our proposed method. Specifically, the comparison makes sense with methods that may make similar assumptions to those made for this research, or that may be applicable to domains for which our method has been applied. This is the idea of research question 4.

In this section we present results obtained using a detector which has been widely used in the area of Hyperspectral image analysis. Furthermore, we present results obtained using our GUSSS method on two Hyperspectral datasets.

### 5.6.1 Hyperspectral Target Detection

Spectral imaging sensors can measure scene spectra, in which various different material substances contribute. Hyperspectral sensors are able to capture hundreds of spectral bands, which significantly improves the resolution [130]. Hyperspectral target detection is the task of searching for a known signature within hyperspectral scenes [131]. Spectral matching algorithms can be used for the identification of unknown spectra based on a measure of similarity with one or more known spectra [132].

Both linear and nonlinear models have been developed for characterizing hyperspectral measurements. Using a linear mixing model, an observed spectrum $x$ for any pixel in a scene is given by

$$\begin{aligned} x &= a_1 s_1 + \ldots + a_N s_N + n \\ &= \sum_{i=1}^{N} a_i s_i + n = \mathbf{a}^T \mathbf{S} + n \end{aligned}$$

where $s_i$ is the $i^{th}$ endmember or signature spectrum, $a_i$ is the corresponding abundance, $N$ is the number of signatures, and $n$ is noise. $\mathbf{a}$ and $\mathbf{S}$ are vector representations of the abundances and signatures, respectively [130]. Note that this model is similar to those we have used in our various methods and applications.

Target detection can be seen as binary hypothesis test: target absent or target present. A well known detector in Hyperspectral image analysis is the Adaptive Coherence/Cosine Estimator (ACE) [131–133]. ACE computes the spectral angle between a pixel and the target signature after whitening and normalization by scene background statistics:

$$D_{ACE}(x) = \frac{(s-\mu)^T \sum_b^{-1} (x-\mu)}{\sqrt{(s-\mu)^T \sum_b^{-1} (s-\mu)} \sqrt{(x-\mu)^T \sum_b^{-1} (x-\mu)}} = cos(\theta)$$

where $\mu$ and $\sum_b$ are the mean vector and covariance matrix of the background noise

(target absent), and $\theta$ is the angle between test point $x$ and target signature $s$.

### 5.6.2 ACE Estimator on sEMG Data

We tested the ACE detector on the sEMG dataset used in [55] and in the previous section. The target signatures were calculated as described in Sec. 5.5. For calculating the background noise statistics, we used sEMG signals captured when the gestures were not being performed. The class assignment was made based on the highest ACE detector value (absolute value). Table 5.6 shows the results for 3, 4 and 5 gestures. The results shown are averages over all test subjects.

Table 5.6: Confusion matrices for 3, 4 and 5 gestures. The values are average percentages over a 5-fold cross validation, over all test subjects.

| ACE | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 58.5 | 19.5 | 22.0 |
| | 2 | 44.1 | 33.9 | 22.0 |
| gesture | 3 | 44.9 | 21.3 | 33.8 |
| Correct classification: 42.1% | | | | |

| ACE | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| True | 1 | 42.0 | 28.7 | 19.8 | 9.5 |
| | 2 | 36.4 | 42.0 | 12.8 | 8.8 |
| gesture | 3 | 39.3 | 27.6 | 20.1 | 13.1 |
| | 4 | 35.8 | 27.3 | 12.7 | 24.2 |
| Correct classification: 32.1% | | | | | |

| ACE | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| True | 1 | 36.8 | 20.0 | 13.1 | 10.2 | 19.9 |
| | 2 | 27.1 | 31.5 | 10.9 | 10.2 | 20.3 |
| | 3 | 29.7 | 17.8 | 17.4 | 11.6 | 23.4 |
| gesture | 4 | 27.6 | 20.3 | 11.3 | 18.6 | 22.1 |
| | 5 | 26.4 | 20.6 | 15.7 | 11.4 | 25.9 |
| Correct classification: 26.0% | | | | | | |

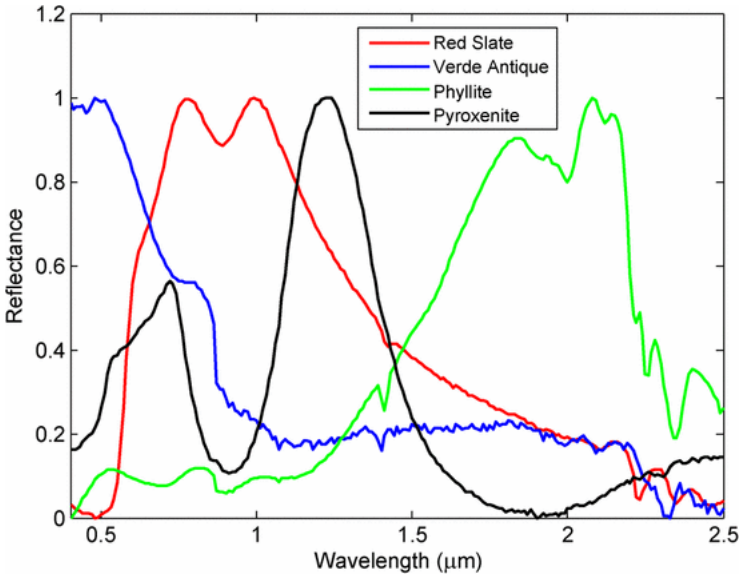



Figure 5.22: Signatures from ASTER library.

The performance obtained with the ACE detector is very sub-par. It is important to mention that background noise is assumed to be Gaussian. Good estimates of the background statistics rely on enough data samples. An important limitation of typical sEMG-based applications is the reduced number of training signals – here we had $\sim 100$ samples per gesture, per subject. We attribute the poor performance of ACE to the inadequate number of signals available to calculate the necessary statistics.

### 5.6.3 GUSSS on Synthetic Hyperspectral Data

We tested the GUSSS method on simulated data generated from 4 spectra selected from the ASTER spectral library [134]. The spectra consists of 211 bands and the wavelengths range from 0.4 to 2.5 $\mu m$ [131].

Figure 5.22 shows the signatures used to create the artificial mixtures. The dataset consists of 5000 mixture signals (downloaded from [135]). We used the given signatures to calculate GUSSS ratios, in order to determine the main components of the mixtures. Table 5.7 shows the classification results.

Table 5.7: Confusion matrices, artificial mixtures.

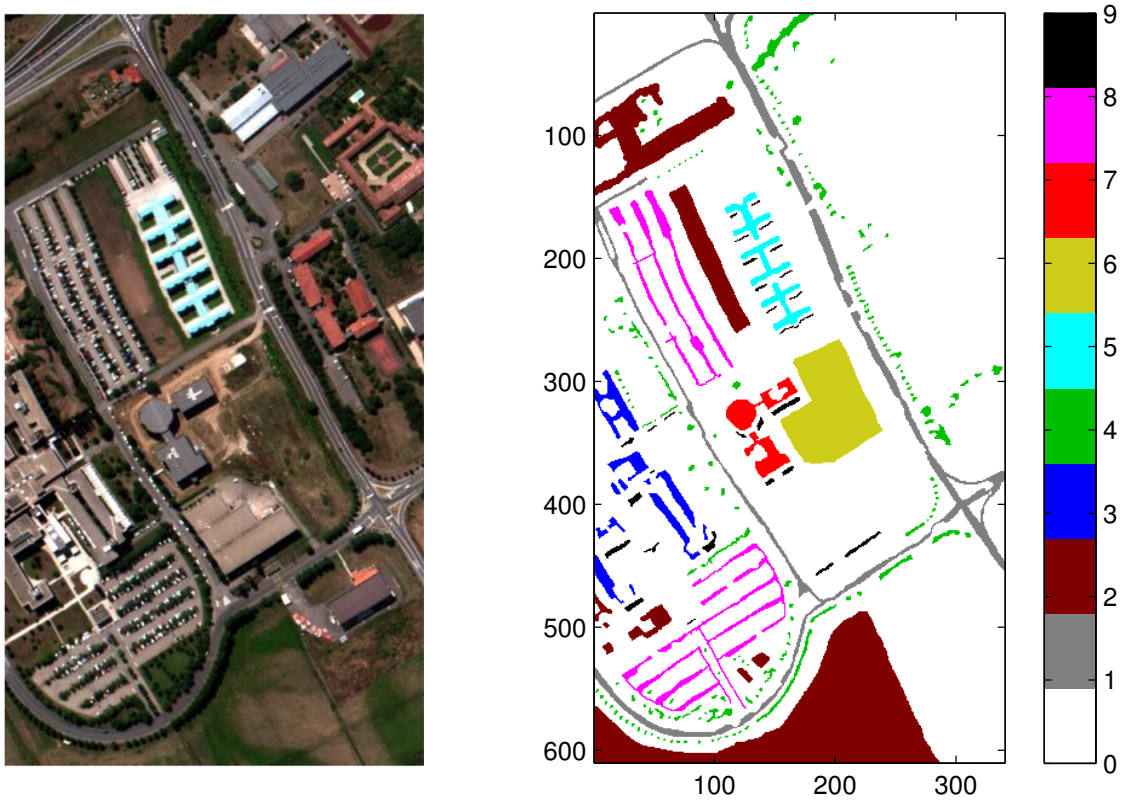
| GUSSS Ratio | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | Red S. | Verde A. | Phyllite | Pyroxenite |
| True gesture | Red S. | 86.1 | 0.0 | 5.9 | 8.0 |
| | Verde A. | 1.9 | 93.6 | 3.8 | 0.6 |
| | Phyllite | 3.4 | 5.6 | 90.5 | 0.5 |
| | Pyroxenite | 14.7 | 5.3 | 1.0 | 79.0 |
| Correct classification: 87.6% | | | | | |

The results show that, for the most part, the GUSSS ratio approach was able to detect the main contributions of the different signatures in the artificially created mixtures. Next we explore our method on real data.

### 5.6.4 GUSSS on Real Hyperspectral Data

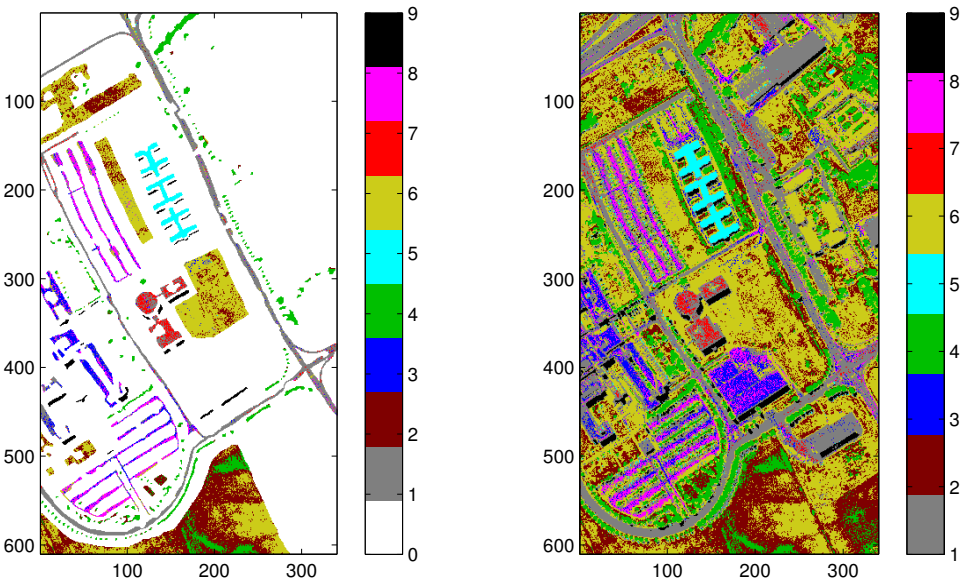
We also tested the method on a subset of the ROSIS Pavia University data set [131,136], which was collected over an urban area of Pavia, Italy. Figure 5.23 (a) shows the RGB image of the scene. The image consists of $610 \times 340$ pixels, each one represented by a signal consisting of 103 bands in the $430-850\ nm$ wavelength range. Out of the 207400 total pixels, a subset of 42776 pixels ($\sim 20\%$) of the dataset are labeled (there are 9 different classes). Figure 5.23 (b) shows those pixels color-labeled. The remaining, non-labeled pixels are shown in white. The classes are the following: 1) Asphalt; 2) Meadows; 3) Gravel; 4) Trees; 5) Painted metal sheets; 6) Bare soil; 7) Bitumen; 8) Self-blocking bricks; 9) Shadows.

We ran our GUSSS method to classify both the labeled and non-labeled pixels. For that, we used $T = 50$ training signals per class. It is relevant to mention that we used only $\sim 1\%$ of the labeled pixels ($\sim 0.2\%$ of the total pixels) for training. Figure 5.24 (a) shows the classification results obtained for the labeled pixels. The same color code as in Figure 5.23 (b) is used. Table 5.8 shows the corresponding confusion matrix and correct classification percentage. Figure 5.24 (b) shows the classification results for all the pixels.




(a) (b)

Figure 5.23: Pavia University data set. (a) RGB image; (b) Labeled pixels (colors) and non-labeled pixels (white). Legend: 0) No label; 1) Asphalt; 2) Meadows; 3) Gravel; 4) Trees; 5) Painted metal sheets; 6) Bare soil; 7) Bitumen; 8) Self-blocking bricks; 9) Shadows.




(a) (b)

Figure 5.24: Classification maps. Pavia dataset. (a) Labeled pixels only; (b) All pixels.

Table 5.8: Confusion matrix (percentage values). Pavia dataset, labeled pixels. Legend: 0) No label; 1) Asphalt; 2) Meadows; 3) Gravel; 4) Trees; 5) Painted metal sheets; 6) Bare soil; 7) Bitumen; 8) Self-blocking bricks; 9) Shadows.

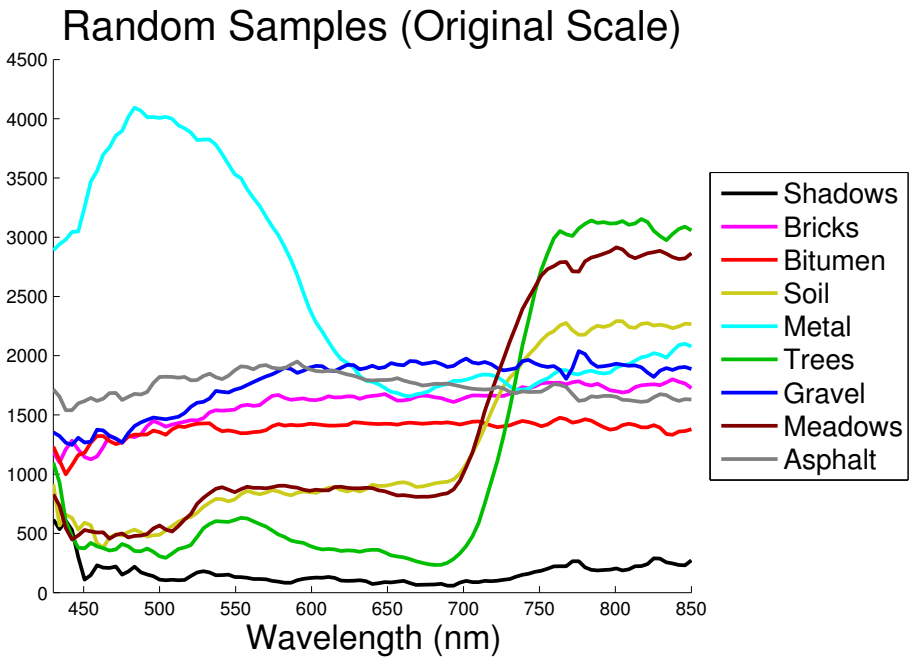
| GUSSS | | Assigned label | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| T | 1 | 89.2 | 0.2 | 2.8 | 0.0 | 0.2 | 2.5 | 3.2 | 1.9 | 0.0 |
| r | 2 | 0.3 | 47.0 | 0.0 | 7.9 | 0.0 | 44.8 | 0.0 | 0.0 | 0.0 |
| u | 3 | 21.0 | 0.0 | 57.6 | 0.0 | 0.0 | 3.0 | 0.4 | 18.0 | 0.0 |
| e | 4 | 0.0 | 6.4 | 0.0 | 91.4 | 0.0 | 2.1 | 0.0 | 0.0 | 0.0 |
| L | 5 | 0.2 | 0.0 | 0.0 | 0.0 | 99.1 | 0.7 | 0.0 | 0.0 | 0.0 |
| a | 6 | 0.9 | 14.2 | 1.0 | 0.1 | 0.1 | 83.8 | 0.0 | 0.0 | 0.0 |
| b | 7 | 49.0 | 0.0 | 1.0 | 0.0 | 0.0 | 0.2 | 49.3 | 0.5 | 0.0 |
| e | 8 | 10.9 | 0.1 | 23.1 | 0.0 | 0.0 | 7.0 | 0.1 | 58.8 | 0.0 |
| l | 9 | 2.1 | 0.0 | 0.0 | 0.0 | 0.5 | 0.0 | 0.0 | 0.0 | 97.4 |
| Correct classification: 65.4% | | | | | | | | | | |

The confusion matrix above shows that asphalt, trees, painted metal sheets, bare

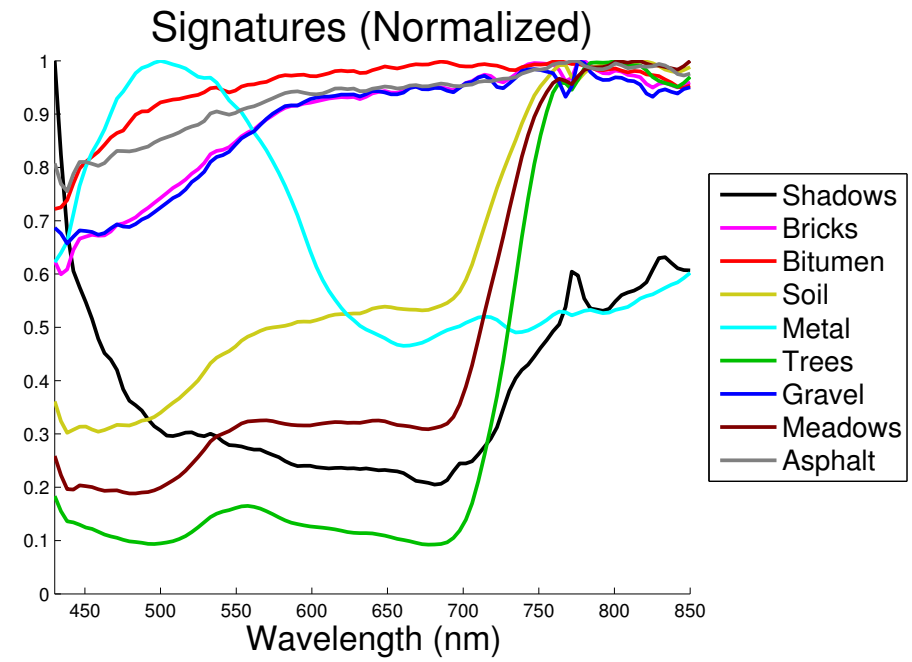
soil and shadows are very distinguishable classes, whereas the meadows are very confused with bare soil, gravel is somewhat confused with asphalt and bricks, bitumen is very confused with asphalt, and bricks are somewhat confused with gravel and asphalt. Note that the confused materials are very similar, indeed. For example, bitumen and asphalt are mostly interchangeable terms.

To verify the similarities from the point of view of the hyperspectral signals, we plotted random samples from the different classes, as well as the learned signatures. Figure 5.25 shows the signals. It can be seen that those confused classes show very similar spectra. To quantify the seemingly high similarity, we calculated MI values between each pair of signatures. Figure 5.26 shows the pairwise MI values. As a reference, the MI value between two identical signals consisting of 103 sample points (same as this dataset) is 2.77. On the other hand, the MI value between two independent signals should be zero. In practice, though, the MI values calculated for signals of finite length is not zero. To get a better idea, we computed the MI value between 103 sample points drawn from a uniform distribution, and 103 sample points drawn from a normal distribution. We ran 10000 cases, and the average MI value that we obtained was 0.0073, with a standard deviation of 0.0140.

Note that most of the MI values between the different signatures are relatively high. In particular, the values are high between classes meadows and soil; gravel and bricks; bitumen and asphalt; bricks and asphalt; and bricks and gravel – classes that were confused in the classification. On the other hand, note that class metal sheets shows the highest overall classification. Accordingly, the lowest MI values are between class metal sheets and the other classes.




(a)




(b)

Figure 5.25: Pavia dataset. (a) Sample signals (original range); (b) Learned Signatures (normalized).

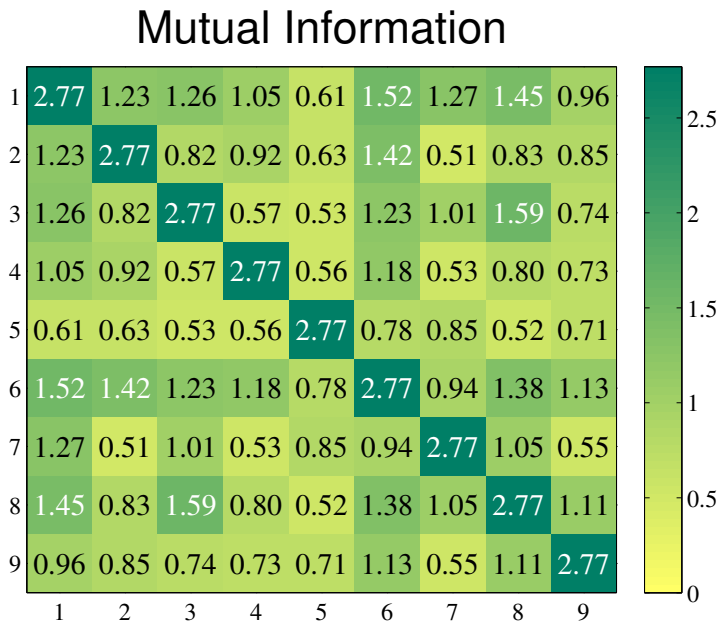



Figure 5.26: Pavia dataset. Mutual Information values between pairs of signatures.

It is worth noting the good classification of the non-labeled pixels, as can be seen in Figure 5.24 (b). The assignments seem to match very well the RGB image shown in Figure 5.23 (a).

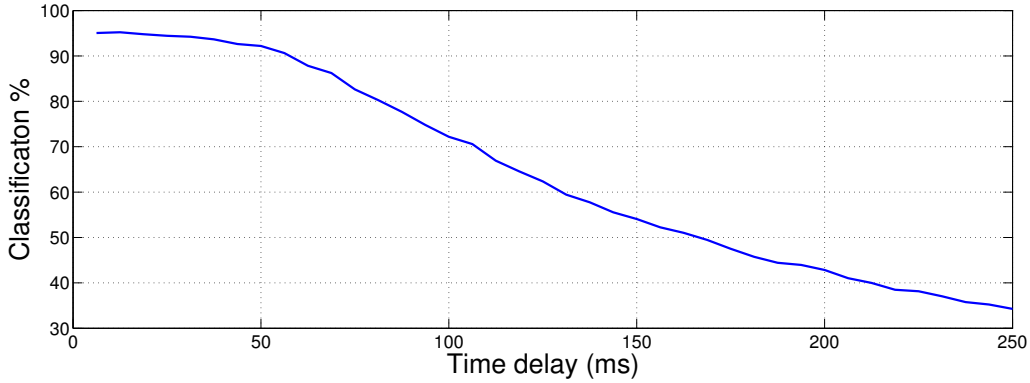
Finally, it should be noted the good performance achieved even with the reduced number of training signals used. This contrasts with the poor performance of the ACE detector on the sEMG dataset.

## 5.7 Tolerance to Time Delays

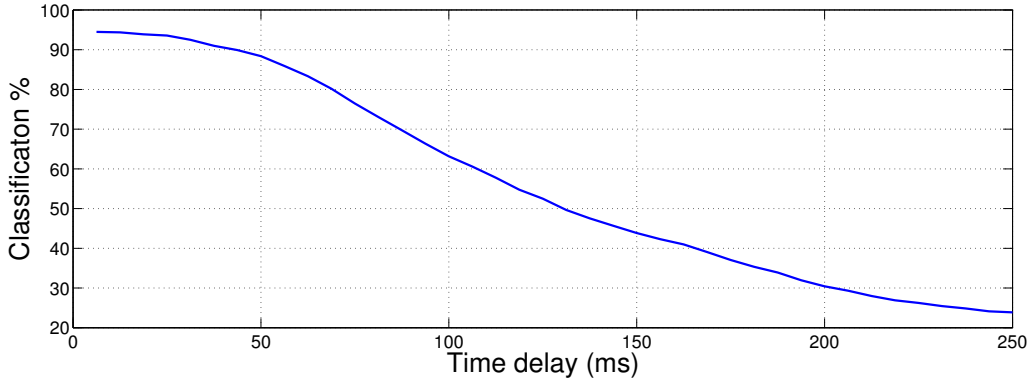
In a typical sEMG application, the first challenge is to detect a gesture from a continuous stream of data samples. That is, before performing any classification task, an sEMG signal must first be detected. Several approaches have been proposed for sEMG signal detection [137]. In every case, because of noise, there is always the chance of capturing shifted (delayed) versions of the corresponding signatures.

We tested classification as a function of how delayed the testing signals are, in order to determine how sensitive is our method to this factor. For that, we took the sEMG

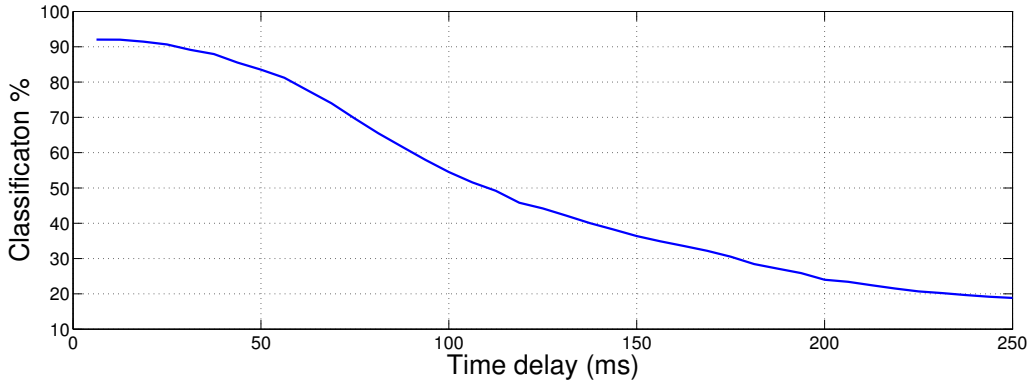
dataset, which consists of $500ms$ long signals, collected at a sampling rate of 4 kHz. As usual, we divided the set into training and testing signals. We left the training set unchanged, but we shifted the testing signals by various $\Delta t$, ranging from $5ms$ to $250ms$. For each case, for all test subject, we ran the GUSSS method and calculated classification accuracy statistics. We ran the tests for 3, 4, and 5 gestures, as in previous sections. Figure 5.27 shows the average results over 5-fold cross-validation, over all test subjects.




(a) 3 gestures




(b) 4 gestures




(c) 5 gestures

Figure 5.27: Classification performance vs. time delay of testing signals. Averages over 5-fold cross validation, over all test subjects.

As expected, the classification performance decreases as the time delay increases. However, the results are still very reasonable. Note that it takes delays of around

$50 - 70$ $ms$ for the classification accuracy to drop 10%. Such a delay is a lot greater than any delay that has been observed in our systems [53, 56].

## 5.8 Non-Linearity

An important assumption that has been made throughout this research is a linear mixing model. One final set of tests that we ran here seek to investigate non-linear ways of mixing source signals. The goal of these tests is to examine some simple cases, the get an idea of how the GUSSS method is affected by removing the linearity condition. The tests are by no means thorough. Future work should include a more complete examination and should investigate specific non-linear models used in different areas of interest.

### 5.8.1 Polynomial Mixture Models

For the following experiments, mixtures were created using more general polynomial models. A mixture of order $q$ is given by

$$x = \sum_{i \neq p} c_i s_i^q + c_p s_p^q + n$$

where $q$ is the power to which the signatures are raised. Given a particular $c_p$ value, the remaining coefficients $c_i$, $i \neq p$, are randomly chosen so that $\sum c_i = 1$. As before, $c_p$ represents a percentage of $s_p^q$ in the mixture. In essence, the mixtures are linear combinations of the $q^{th}$ power of the signatures, with some additive noise levels $n$ ($\sigma$).

### 5.8.2 Experiments and results

As in previous sections, we ran some tests using the sound data set. For each signature $s_p$, sets of 200 polynomial mixtures were created: 100 including $s_p^q$, and 100 without $s_p^q$. GUSSS ratios were calculated between the original signature $s_p$ and each mixture. Averages were calculated. This was repeated for $c_p = 0.9$ and $c_p = 0.5$ values, for different noise levels, and for $q = 1, 2, 3, 4, 5$. Case $q = 1$ corresponds to the linear

model that has been used thus far. It was included here for ease of comparison with the higher order polynomial cases.

The results are shown in Figures 5.28 and 5.29. For example, Figure 5.28 shows bar diagrams representing the average inverse GUSSS ratios for mixtures created using $c_p = 0.9$ – in the "Present" ($P$) cases. The "Not Present" bars ($NP$) represent the average values obtained for the mixtures not containing the target signatures. Each row of diagrams corresponds to different noise levels used. Each column of diagrams corresponds to a different polynomial order. The results are averages over all eight sound signatures. Inverse Ratios are used here instead of the actual ratios for an easier comparison between previous sections' results. Note that for class separability, there is no difference between using ratios or inverse ratio values.

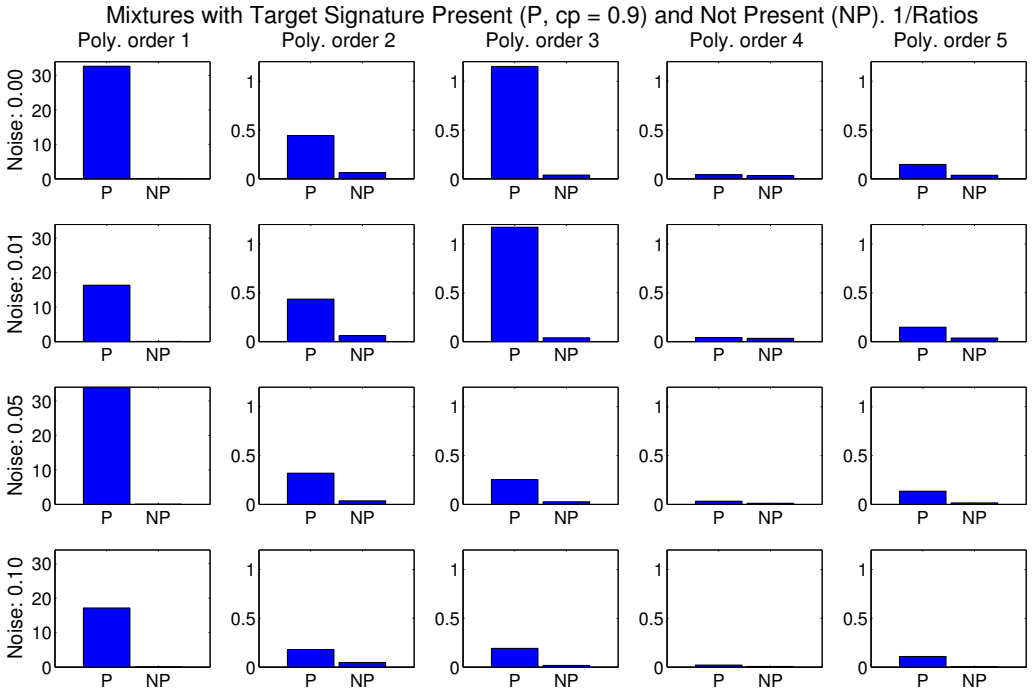



Figure 5.28: Average inverse GUSSS ratio values. $c_p = 0.9$.

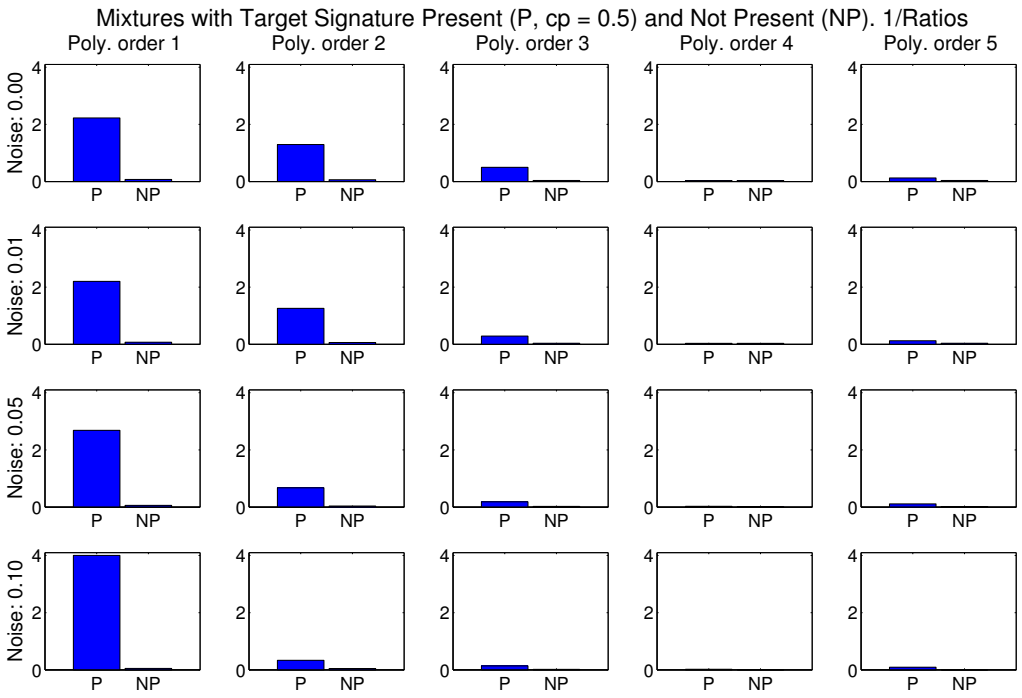



Figure 5.29: Average inverse GUSSS ratio values. $c_p = 0.5$.

The results above show that the discriminative power of the GUSSS ratio is, indeed, affected by non-linear mixing of sources. Nevertheless, there is still separation between classes "Present" and "Not Present" in several cases above. Figure 5.28 shows the case $c_p = 0.9$. As expected, case $q = 1$ shows very high separation between classes, even for high noise levels. Cases $q = 2$ and especially $q = 3$ show good separability, as well. Note, however, cases $q = 5$ and $q = 4$. Separability is very little in the former, and there is practically no difference between class $P$ and $NP$ in the latter.

Similar tendencies can be observed for $c_p = 0.5$. As expected, the average values for the $P$ class are smaller than for $c_p = 0.9$. But the separability is still very high for $q = 1$, and relatively high for $q = 2$, at least for the lower noise levels. However, there is virtually no separation of classes for $q = 4$ and $q = 5$.

It is interesting to note that even polynomial orders lead to worse results than odd orders. While the tendency is for the separability of classes to decrease with higher orders, $q = 3$ is better than $q = 2$, and $q = 5$ is better than $q = 4$, at least for $c_p = 0.9$.

For $c_p = 0.5$, the worst results are obtained with $q = 4$, although $q = 2$ seems to be better than $q = 3$. This makes sense if we consider the mutual information between the original signatures $s_p$ and their powers $s_p^q$. For odd powers, the average MI value for the eight sound signatures used is 6.9, whereas for even powers, the average MI value is 5.2. Added noise and different contributions of non-target signatures would cause variations in terms of the dependence between the mixtures and the target signatures, and thus, in the GUSSS ratios. But the fact that an even power of a source distribution is less dependent to the original distribution, compared to an odd power, explains the results obtained for these polynomial models.

# Chapter 6

# Discussion and Conclusions

This work addressed the question of whether signal processing can offer insights into recognizing multiple patterns present in single observations. We presented a method which combines pattern separation and recognition through a classical signal processing technique – Source (Signal) Separation. Our method, called Guided Under-determined Source Signal Separation (GUSSS), relies on Independent Component Analysis and it is useful when there is a need of separating individual components from mixtures captured by a single sensor. GUSSS differs from Under-determined Blind Source Signal Separation in that we use prior knowledge of particular source signals (signatures) that might be present within the sensed mixture signals. By incorporating that prior information we go from an under-determined problem, where we have fewer sensed signals than components, to a well determined problem, where we have two equations and two components that we want to estimate. This allows the use of traditional ICA algorithms to estimate the independent components and the mixing coefficients. Using those coefficients we introduced the GUSSS ratio, which can be used to determine if a particular signature is present or not in a sensed signal, or as a feature in a classification problem.

Early experiments on artificial sound mixtures yielded very promising results. We then developed frameworks for applying our methods on very different areas: material detection using THz signatures; root phenotyping using THz imaging; assistive technology and voice disorders recognition using sEMG signals. The results that we have

obtained in those areas are also very satisfying.

To test our framework for detecting the presence of particular chemicals or materials in a mixture, we used THz signatures of various materials obtained from public databases. Our method yielded very good results for detecting a variety of such materials under various assumptions on the percentage of the target material present in the scene. For those experiments, we used very simple classification schemes and only a single feature – the GUSSS ratio. Our goal was to demonstrate the good performance of the method when it relies solely on the ratio. Unlike other methods that emphasize the detection of individual compounds, the results obtained demonstrated that the proposed approach allows the detection of materials within mixtures. The results also support the claim that GUSSS presents a very powerful tool for the separation and identification of patterns in signals. Our proposed approach could lead to the development of systems capable of detecting hazardous materials, such as RDX, TNT, drugs, etc., enhancing and simplifying law enforcement, anti-terrorism security, etc.

Also in the area of THz technology, the results presented in this work demonstrate the potential of THz imaging to detect and identify the different objects buried in sand and potting soil. We believe that our framework has the potential of revolutionizing root phenotyping in situ, and thus genetic improvement on the basis of root characteristics.

For the area of assistive technology we focused on classifying sEMG signals, which can be used for controlling devices such as power wheelchairs. In order to use GUSSS, we proposed a framework to detect and classify different "gestures" (e.g. hand or eyebrow movements, etc.), which can be related to commands for a wheelchair. Our method allowed using a single sensor and a simple classification scheme. In search of improving classification performance and increasing the number of recognizable gestures, we showed how to combine the GUSSS ratio with other features, and we developed a hierarchical method – HiGUSSS. The performance of our enhanced methods were considerably better than our original approach, and comparable to other methods in the literature, which rely on more sensors and complex classification schemes.

Further usefulness of sEMG signals was demonstrated in the area of otolaryngology.

Specifically, we showed the potential of applying our proposed frameworks to the early detection of vocal disorders through the detection of changing and/or emerging patterns in those sEMG signals.

Even though our experimental results had been very satisfying, there were still research questions related to our methods that needed to be answered. The first question addressed the robustness of the method. Many experiments were run to explore the effects of the different parameters of the GUSSS method, including initialization and choice of the source separation algorithm used. Further tests were made to study the stability and resilience to noise of the method.

Another important question addressed had to do with the assumptions made, most importantly, the statistical dependence between the components or signatures that may constitute a mixture. We explored how the GUSSS method is affected by various degrees of dependence. As expected, a lower discriminative power of the GUSSS ratio correlates with higher dependence between the different signatures that may be considered.

In the areas of assistive technology, otolaryngology, and others, where true signatures may not be available, there is a need of learning the signatures from training data. The initial results were obtained by using a simple mean based signature. We later investigated other different approaches for learning those signatures. We explored clustering based approaches, and we also explored an ICA decomposition based approach inspired by the Bag-of-Words model. The latter approach led to an overall improvement in the classification performance of sEMG signals.

It is always important to compare a proposed method to other methods that have been applied to the same domains. It also makes sense to compare to methods that may have been applied to other domains, but that may make similar assumptions to those made for one's method. In this research, we compared the GUSSS method with commonly used methods such as SVM and others. In the areas of assistive technology, for example, the comparisons are not direct, given that most approaches make use of many sensors – as opposed to our approach. We also compared to a method used in Hyperspectral Image analysis, the Adaptive Coherence/Cosine Estimator (ACE),

which assumes a similar linear model for mixtures. In fact, we did a two-way comparison: we applied GUSSS to Hyperspectral data, and we applied ACE on sEMG data. The results using GUSSS were very satisfying, whereas the results using ACE were underwhelming, highlighting a disadvantage of many methods with respect to GUSSS: the need for a large amount of training signals.

Further experiments were performed to study the effect of time delays on the GUSSS classification performance. This is relevant in applications such as assistive technology, where the sEMG signals need to be detected on-line. We observed that there is, in fact, a degraded classification performance with long delays. However, the drop is not too important for typical delays observed in real systems.

Finally, we ran some initial tests on mixtures produced using non-linear models. The goal was to get a sense of how our method is affected when removing the linearity condition. Even though the results are very preliminary, we observed that the GUSSS method may still able to recognize the presence or not of signatures. Further comments on non-linear models will be made in Chapter 7. However, it is worth highlighting the appropriateness and usefulness of linear models, particularly in the areas tackled in this research.

# Chapter 7

# Future Work

The linear mixing model is a simple, yet powerful model that has been used in many areas, such as those described in this thesis. However, non-linear mixing models are able to represent a wider range of scenarios and applications. For example, in the area of hyperspectral image analysis, a linear model is appropriate for a well defined proportional "checkerboard" mixture of materials, and the assumption that incident radiation bounces only once at the surface. However, a more realistic assumption is that the materials are randomly distributed, forming a homogeneous mixture, and that the radiation reflects multiple times, with multiple substances. In that case, the spectrum of reflected radiation cannot be properly model in a linear way, so a non-linear model would be needed [130]. Future work could focus on exploring non-linear mixing models, specific to the application areas, and incorporating those into the GUSSS method.

Future work could also investigate the more general under-determined case, that is, when the number of sensors is still smaller than the number of independent sources ($M < N$), but not necessarily equal to one. In the work described in Section 4.3 we actually used 4 sensors. However, the strategy then was to replicate the single-channel HiGUSSS method for all channels and then average the confidence vectors to input the Multi-class classifier. Instead, the research would aim at a generalization of the GUSSS method for any case.

Along the lines of the applications using sEMG signals, the method could also be applied to other types of bio-electrical signals, such as Electroencephalographic (EEG)

or Electrocardiographic (EKG) signals. EEG signals, for instance, are widely used for clinical assessment and neurological studies and research. Collecting those signals requires a lot of sensors. The use of the GUSSS method may allow for the reduction of the number of sensors, making the process of data collection easier for the patients, doctors and researchers.

From the application point of view, future work should not be limited to the areas tackled in this work. The proposed method could be applied, for example, to image processing and computer vision. GUSSS could be used to detect specific patterns in an image or could be added to background subtraction algorithms. In essence, GUSSS could potentially be used in many applications where the concept of "signature" would fit.

# Bibliography

[1] L. A. Rivera, "Pattern recognition of mixed signals using guided underdetermined source signal separation," Master's thesis, University Of Missouri, Columbia, MO, USA, Dec. 2011.

[2] S. Haykin and Z. Chen, "The cocktail party problem," *Neural Computation*, vol. 17, pp. 1875–1902, Sep. 2005.

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

[4] A. Hyvärinen, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, pp. 94–128, 1999.

[5] K. Hild, D. Pinto, D. Erdogmus, and J. Principe, "Convolutive blind source separation by minimizing mutual information between segments of signals," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 52, no. 10, pp. 2188–2196, Oct 2005.

[6] M. Jafari, W. Wang, J. Chambers, T. Hoya, and A. Cichocki, "Sequential blind source separation based exclusively on second-order statistics developed for a class of periodic signals," *Signal Processing, IEEE Transactions on*, vol. 54, no. 3, pp. 1028–1040, March 2006.

[7] D. N. Levin, "Performing nonlinear blind source separation with signal invariants," *Signal Processing, IEEE Transactions on*, vol. 58, no. 4, pp. 2131–2140, April 2010.

[8] C. Choi, W. Chang, and S. Lee, "Blind source separation of speech and music signals using harmonic frequency dependent independent vector analysis," *Electronics Letters*, vol. 48, no. 2, pp. 124–125, January 2012.

[9] G. Fabrizio and A. Farina, "Blind source separation with the generalised estimation of multipath signals algorithm," *Radar, Sonar Navigation, IET*, vol. 8, no. 9, pp. 1255–1266, 2014.

[10] J.-F. Cardoso, "Blind signal separation: statistical principles," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009–2025, Oct 1998.

[11] L. Tomazeli Duarte, S. Moussaoui, and C. Jutten, "Source separation in chemical analysis : Recent achievements and perspectives," *Signal Processing Magazine, IEEE*, vol. 31, no. 3, pp. 135–146, May 2014.

[12] E. Oja, K. Kiviluoto, and S. Malaroiu, "Independent component analysis for financial time series," in *The IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Oct. 2000, pp. 111–116.

[13] T. Jung, S. Makeig, M. J. McKeown, A. J. Bell, T. Lee, and T. J. Sejnowski, "Imaging brain dynamics using independent component analysis," in *Proceedings of the IEEE*, vol. 89, no. 7, Jul. 2002, pp. 1107–1122.

[14] Y. Li, K. Ho, and M. Popescu, "Efficient source separation algorithms for acoustic fall detection using a microsoft kinect," *Biomedical Engineering, IEEE Transactions on*, vol. 61, no. 3, pp. 745–755, March 2014.

[15] M. Li, Y. Liu, F. Chen, and D. Hu, "Including signal intensity increases the performance of blind source separation on brain imaging data," *Medical Imaging, IEEE Transactions on*, vol. 34, no. 2, pp. 551–563, Feb 2015.

[16] A. Ozerov, E. Vincent, and F. Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 4, pp. 1118–1133, May 2012.

[17] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *Signal Processing, IEEE Transactions on*, vol. 45, no. 2, pp. 434–444, Feb 1997.

[18] P. Georgiev, F. Theis, and A. Cichocki, "Sparse component analysis and blind source separation of underdetermined mixtures," *Neural Networks, IEEE Transactions on*, vol. 16, no. 4, pp. 992–996, July 2005.

[19] Y. Li, S. Amari, A. Cichocki, D. W. C. Ho, and S. Xie, "Underdetermined blind source separation based on sparse representation," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 423–437, Feb. 2006.

[20] S. Araki, H. Sawada, R. Mukai, and S. Makino, "Underdetermined blind sparse source separation for arbitrarily arranged multiple sensors," *Signal Processing*, vol. 87, pp. 1833–1847, Mar. 2007.

[21] W. Yang, J. Pu, F. Zhang, H. Zhang, and Y. Lin, "Underdetermined blind source separation based on linear membership function," in *Biomedical Engineering and Computer Science (ICBECS), 2010 International Conference on*, April 2010, pp. 1–6.

[22] A. Liutkus, R. Badeau, and G. Richard, "Gaussian processes for underdetermined source separation," *Signal Processing, IEEE Transactions on*, vol. 59, no. 7, pp. 3155–3167, July 2011.

[23] S. Xie, L. Yang, J.-M. Yang, G. Zhou, and Y. Xiang, "Time-frequency approach to underdetermined blind source separation," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 23, no. 2, pp. 306–316, Feb 2012.

[24] B. Liu, V. Reju, and A. Khong, "A linear source recovery method for underdetermined mixtures of uncorrelated ar-model signals without sparseness," *Signal Processing, IEEE Transactions on*, vol. 62, no. 19, pp. 4947–4958, Oct 2014.

[25] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," *Signal Processing*, vol. 81,

no. 11, pp. 2353 – 2362, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168401001207

[26] L. De Lathauwer, J. Castaing, and J. Cardoso, "Fourth-order cumulant-based blind identification of underdetermined mixtures," *Signal Processing, IEEE Transactions on*, vol. 55, no. 6, pp. 2965–2973, June 2007.

[27] A. Karfoul, L. Albera, and G. Birot, "Blind underdetermined mixture identification by joint canonical decomposition of ho cumulants," *Signal Processing, IEEE Transactions on*, vol. 58, no. 2, pp. 638–649, Feb 2010.

[28] L. Benaroya, F. Bimbot, and R. Gribonval, "Audio source separation with a single sensor," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 1, pp. 191–199, Jan 2006.

[29] M. Davies and C. James, "Source separation using single channel {ICA}," *Signal Processing*, vol. 87, no. 8, pp. 1819 – 1832, 2007, independent Component Analysis and Blind Source Separation. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168407000151

[30] B. Mijović, M. De Vos, I. Gligorijević, J. Taelman, and S. Van Huffel, "Source separation from single-channel recordings by combining empirical-mode decomposition and independent component analysis," *Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 9, pp. 2188–2196, Sept 2010.

[31] Y. Guo, G. Naik, and H. Nguyen, "Single channel blind source separation based local mean decomposition for biomedical applications," in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, July 2013, pp. 6812–6815.

[32] Y. Guangying, "Study of myoelectric prostheses hand based on independent component analysis and fuzzy controller," in *Eighth International Conference on Electronic Measurement and Instruments*, Aug. 2007, pp. 1–174–1–178.

[33] P. Comon and C. Jutten, *Handbook of Blind Source Separation - Independent Component Analysis and Applications*. Academic press, 2010.

[34] K. P. Balanda and H. L. MacGillivray, "Kurtosis a critical review," *The American Statistician*, vol. 42, no. 2, pp. 111–119, May 1988.

[35] D. E. Lake, "Renyi entropy measures of heart rate gaussianity," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 1, pp. 21–27, Jan. 2006.

[36] G. Szkely, M. Rizzo, and N. Bakirov, "Measuring and testing dependence by correlation of distances," *The Annals of Statistics*, vol. 35, no. 6, pp. 2769–2794, 2007.

[37] A. Gretton, R. Herbrich, A. Smola, O. Bousquet, and B. Schölkopf, "Kernel methods for measuring independence," *Journal of Machine Learning Research*, vol. 6, pp. 2075–2129, 2005.

[38] F. R. Bach and M. I. Jordan, "Kernel independent component analysis," *J. Mach. Learn. Res.*, vol. 3, pp. 1–48, mar 2003. [Online]. Available: http://dx.doi.org/10.1162/153244303768966085

[39] J. Cardoso and A. Souloumiac, "Blind beamforming for non-gaussian signals," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, no. 6, pp. 362–370, Dec 1993.

[40] J.-F. Cardoso, "High-order contrasts for independent component analysis," *Neural Comput.*, vol. 11, no. 1, pp. 157–192, jan 1999. [Online]. Available: http://dx.doi.org/10.1162/089976699300016863

[41] A. Hyvärinen and E. Oja, "A fast fixed-point algorithm for independent component analysis," *Neural Comput.*, vol. 9, no. 7, pp. 1483–1492, oct 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.7.1483

[42] ——, "FastICA," neural Networks Research Centre, Helsinki University of Technology, Finland. [Online]. Available: http://research.ics.tkk.fi/ica/fastica/

[43] E. G. Learned-Miller and J. W. F. III, "Ica using spacings estimates of entropy," *J. Mach. Learn. Res.*, vol. 4, pp. 1271–1295, dec 2003. [Online]. Available: http://dl.acm.org/citation.cfm?id=945365.964306

[44] H. Stögbauer, A. Kraskov, S. A. Astakhov, and P. Grassberger, "Least-dependent-component analysis based on mutual information," *Phys. Rev. E*, vol. 70, p. 066123, Dec 2004. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.70.066123

[45] F. R. Bach and M. I. Jordan, "Finding clusters in independent component analysis," in *4th Intl. Symp. on Idependent Component Analysis and Signal Separation (ICA2003)*, 2003, pp. 891–896.

[46] A. Chen, "Fast kernel density independent component analysis," in *Independent Component Analysis and Blind Signal Separation*, ser. Lecture Notes in Computer Science, J. Rosca, D. Erdogmus, J. PrÃncipe, and S. Haykin, Eds. Springer Berlin Heidelberg, 2006, vol. 3889, pp. 24–31.

[47] H. Shen, S. Jegelka, and A. Gretton, "Fast kernel-based independent component analysis," *Signal Processing, IEEE Transactions on*, vol. 57, no. 9, pp. 3498–3511, Sept 2009.

[48] Z. Koldovsky, P. Tichavsky, and E. Oja, "Efficient variant of algorithm fastica for independent component analysis attaining the cramér-rao lower bound," *Neural Networks, IEEE Transactions on*, vol. 17, no. 5, pp. 1265–1277, Sept 2006.

[49] L. A. Rivera and G. N. DeSouza, "Detecting terahertz signatures using guided under-determined source signal separation," in *40th International Conference on Infrared, Millimeter, and Terahertz Waves*, Aug. 2015, IRMMW-THz, Hong Kong.

[50] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2nd Ed.* Wiley Interscience, 2001.

[51] K. Fukunaga, *Introduction to statistical pattern recognition.* Academic press, 2013.

[52] L. A. Rivera and G. N. DeSouza, "Recognizing hand movements from a single sEMG sensor using guided under-determined source signal separation," in *12th IEEE International Conference on Rehabilitation Robotics*, Jun. 2011, pp. 450–455, ETH Zurich, Switzerland.

[53] ——, "A power wheelchair controlled using hand gestures, a single sEMG sensor, and guided under-determined source signal separation," in *4th IEEE RAS & EMBS International Conference on Biomedical Robotics*, Jun. 2012, pp. 1535–1540, BioRob2012, Rome, Italy.

[54] ——, *Haptic and Gesture-Based Assistive Technologies for People with Motor Disabilities.* IGI Global, 2013, ch. 1, pp. 1–27.

[55] L. A. Rivera, N. R. Smith, and G. N. DeSouza, "High-accuracy recognition of muscle activation patterns using a hierarchical classifier," in *5th IEEE RAS & EMBS International Conference on Biomedical Robotics*, Aug. 2014, BioRob2014, Sao Paulo, Brazil.

[56] N. R. Smith, L. A. Rivera, M. Dietrich, C. R. Shyu, M. P. Page, and G. N. DeSouza, "Detection of simulated vocal dysfunctions using complex sEMG patterns," *IEEE Journal of Biomedical and Health Informatics*, 2015, accepted for publication.

[57] N. R. Smith, L. A. Rivera, N. Burford, T. Bowman, M. O. El-Shenawee, and G. N. DeSouza, "Towards root phenotyping in situ using THz imaging," in *40th International Conference on Infrared, Millimeter, and Terahertz Waves*, Aug. 2015, IRMMW-THz, Hong Kong.

[58] P. H. Siegel, "Terahertz technology," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 3, pp. 910–928, Mar. 2002.

[59] Y. C. Shen, T. Lo, P. F. Taday, B. E. Cole, W. R. Tribe, and M. C. Kemp, "Detection and identification of explosives using terahertz pulsed spectroscopic imaging," *Applied Physics Letters*, vol. 86, p. 241116, 2005.

[60] B. Ferguson and X. Zhang, "Materials for terahertz science and technology," *Nature Materials*, vol. 1, pp. 26–33, Sep. 2002.

[61] I. Hosako, N. Sekine, M. Patrashin, S. Saito, K. Fukunaga, Y. Kasai, P. Baron, T. Seta, J. Mendrok, S. Ochiai, and H. Yasuda, "At the dawn of a new era in terahertz technology," *Proceedings of the IEEE*, vol. 95, no. 8, pp. 1611–1623, Aug. 2007.

[62] M. C. Kemp, "Detecting hidden objects: Security imaging using millimetre-waves and terahertz," in *IEEE Conference on Advanced Video and Signal Based Surveillance*, Sept. 2007, pp. 7–9.

[63] A. Redo-Sanchez and X. Zhang, "Terahertz science and technology trends," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 14, no. 2, pp. 260–269, Mar./Apr. 2008.

[64] T. Bowman, M. El-Shenawee, and S. G. Sharma, "Terahertz spectroscopy for the characterization of excised human breast tissue," in *2014 IEEE MTT-S International Microwave Symposium*, Jun 2014, tampa Bay, FL.

[65] T. C. Bowman, M. El-Shenawee, and L. K. Campbell, "Terahertz imaging of excised breast tumor tissue on paraffin sections," *IEEE Trans. on Antennas and Propagation*, vol. 63, no. 5, 2015.

[66] M. Kemp, "Millimetre wave and terahertz technology for the detection of concealed threats - a review," in *Proc. SPIE*, vol. 6402, 2006, pp. 1–19.

[67] K. Yamamoto, M. Yamaguchi, F. Miyamaru, M. Tani, M. Hangyo, T. Ikeda, A. Matsushita, K. Koide, M. Tatsuno, and Y. Minami, "Non-invasive inspection of c-4 explosive in mails by terahertz time-domain spectroscopy," *Japanese Journal of Applied Physics*, vol. 43, no. 3B, pp. L414–L417, 2004.

[68] F. Huang, B. Schulkin, H. Altan, J. Federici, D. Gary, R. Barat, D. Zimdars, M. Chen, and D. B. Tanner, "Terahertz study of 1,3,5-trinitro-s-triazine by time-domain and fourier transform infrared spectroscopy," *Applied Physics Letters*, vol. 85, no. 23, pp. 5535–5537, 2004.

[69] J. Chen, H. Zhao, G. J. Bastiaans, and X.-C. Zhang, "Absorption coefficients of selected explosives and related compounds in the range of 0.1-2.8 thz," *Optics Express*, vol. 15, no. 19, pp. 12 060–12 067, Sep. 2007.

[70] K. Kawase, Y. Ogawa, and Y. Watanabe, "Non-destructive terahertz imaging of illicit drugs using spectral fingerprints," *Optics Express*, vol. 11, no. 20, pp. 2549–2554, Oct. 2003.

[71] Y. Sasaki, M. Yamashita, A. Dobroiu, T. Shibuya, C. Otani, and K. Kawase, "Noninvasive detection of concealed powders using terahertz wave scattering," in *Infrared and Millimeter Waves and 13th International Conference on Terahertz Electronics*, vol. 2, Sept. 2005, pp. 648–649.

[72] "Thz-bridge," tera-phtonics Laboratory, RIKEN Sendai. [Online]. Available: http://www.frascati.enea.it/thz-bridge/

[73] "Terahertz database." [Online]. Available: http://thzdb.org

[74] "Thz database web," tera-phtonics Laboratory, RIKEN Sendai. [Online]. Available: http://www.riken.jp/THzdatabase/

[75] H. Zhong, A. Redo-Sanchez, and X.-C. Zhang, "Identification and classification of chemicals using terahertz reflective spectroscopic focal plane imaging system," *Optics Express*, vol. 14, no. 20, pp. 9130–9141, Oct. 2006.

[76] X. Yin, "Pattern recognition and tomographic reconstruction with terahertz signals for applications in biomedical engineering," 2008, ph.D. dissertation, Department of Electrical and Electronic Engineering, University of Adelaide.

[77] T. Nakini and G. DeSouza, "Distortion correction in 3D-modeling of roots for plant phenotyping," in *European Conference on Computer Vision (ECCV) Workshop on Computer Vision for Plant Phenotyping*, Sept. 2014, Zurich, Switzerland.

[78] D. Woolard, E. Brown, M. Pepper, and M. Kemp, "Terahertz frequency sensing and imaging - a time of reckoning future applications?" *Proceedings of the IEEE*, vol. 93, no. 10, pp. 1722–1743, Oct 2005.

[79] M. Aquilano, C. Salatino, and M. C. Carrozza, "Assistive technology: a new approach to evaluation," in *10th IEEE International Conference on Rehabilitation Robotics*, Jun. 2007, pp. 809–819.

[80] D. S. Andreasen and D. L. Gabbert, "Electromyographic switch navigation of power wheelchairs," in *Annual conference of the Rehabilitation Engineering and Assistive Technology Society of North America*, Jun. 2006.

[81] A. O'Brien and R. M. Ruairi, "Survey of assistive technology devices and applications for aging in place," in *Second International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies, and Services*, Sept. 2009, pp. 7–12.

[82] A. Koller-Hodac, D. Leonardo, S. Walpen, and D. Felder, "Knee orthopaedic device, how robotic technology can improve outcome in knee rehabilitation," in *12th IEEE International Conference on Rehabilitation Robotics*, Jun. 2011, pp. 186–191, ETH Zurich, Switzerland.

[83] J. Bae and I. Moon, "Biomechanical assessment of electric lifting chair for persons with disability," in *12th IEEE International Conference on Rehabilitation Robotics*, Jun. 2011, pp. 505–509, ETH Zurich, Switzerland.

[84] A. Basteris, A. D. Luca, V. Sanguineti, C. Solaro, M. Mueller, I. Carpinella, D. Cattaneo, R. Bertoni, and M. Ferrarin, "A tailored exercise of manipulation of virtual tools to treat upper limb impairment in multiple sclerosis," in *12th IEEE International Conference on Rehabilitation Robotics*, Jun. 2011, pp. 1112–1116, ETH Zurich, Switzerland.

[85] Y. Hasegawa and S. Oura, "Exoskeletal meal assistance system (emas ii) for progressive muscle dystrophy patient," in *12th IEEE International Conference on Rehabilitation Robotics*, Jun. 2011, pp. 725–730, ETH Zurich, Switzerland.

[86] V. Jaijongrak, I. Kumazawa, and S. Thiemjarus, "A haptic and auditory assistive user interface: Helping the blinds on their computer operations," in *12th IEEE International Conference on Rehabilitation Robotics*, Jun. 2011, pp. 154–159, ETH Zurich, Switzerland.

[87] J. G. Webster, Ed., *Encyclopedia of Medical Devices and Instrumentation. Electromyography.* John Wiley, 2006.

[88] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, "Techniques of EMG signal analysis: detection, processing, classification and applications," *Biol. Proced. Online*, vol. 1, no. 8, pp. 11–35, March 2006.

[89] M. R. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "Advances in electromyogram signal classification to improve the quality of life for the disabled and aged people," *Journal of Computer Science*, vol. 7, no. 6, pp. 706–715, 2010.

[90] C. S. L. Tsui, P. Jia, J. Q. Gan, H. Hu, and K. Yuan, "EMG based hands free wheelchair control with eog attention shift detection," in *Proceedings of the 2007 IEEE International Conference on Robotics and Biomimetics*, Dec. 2007, pp. 1266–1271.

[91] K. Choi, M. Sato, and Y. Koike, "A new, human-centered wheelchair system controlled by the EMG signal," in *International Joint Conference on Neural Networks*, Jul. 2006, pp. 4664–4671.

[92] Y. Wang, X. Zhang, J. Zhao, and C. He, "Pattern recognition of electromyography applied to exoskeleton robot," in *3rd International Congress on Image and Signal Processing (CISP)*, Oct. 2010, pp. 3802–3805.

[93] M. Khezri and M. Jahed, "Real-time intelligent pattern recognition algorithm for surface EMG signals," *BioMedical Engineering OnLine*, vol. 6, 2007.

[94] ——, "A novel approach to recognize hand movements via sEMG patterns," in *29th Annual International Conference of the IEEE EMBS*, Aug. 2007, pp. 4907–4910.

[95] G. Shuman, "Using forearm electromyograms to classify hand gestures," in *IEEE International conference on Bioinformatics and Biomedicine*, 2009, pp. 261–264.

[96] J. Han, Z. Zenn, D. Kim, H. Lee, and J. Kim, "Human-machine interface for wheelchair control with EMG and its evaluation," in *Proceedings of the 25th Annual International Conference of the IEEE EMBS*, Sep. 2003, pp. 1602–1605.

[97] N. Roy, R. Merrill, S. Thibeault, R. Parsa, S. Gray, and E. Smith, "Prevalence of voice disorders in teachers and the general population," *J. Speech. Lang. Hear. Res*, vol. 47, pp. 281–293, 2004.

[98] K. Verdolini, C. Rosen, and R. Branski, *Classification Manual for Voice Disorders*. Lawrence Erlbaum Associates, 2006.

[99] S. Joshi, A. Wexler, C. Perez-Maldonado, and S. Vernon, "Brain-muscle-computer interface using a single surface electromyographic signal: Initial results," in *Neural Engineering (NER), 2011 5th International IEEE/EMBS Conference on*, April 2011, pp. 342–347.

[100] K. Singh, R. Bhatia, and H. Ryait, "Precise and accurate multifunctional prosthesis control based on fuzzy logic techniques," in *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, June 2011, pp. 188–193.

[101] C. Stepp, J. Heaton, M. Braden, T. Stadelman-Cohen, M. Jetté, and R. Hillman, "Comparison of neck tension palpation rating systems with surface electromyographic and acoustic measures in vocal hyperfunction," *Journal of Voice*, vol. 25, 2011.

[102] C. Stepp, "Surface electromyography for speech and swallowing systems: measurement, analysis, and interpretation," *J. Speech. Lang. Hear. Res*, vol. 55, pp. 1232–1246, 2012.

[103] R. Hillman, J. Heaton, A. Masaki, S. Zeitels, and H. Cheyne, "Ambulatory monitoring of disordered voices," *Ann Otol Rhinol Laryngol*, vol. 115, no. 11, pp. 795–801, 2006.

[104] P. Popolo, J. Švec, and I. Titze, "Adaptation of a pocket pc for use as a wearable voice dosimeter," *J. Speech. Lang. Hear. Res*, vol. 48, no. 4, pp. 780–791, 2005.

[105] J. H. Van Stan, J. Gustafsson, E. Schalling, and R. E. Hillman, "Direct comparison of three commercially available devices for voice ambulatory monitoring and biofeedback," *SIG 3 Perspectives on Voice and Voice Disorders*, vol. 24, no. 2, pp. 80–86, 2014.

[106] D. Mehta, M. Zañartu, S. Feng, H. C. II, and R. Hillman, "Mobile voice health monitoring using a wearable accelerometer sensor and a smartphone platform," *IEEE Trans. on Biom. Eng.*, vol. 59, no. 11, pp. 3090–3096, 2012.

[107] R. E. Hillman, M. Zañartu, M. Ghassemi, D. D. Mehta, J. H. Van Stan, H. A. Cheyne II, and J. V. Guttag, "Future directions in the development of ambulatory monitoring for clinical voice assessment," in *10th International Conference Advances in Quantitative Laryngology*, 2013, p. 23.

[108] D. D. Mehta, M. Zanartu, J. H. Van Stan, S. W. Feng, H. A. Cheyne II, and R. E. Hillman, "Smartphone-based detection of voice disorders by long-term monitoring of neck acceleration features," in *Body Sensor Networks (BSN), 2013 IEEE International Conference on.* IEEE, 2013, pp. 1–6.

[109] M. Zañartu, V. Espinoza, D. D. Mehta, J. H. Van Stan, H. A. C. II, M. Ghassemi, J. V. Guttag, and R. E. Hillman, "Toward and objective aerodynamic assessment of vocal hyperfunction using a voice health monitor," *Models and Analysis of Vocal Emissions for Biomedical Applications*, p. 167, 2013.

[110] J. L. Spielman, E. J. Hunter, A. E. Halpern, and I. R. Titze, "Measuring improvement in teachers with voice complaints using the inability to produce soft voice (IPSV): Preliminary data."

[111] N. Smith, T. Klongtruagrok, G. DeSouza, C. Shyu, M. Dietrich, and M. Page, "Non-invasive ambulatory monitoring of complex sEMG patterns and its potential application in the detection of vocal dysfunctions," in *e-Health Networking, Applications and Services (Healthcom), 2014 IEEE 16th International Conference on*, Oct. 2014, pp. 447–452.

[112] A. Fridlund and J. Cacioppo, "Guidelines for human electromyographic research," *Psychophysiology*, vol. 23, no. 5, pp. 567–589, 1986.

[113] T. Hixon, G. Weismer, and J. Hoit, *Preclinical speech science. Anatomy, physiology, acoustics, perception, 2nd ed.* Plural, 2013.

[114] M. Dietrich and K. V. Abbott, "Vocal function in introverts and extraverts during a psychological stress reactivity protocol," *Journal of Speech, Language, and Hearing Research*, vol. 55, no. 3, pp. 973–87, 2012.

[115] R. Ding, C. Larson, J. Logemann, and A. Rademaker, "Surface electromyographic and electroglottographic studies in normal subjects under two swallow conditions: Normal and during the mendelsohn manuever," *Dysphagia*, vol. 17, pp. 1–12, 2002.

[116] A. VanBoxtel, "Optimal signal bandwidth for the recording of surface EMG activity of facial, jaw, oral, and neck muscles," *Psychophysiology*, vol. 38, pp. 22–34, 2001.

[117] E. Yiu, K. Verdolini, and L. Chow, "Electromyographic study of motor learning for a voice production task," *J. Speech. Lang. Hear. Res*, vol. 48, no. 6, pp. 1254–1268, 2005.

[118] C. DeLuca, "The use of surface electromyography in biomechanics," *Journal of Applied Biomechanics*, vol. 13, no. 2, pp. 135–163, 1997.

[119] R. Colton, J. Casper, and R. Leonard, *Understanding voice problems: A physiological perspective for diagnosis and treatment, 3 ed.* Lippincott Williams and Wilkins, 2006.

[120] E. Grillo and K. Verdolini, "Evidence for distinguishing pressed, normal, resonant, and breathy voice qualities by laryngeal resistance and vocal efficiency in vocally trained subjects," *Journal of Voice*, vol. 22, pp. 546–552, 2008.

[121] R. I. Zraick, L. Smith-Olinde, and L. L. Shotts, "Adult normative data for the kaypentax phonatory aerodynamic system model 6600," *Journal of Voice*, vol. 26, pp. 164–176, 2012.

[122] P. Woo, R. Colton, and L. Shangold, "Phonatory airflow analysis in patients with laryngeal disease," *Annals of Otology, Rhinology and Laryngology*, vol. 96, pp. 549–555, 1987.

[123] N. Smith, T. Klongtruagrok, G. DeSouza, C. Shyu, M. Dietrich, and M. Page, "Non-invasive ambulatory monitoring of complex sEMG patterns and its potential application in the detection of vocal dysfunctions," in *e-Health Networking, Applications and Services (Healthcom), 2014 IEEE 16th International Conference on*, Oct 2014, pp. 447–452.

[124] A. Hyvärinen, "Cocktail party problem," ICA Research Centre, Helsinki University of Technology, Finland. [Online]. Available: http://research.ics.aalto.fi/ica/cocktail/cocktail_en.cgi

[125] Z. Szabó, "Information theoretical estimators toolbox," *Journal of Machine Learning Research*, vol. 15, pp. 283–287, 2014, (https://bitbucket.org/szzoli/ite/).

[126] C. D. Manning, P. Raghavan, and H. Schütze, Eds., *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[127] J. Sivic and A. Zisserman, "Efficient visual search of videos cast as text retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 591–606, April 2009.

[128] T. Li, T. Mei, I. S. Kweon, and X. S. Hua, "Contextual bag-of-words for visual categorization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 4, pp. 381–392, April 2011.

[129] F. B. Pokorny, F. Graf, F. Pernkopf, and B. W. Schuller, "Detection of negative emotions in speech signals using bags-of-audio-words," in *Affective Computing and Intelligent Interaction (ACII), 2015 International Conference on*, Sept 2015, pp. 879–884.

[130] N. Keshava, "A Survey of Spectral Unmixing Algorithms," *Lincoln Laboratory Journal*, vol. 14, no. 1, 2003.

[131] C. Jiao and A. Zare, "Functions of multiple instances for learning target signatures," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4670–4686, Aug. 2015.

[132] K. Vongsy and M. J. Mendenhall, "A comparative study of spectral detectors," in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2011 3rd Workshop on*, June 2011, pp. 1–4.

[133] S. Kraut, L. Scharf, and R. Butler, "The adaptive coherence estimator: a uniformly most-powerful-invariant adaptive detection statistic," *Signal Processing, IEEE Transactions on*, vol. 53, no. 2, pp. 427–438, Feb 2005.

[134] A. M. Baldridge, S. J. Hook, C. Grove, and G. Rivera, "The aster spectral library version 2.0," *Remote Sensing of Environment*, vol. 113, no. 4, pp. 711–715, Apr. 2009.

[135] "Tigersense, the machine learning and sensing laboratory." [Online]. Available: http://engineers.missouri.edu/zarea/tigersense/code/

[136] S. Holzwarth, A. Müller, M. Habermeyer, R. Richter, A. Hausold, S. Thiemann, and P. Strobl, "Hysens - DAIS 7915/ ROSIS imaging spectrometers at DLR," in *3rd EARSeL Workshop on Imaging Spectroscopy, Herrsching*, May 2003, pp. 3–14.

[137] A. Merlo, D. Farina, and R. Merletti, "A fast and reliable technique for muscle activity detection from surface EMG signals," *Biomedical Engineering, IEEE Transactions on*, vol. 50, no. 3, pp. 316 –323, march 2003.

# Appendix A

# Complete Set of Results - Sensitivity of the Method

In this Appendix we present the results that we have obtained so far from the experiments described in Section 5.3.

## A.1 Effects of Weights (Sec. 5.3.1)

The next figures ratios obtained using signatures all 8 signatures. For illustration purposes, the ratios are shown as $3D$ surface plots, and as a $2D$ gray scale image. These results were obtained using the FastICA algorithm with randomly chosen initial estimates for the mixing matrix $A_{init}$, and Gaussian non-linearity, as explained in Section 5.3.1. The GUSSS ratios were obtained for different combinations of weights $w_1$ and $w_p$.

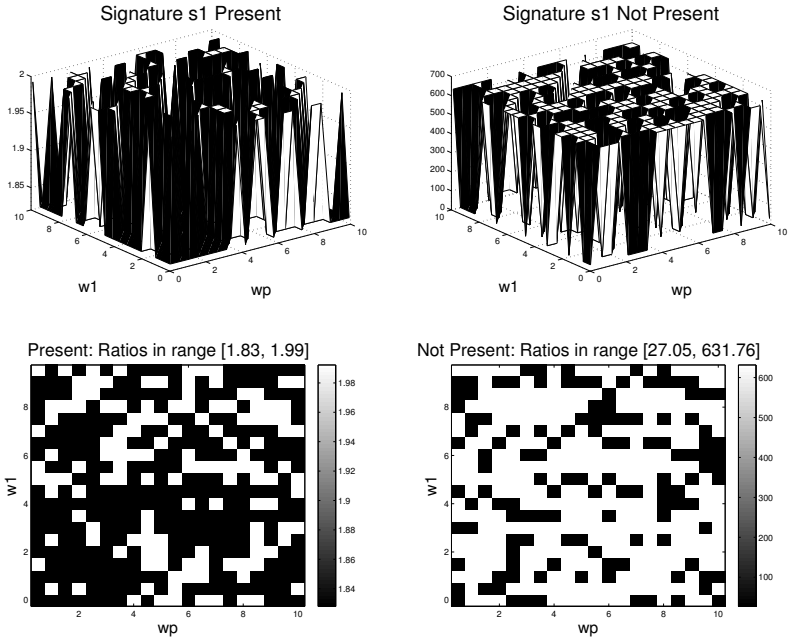



Figure A.1: Sought-after signature $s_1$.

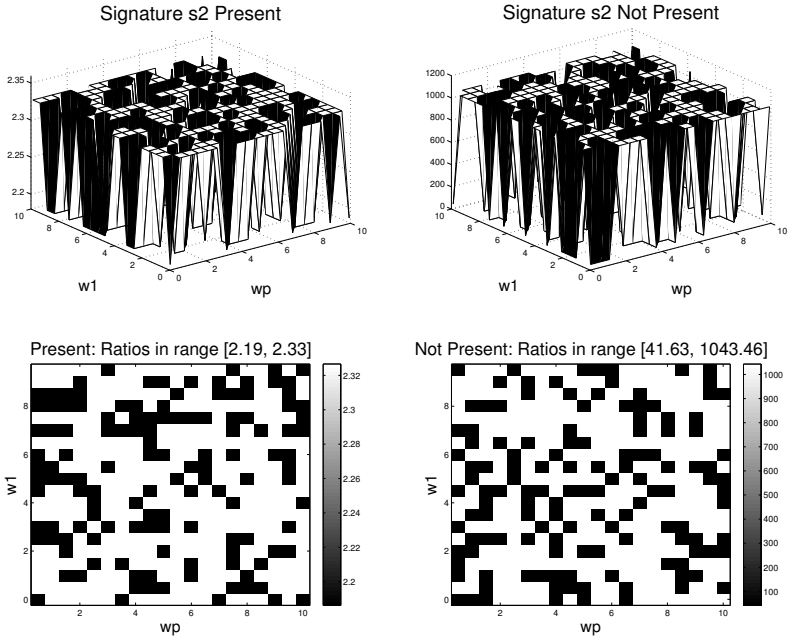



Figure A.2: Sought-after signature $s_2$.

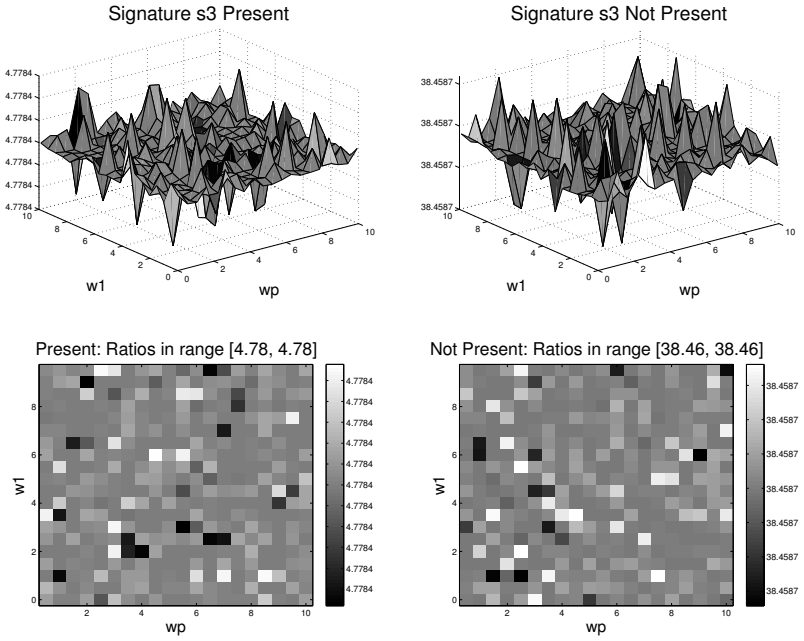



Figure A.3: Sought-after signature $s_3$.

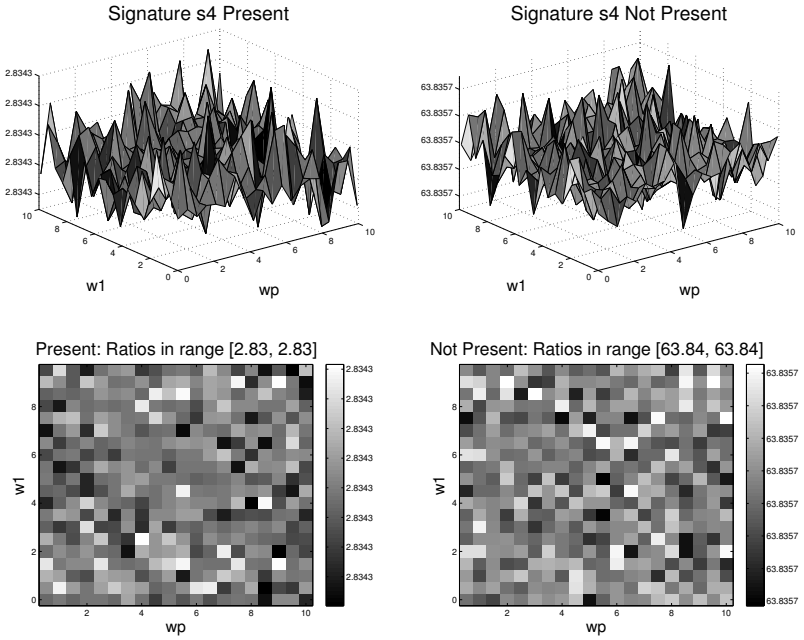



Figure A.4: Sought-after signature $s_4$.

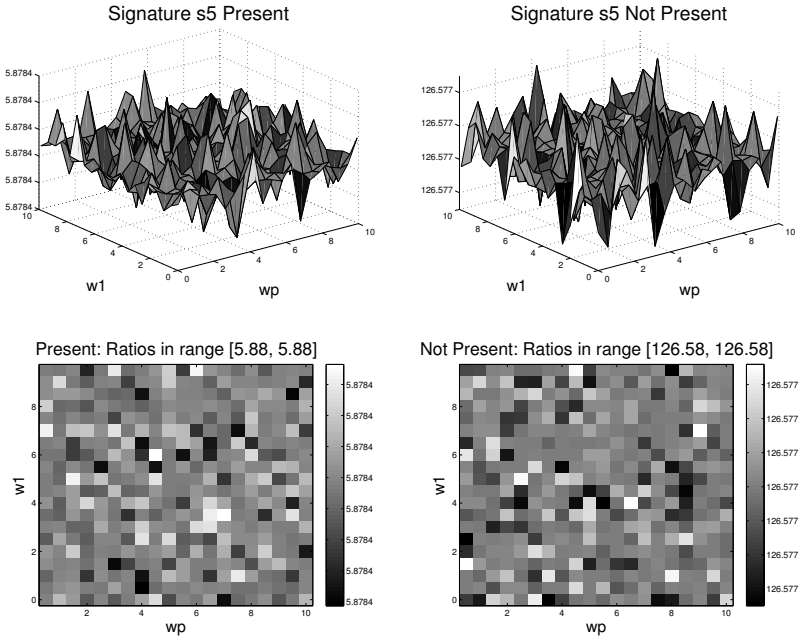



Figure A.5: Sought-after signature $s_5$.

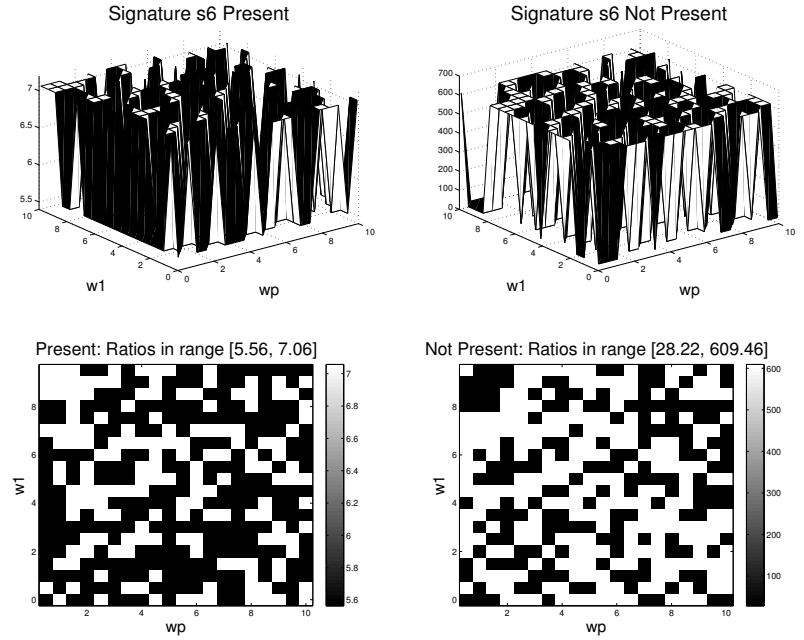



Figure A.6: Sought-after signature $s_6$.

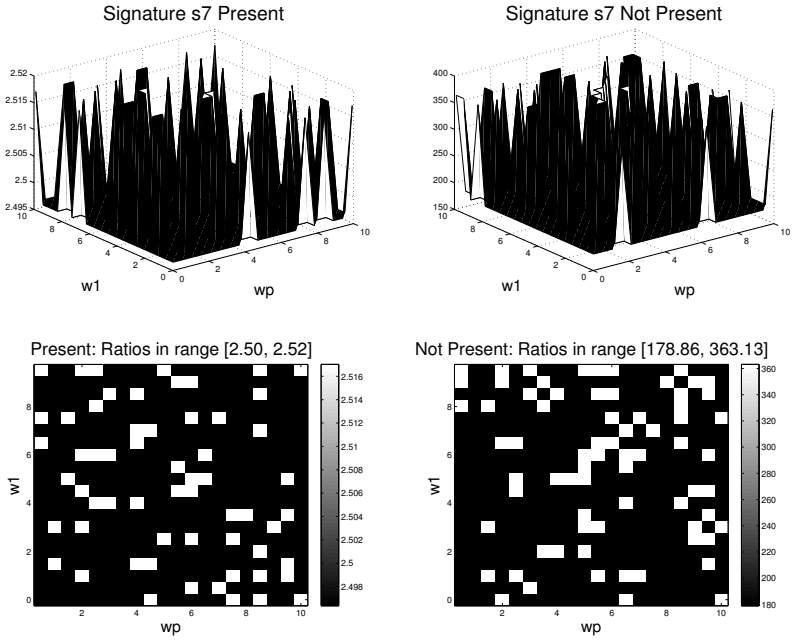



Figure A.7: Sought-after signature $s_7$.

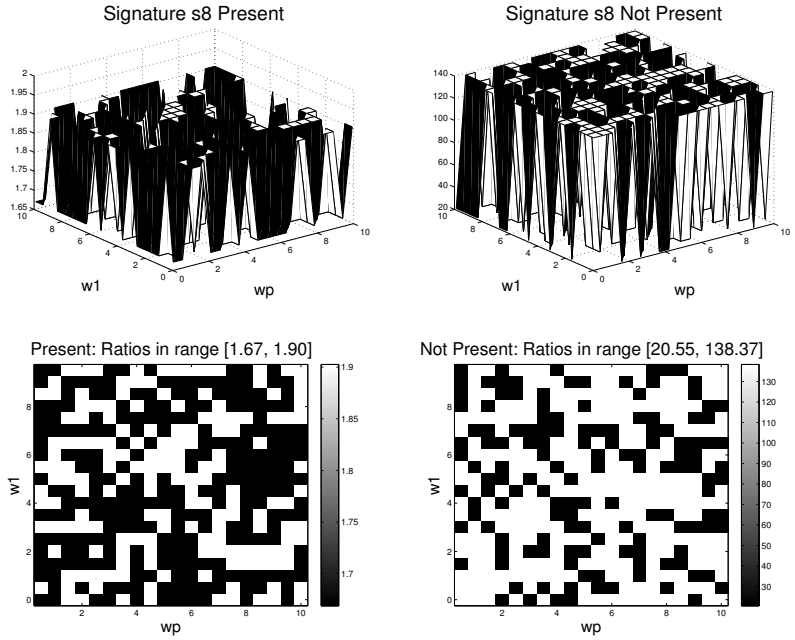



Figure A.8: Sought-after signature $s_8$.

## A.2 Effects of ICA Initialization (Sec. 5.3.2)

Figures A.9 through A.16 show ratios obtained for each signature, using 3 different initial estimates $A_{init}$ of the for the mixing matrix $A$: $A_{init}1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $A_{init}2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ and $A_{init}3 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$. Only $2D$ gray scale images are shown. Notice how $A_{init}2$ produces the more stable results in every case.

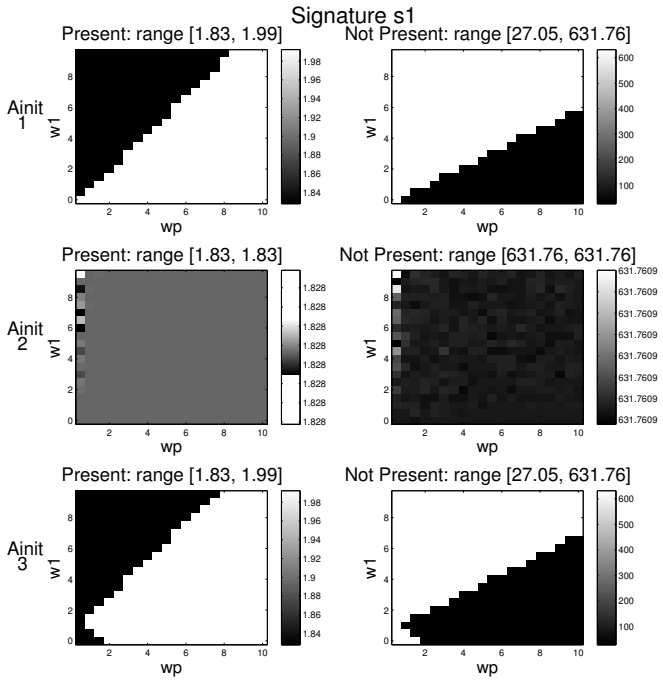



Figure A.9: Sought-after signature $s_1$.

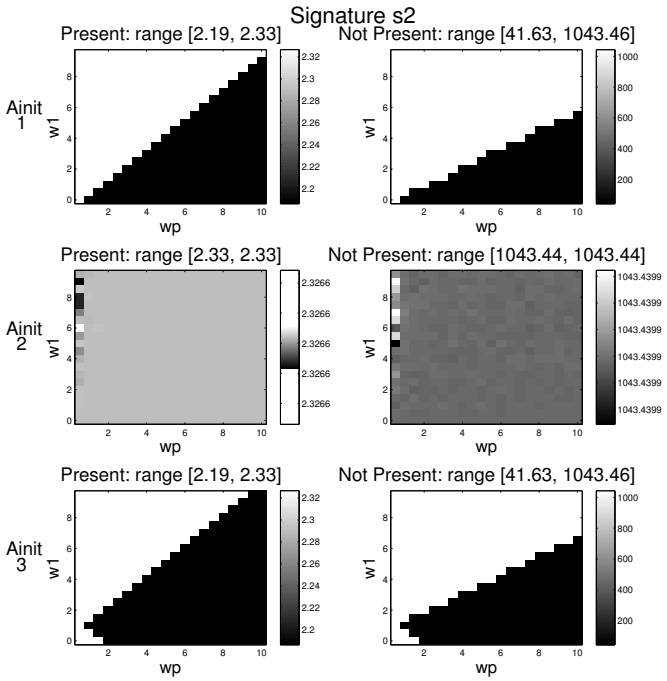



Figure A.10: Sought-after signature $s_2$.

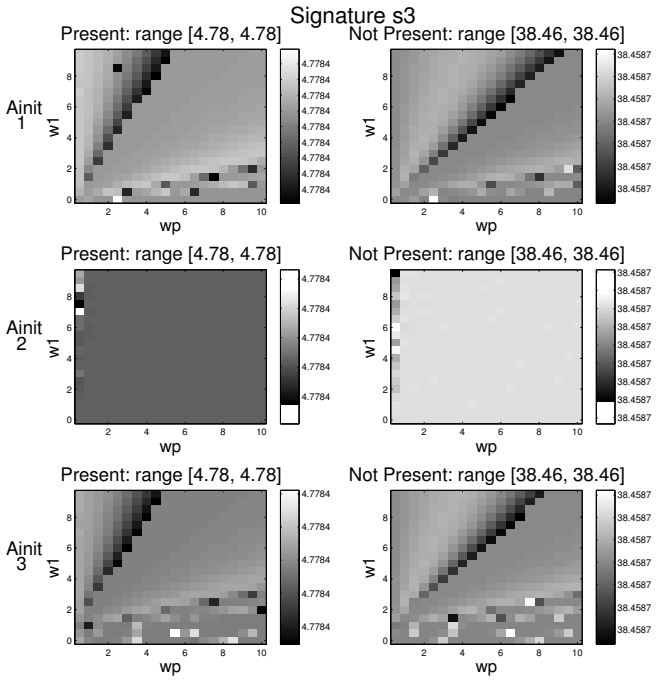



Figure A.11: Sought-after signature $s_3$.

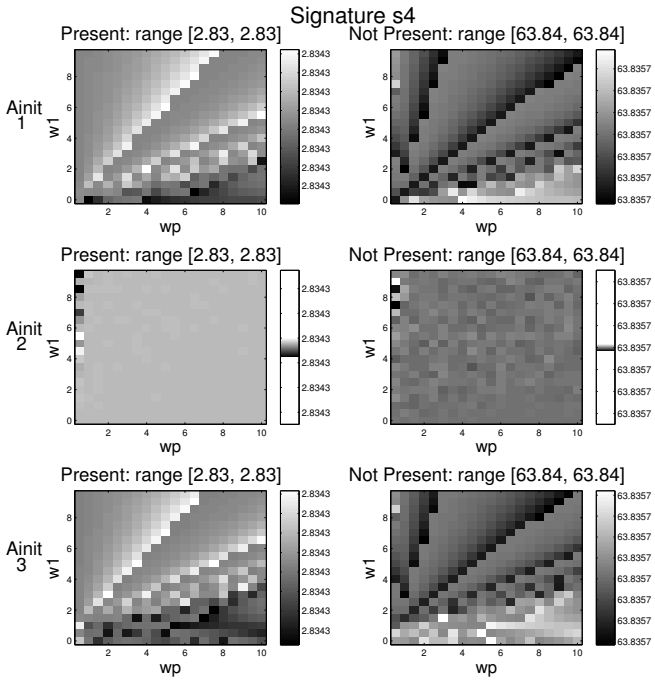



Figure A.12: Sought-after signature $s_4$.

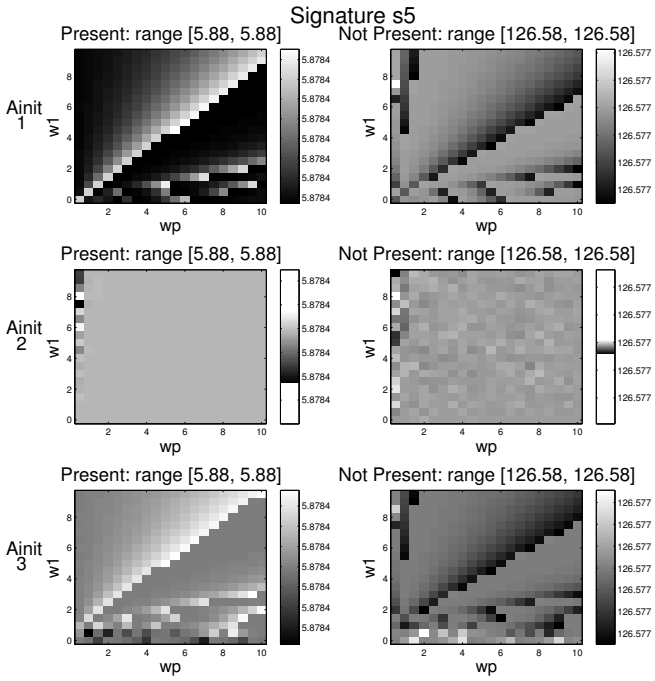



Figure A.13: Sought-after signature $s_5$.

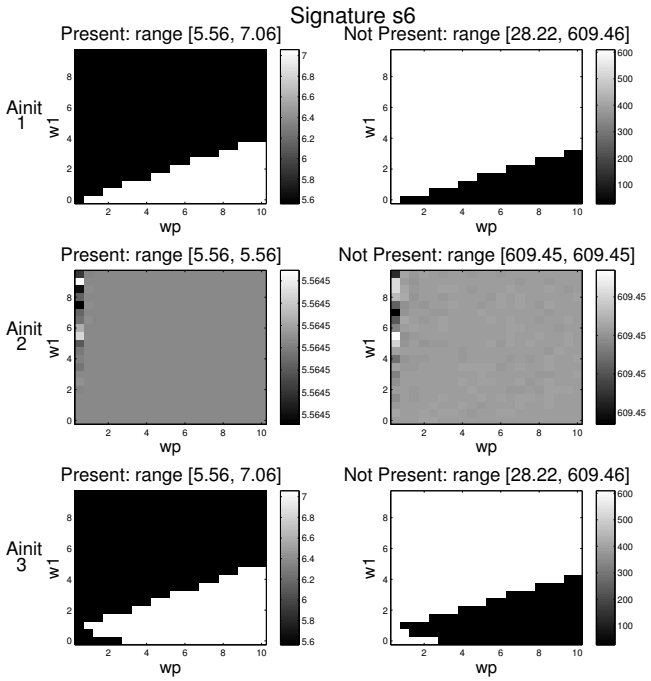



Figure A.14: Sought-after signature $s_6$.

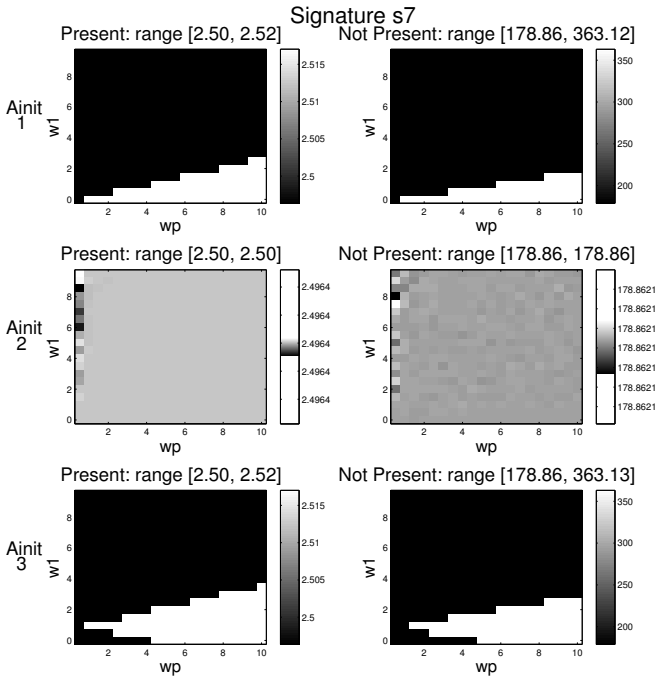



Figure A.15: Sought-after signature $s_7$.

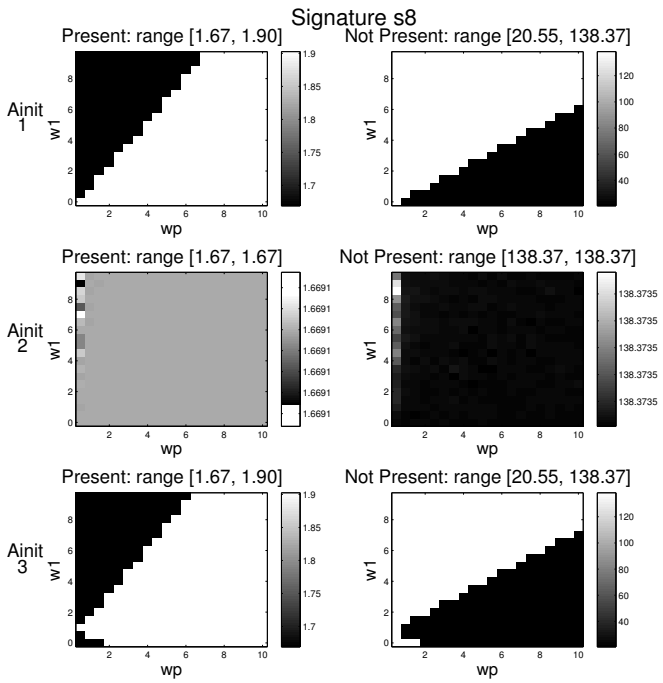



Figure A.16: Sought-after signature $s_8$.

Figure A.17 shows ratios obtained for signature $s_1$ (present and not present), using many additional initial estimates $A_{init}$, all of which are diagonal matrices. The values of the $A_{init}$ elements range from 0.1 to 2. The arrays shown above each 2D gray scale image represent the $A_{init}$ matrices. For example, $[2.0\ 0.0; 0.0\ 1.0] \Leftrightarrow \begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}$. Figures A.17 through A.24 are the corresponding figures for the remaining signatures. Notice how there is no significant difference in the ratio values when using different diagonal matrices as initial estimates.

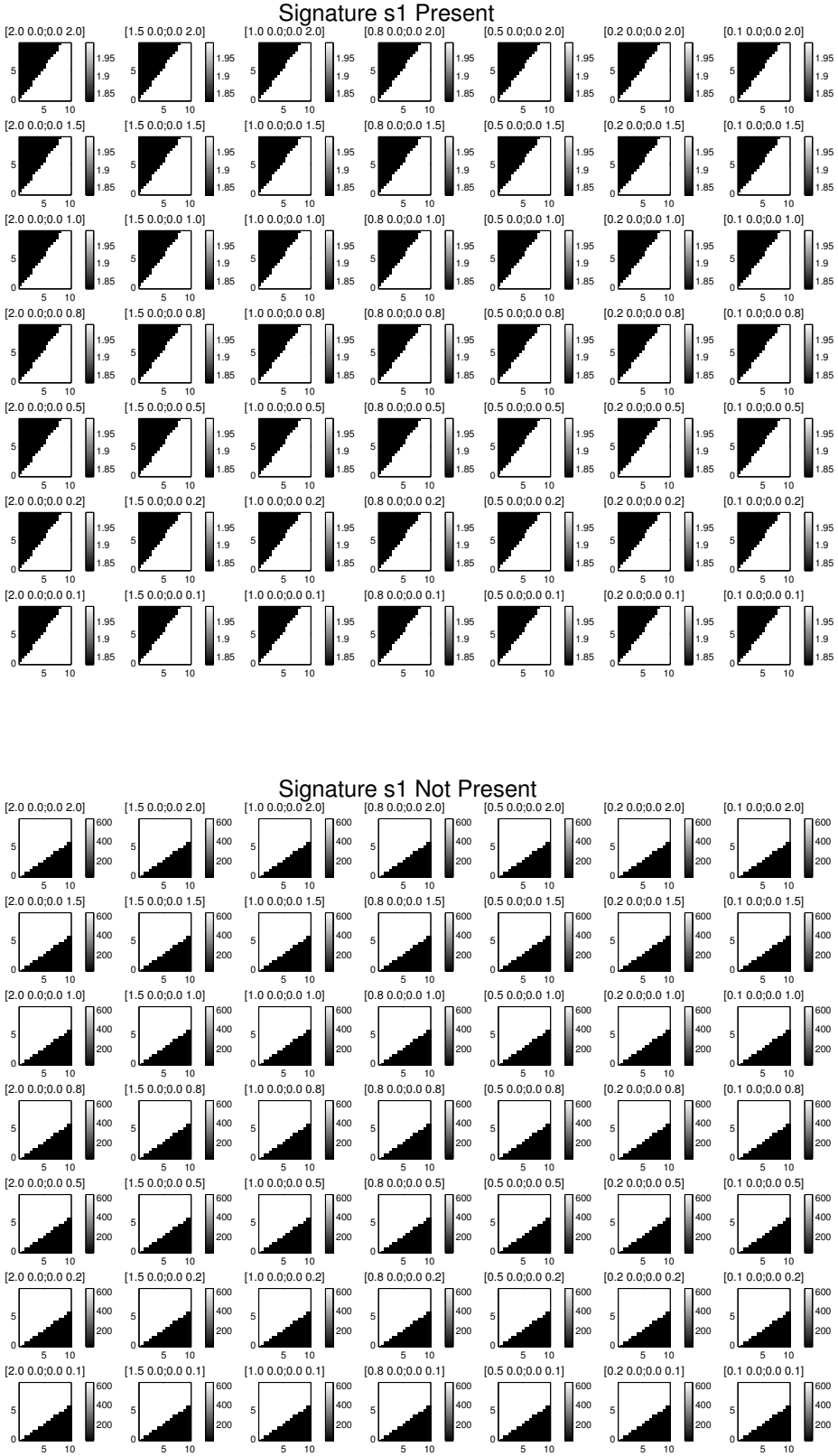



Figure A.17: Sought-after signature $s_1$.

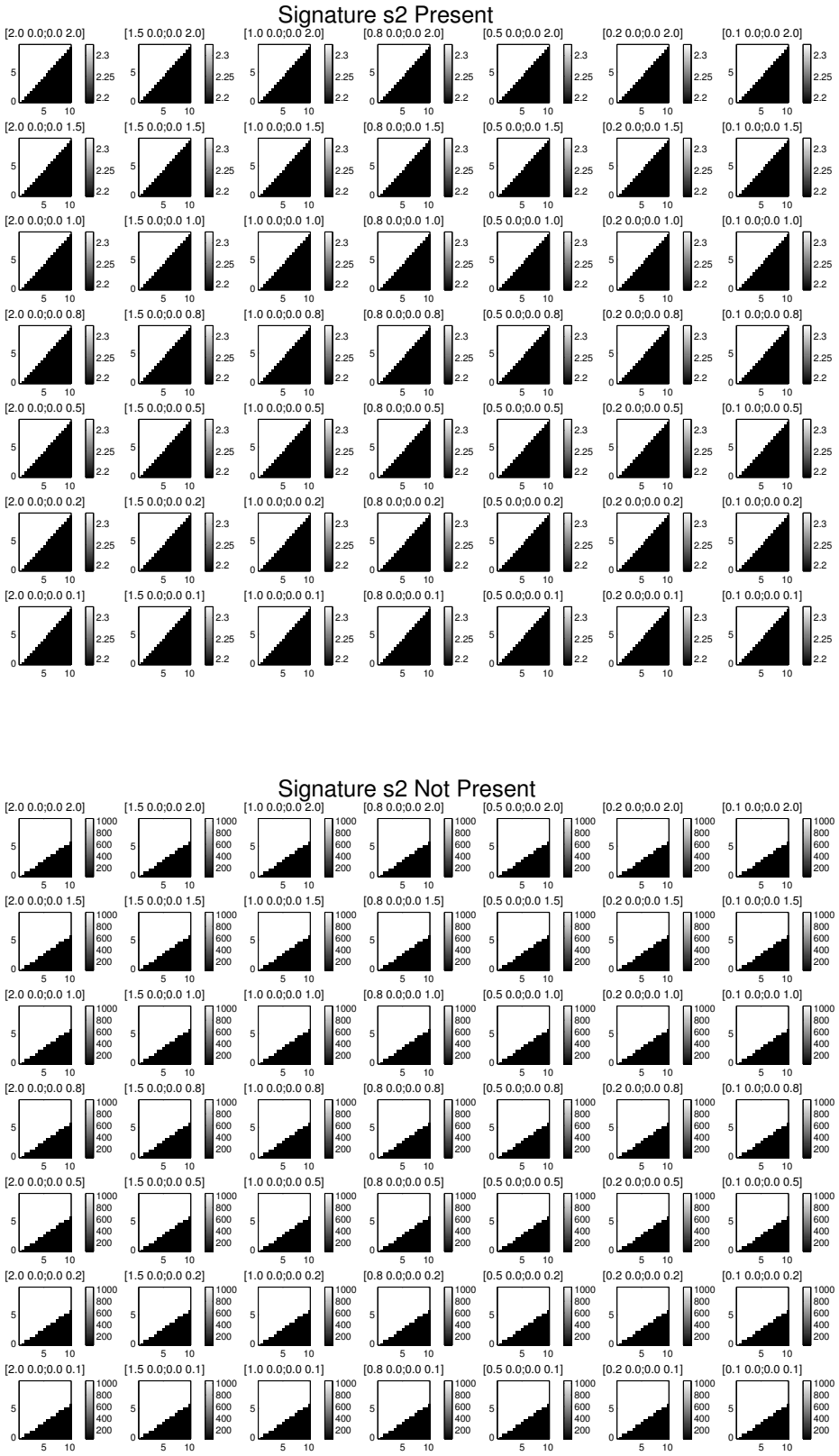



Figure A.18: Sought-after signature $s_2$.

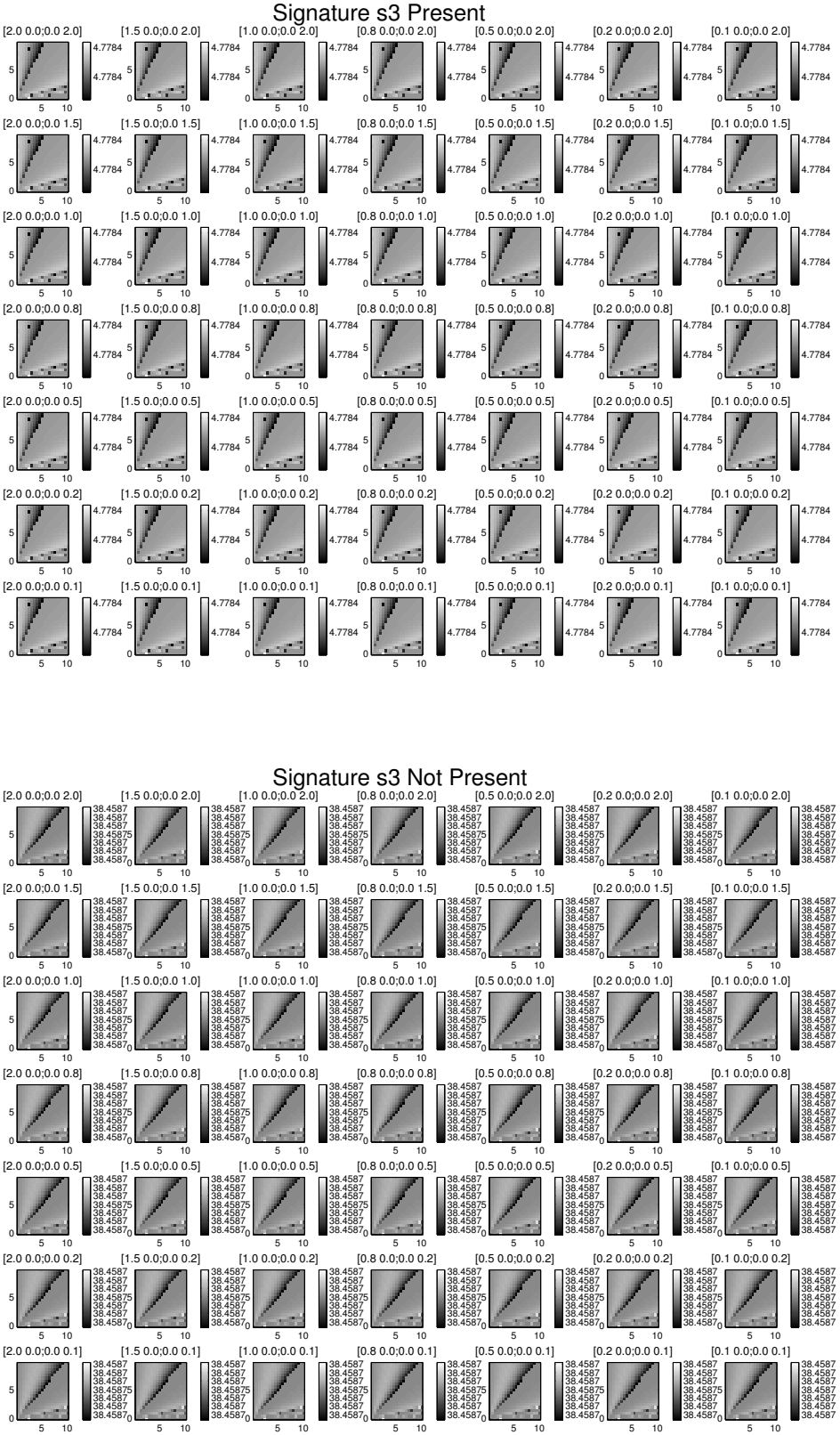



Figure A.19: Sought-after signature $s_3$.

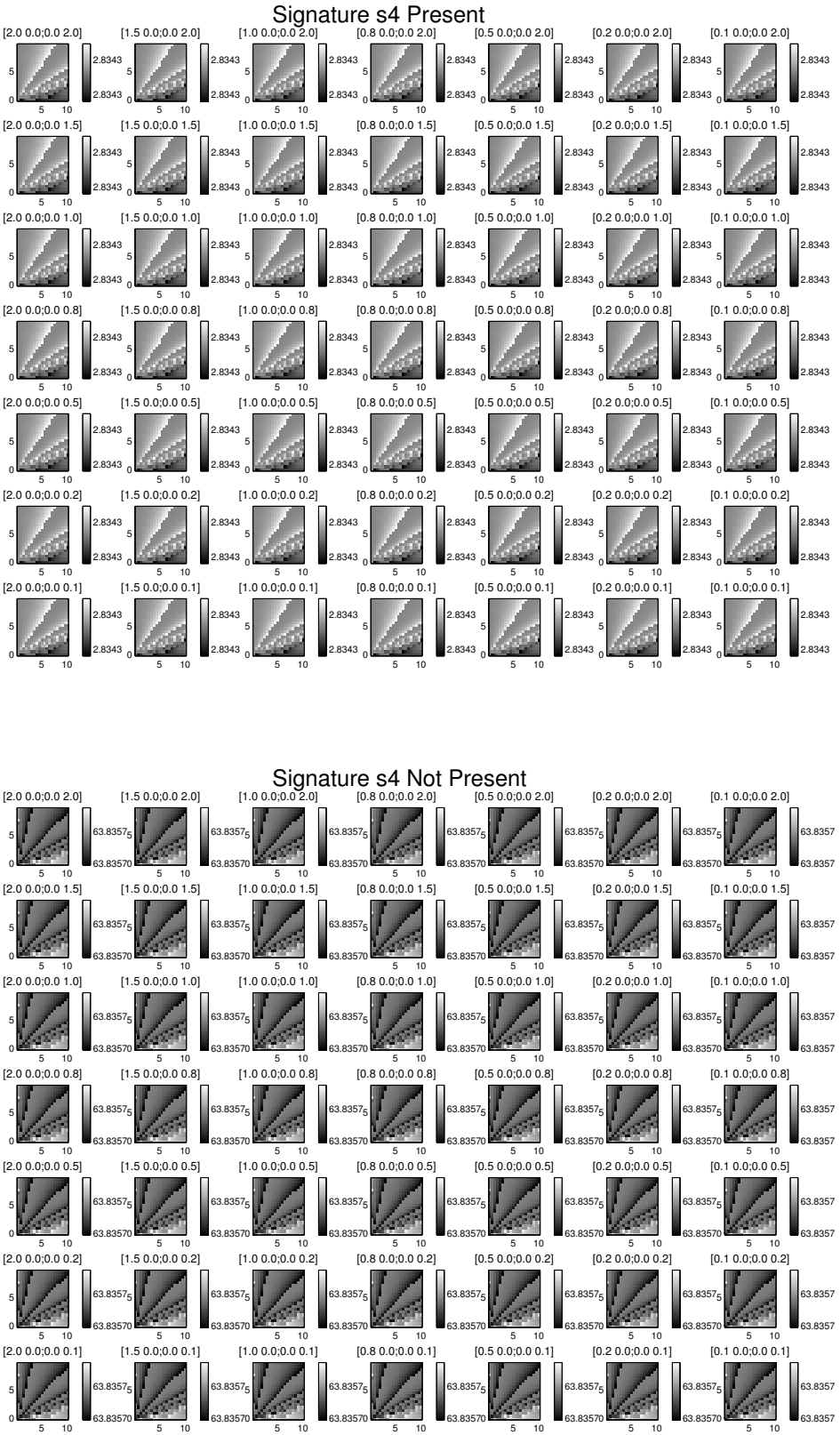



Figure A.20: Sought-after signature $s_4$.

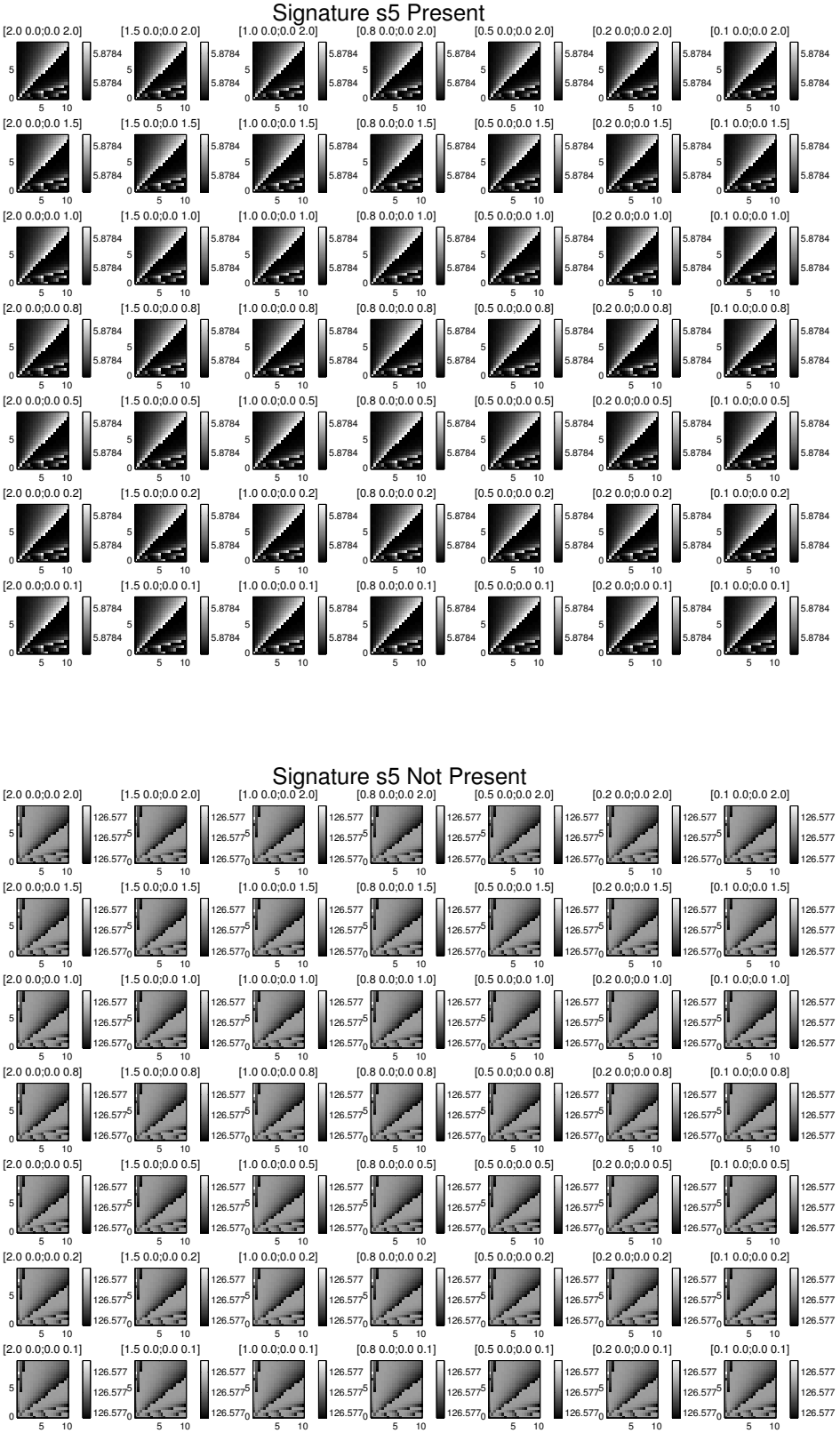



Figure A.21: Sought-after signature $s_5$.

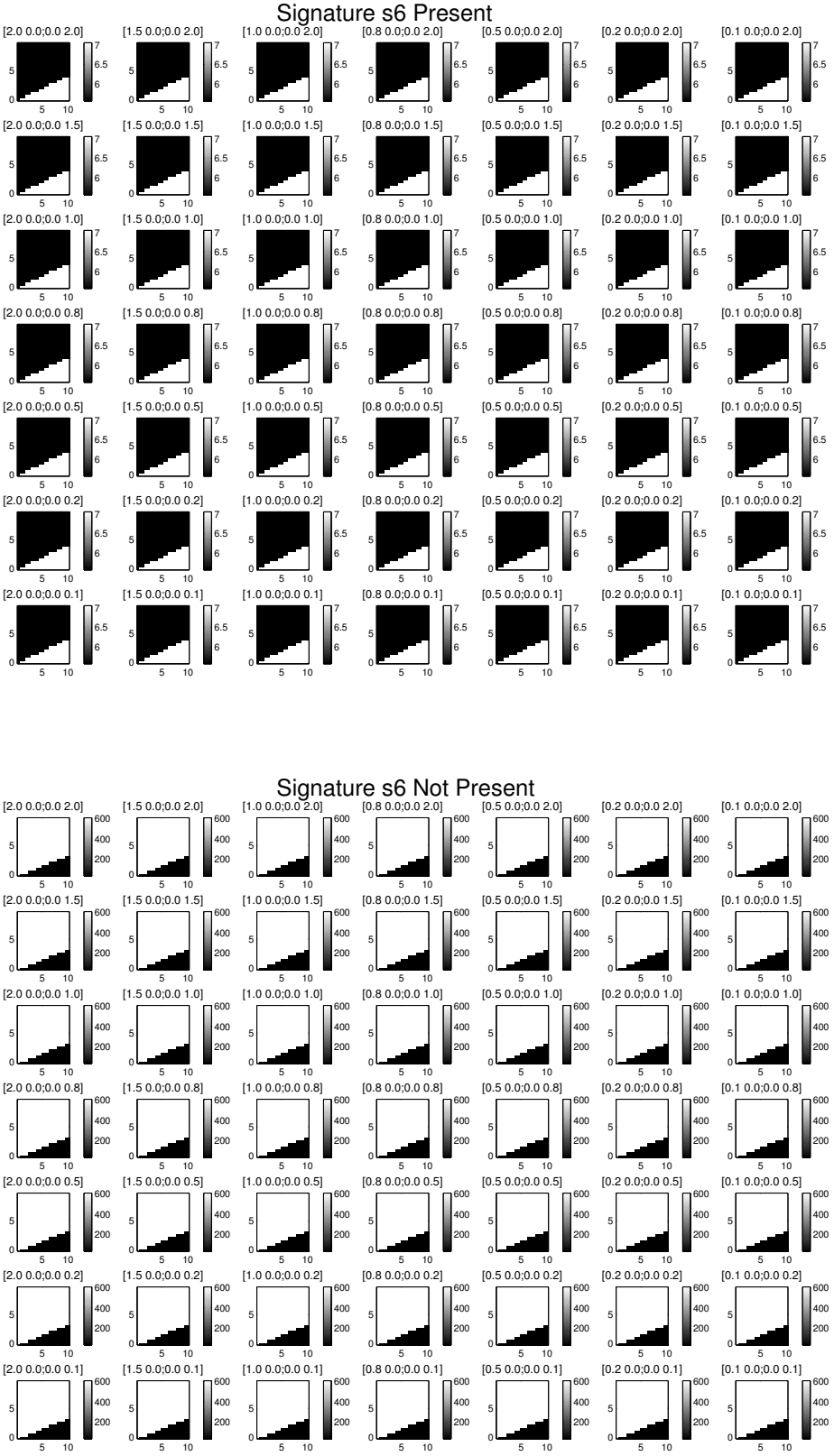



Figure A.22: Sought-after signature $s_6$.

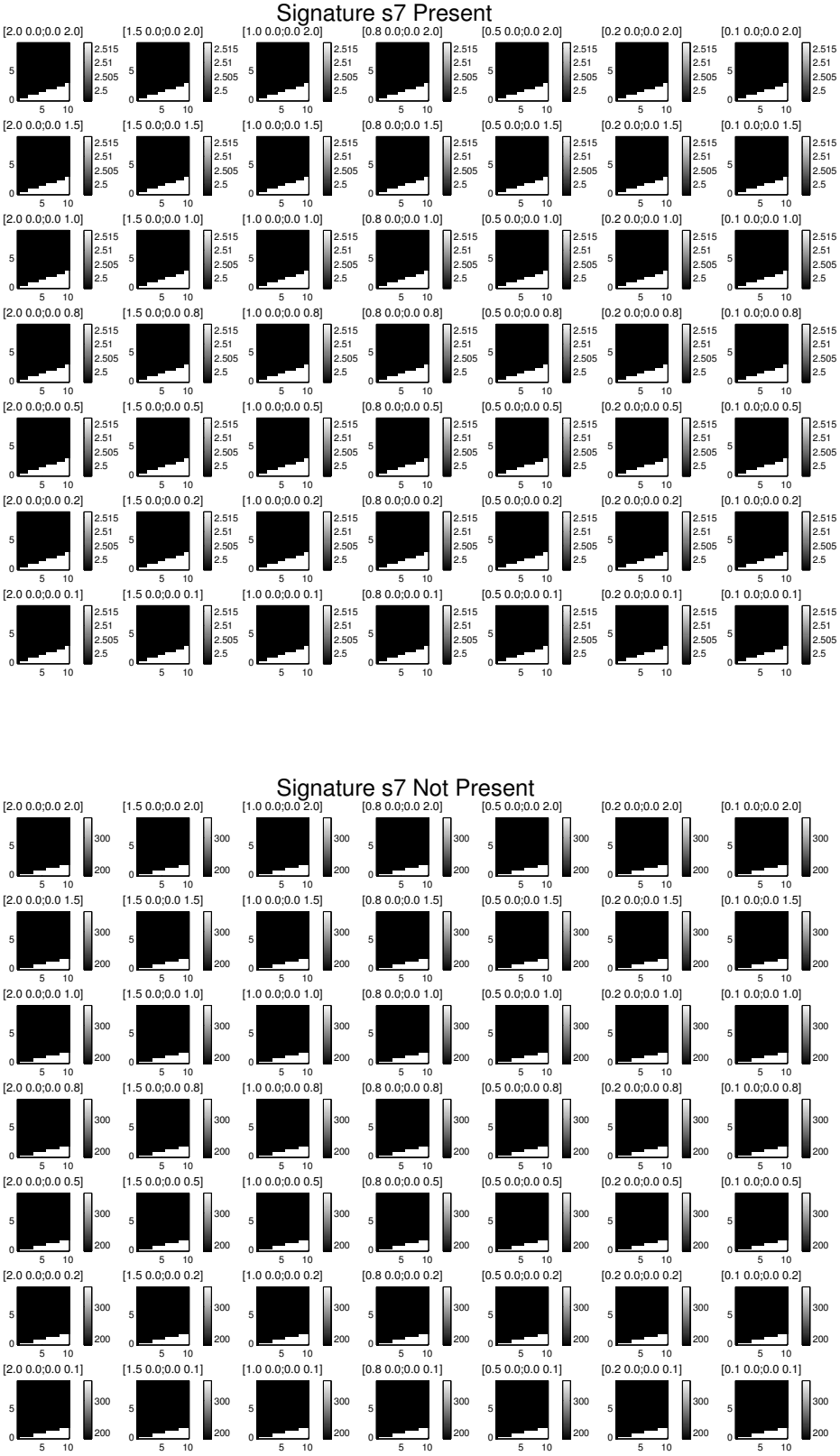



Figure A.23: Sought-after signature $s_7$.

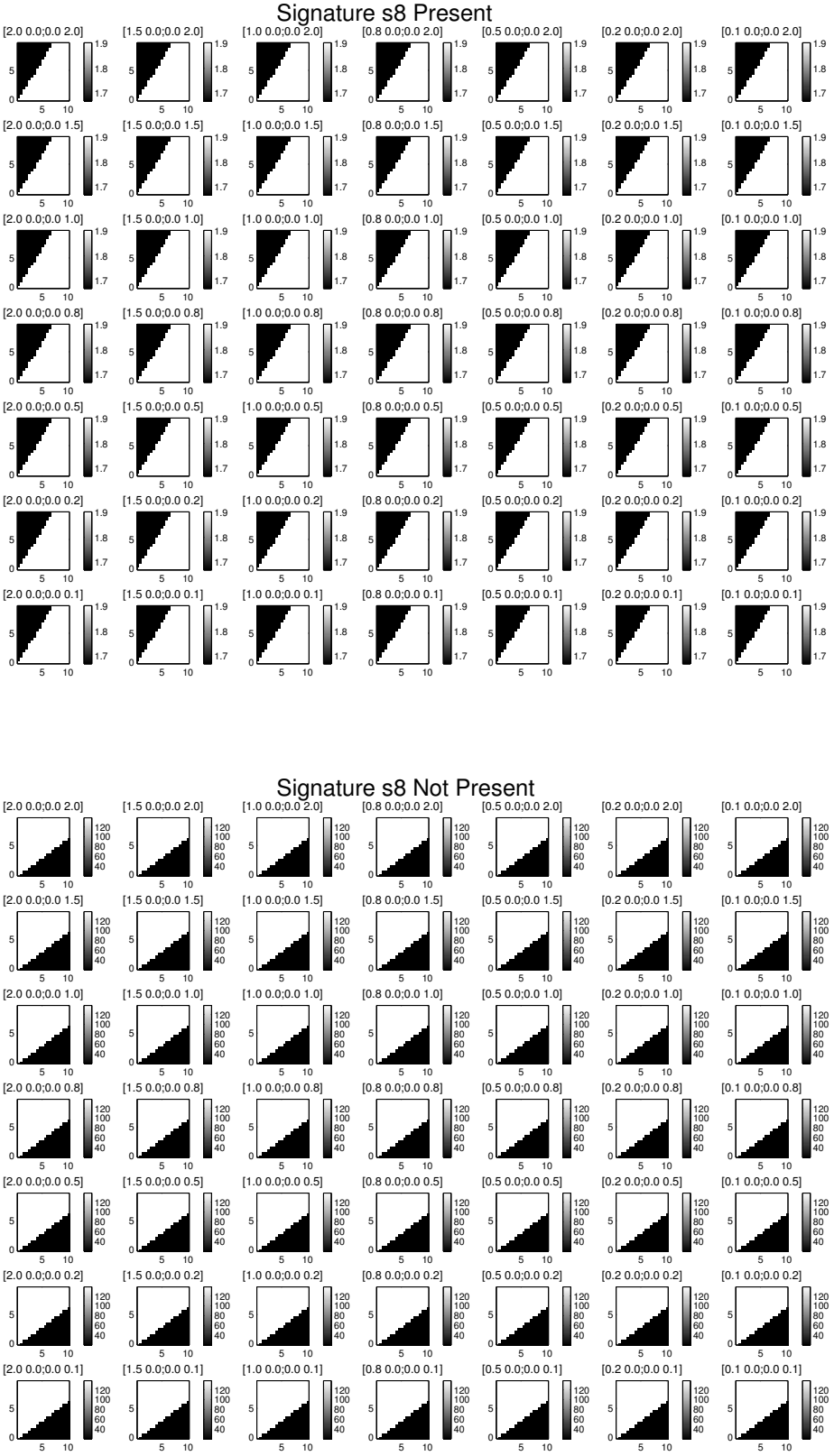



Figure A.24: Sought-after signature $s_8$.

We further tested with non diagonal $A_{init}$ matrices (but including anti-diagonal matrices). Figures A.25 through A.32 show the results. Notice how, in these cases, the ratio values are affected by different initial estimates. Furthermore, notice how the anti-diagonal matrices (right-most column) produce the most stable results, as observed before. The missing images (top right and bottom left) correspond to $A_{init}$ matrices $\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$ and $\begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$, which led to the ICA algorithm not to converge.

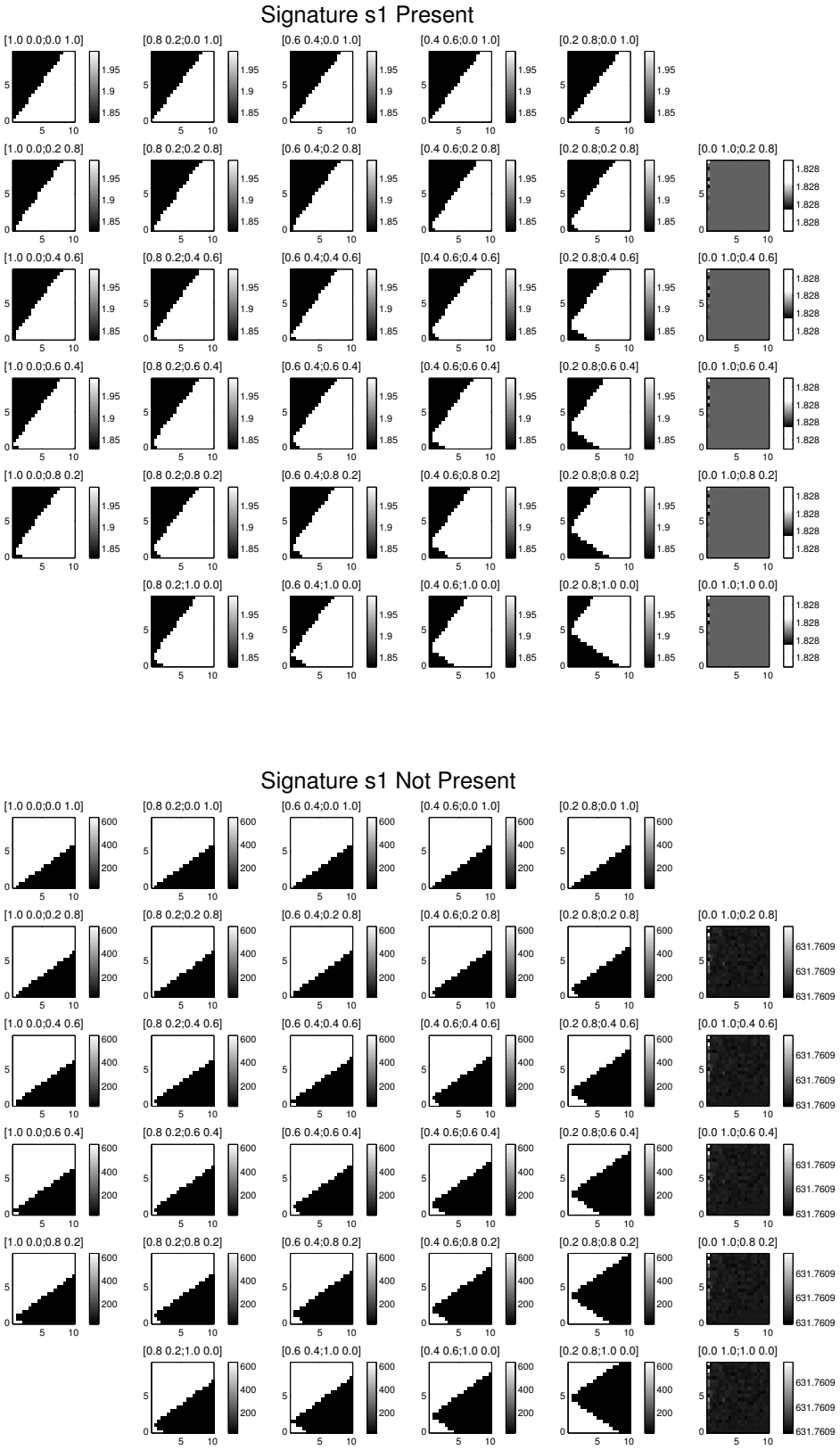



Figure A.25: Sought-after signature $s_1$.

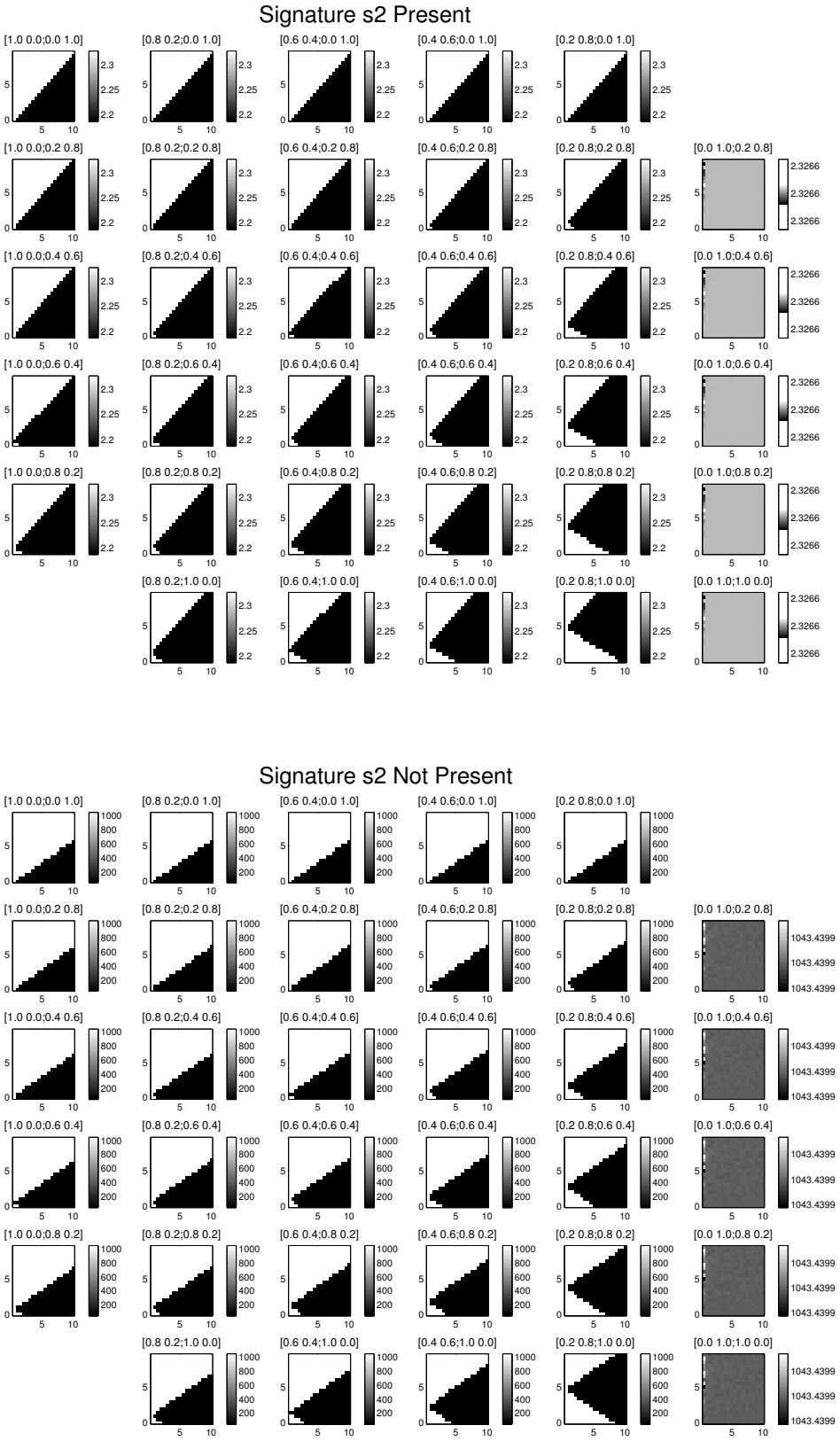



Figure A.26: Sought-after signature $s_2$.

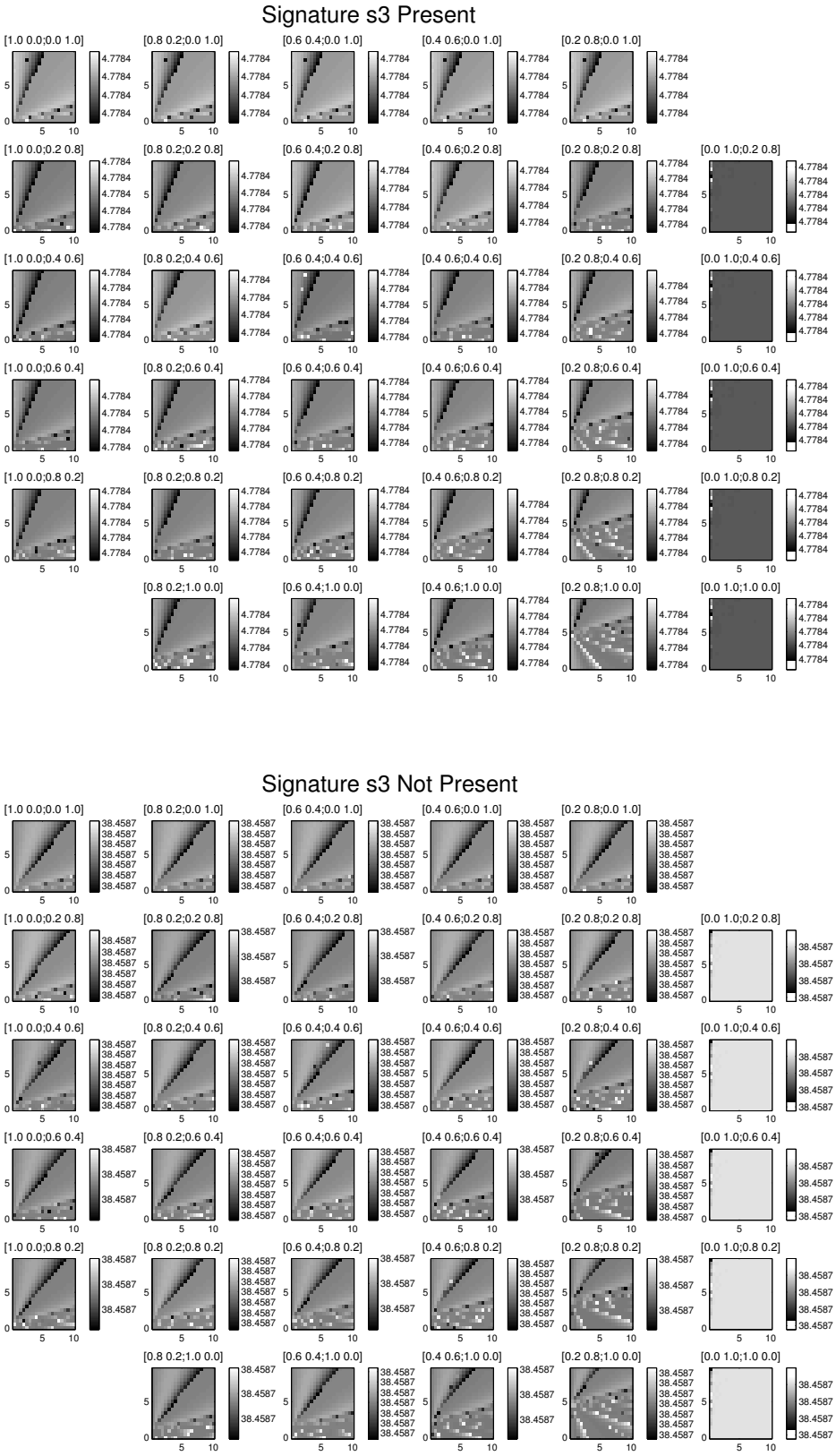



Figure A.27: Sought-after signature $s_3$.

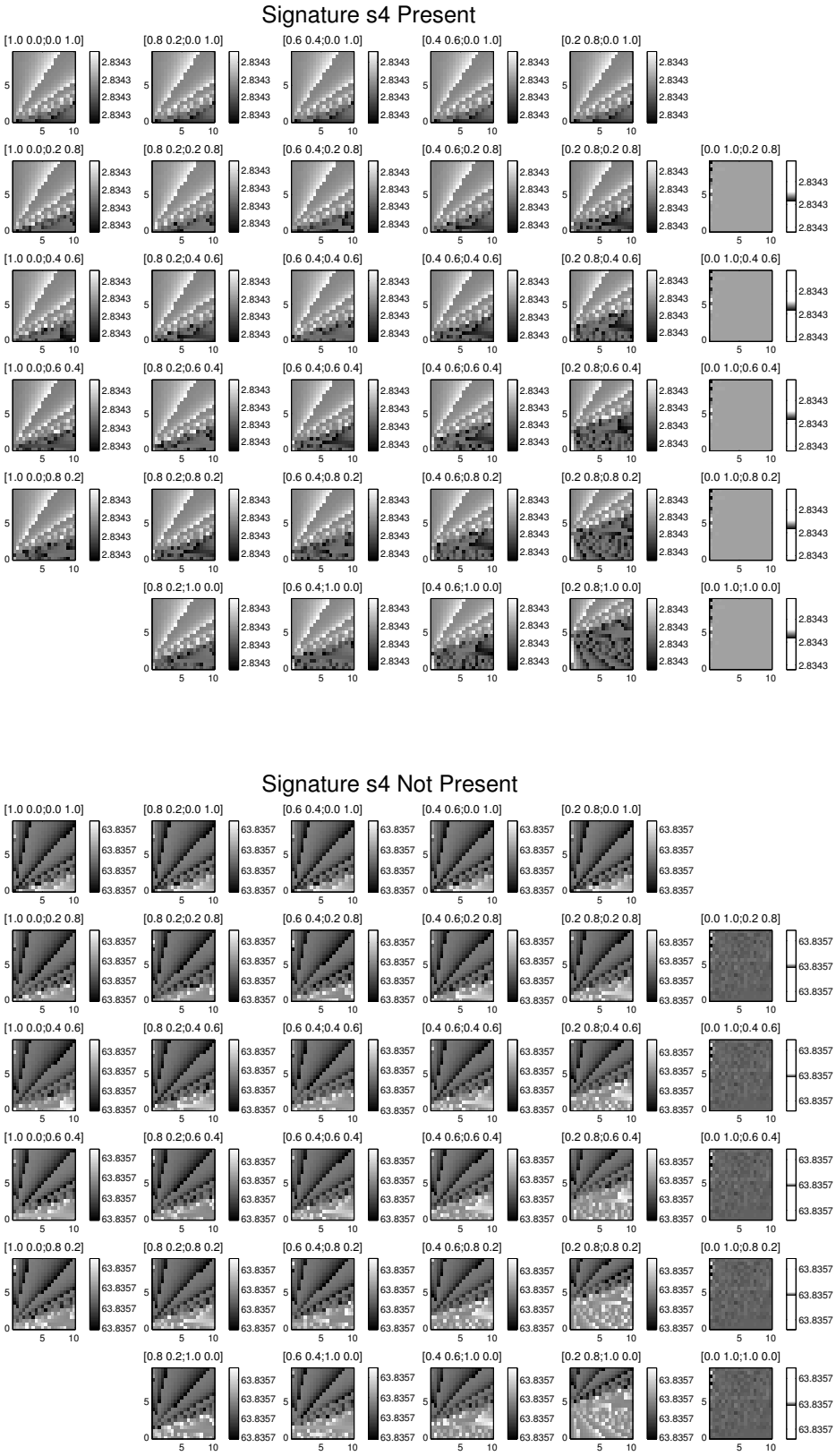



Figure A.28: Sought-after signature $s_4$.

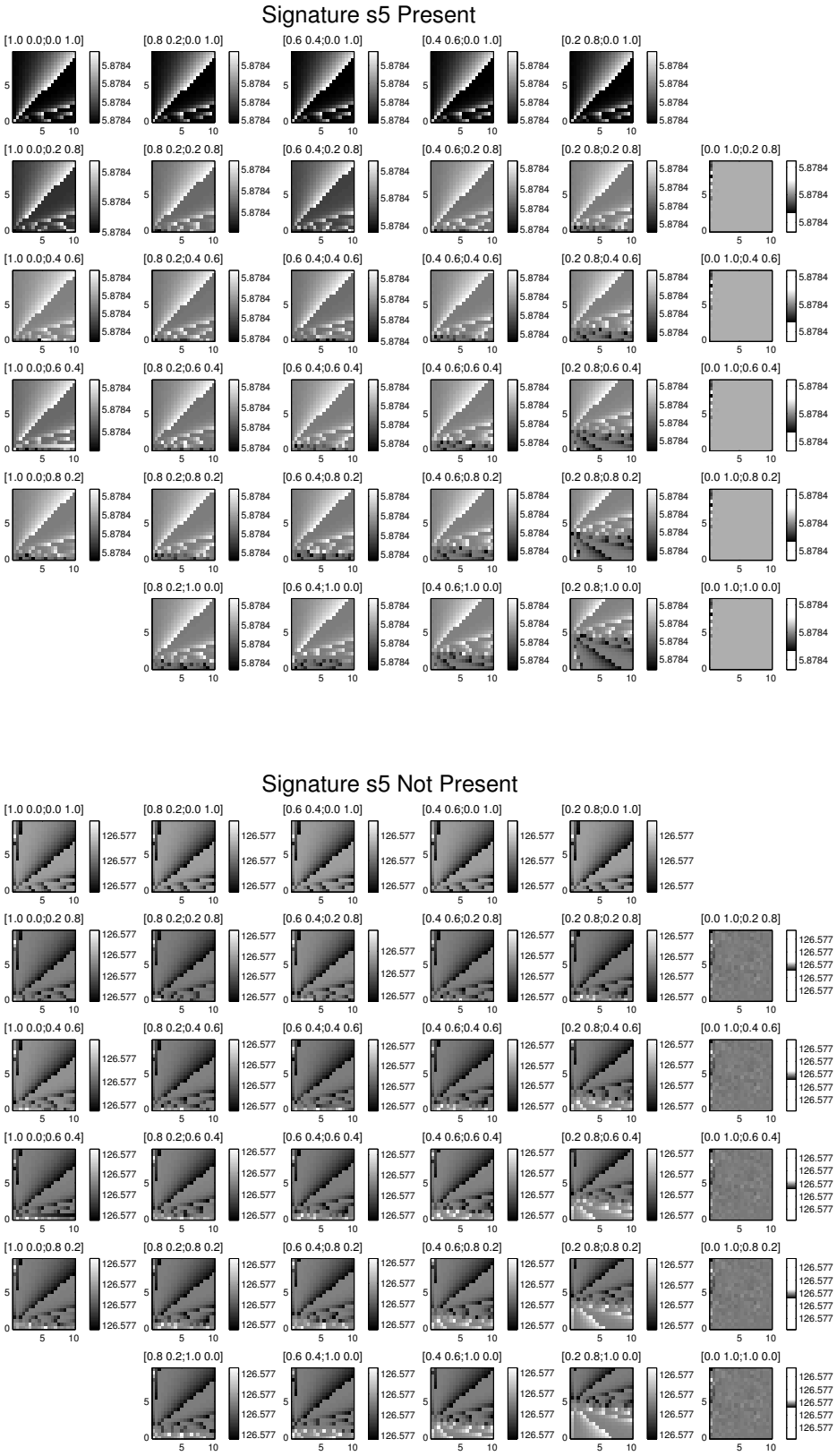



Figure A.29: Sought-after signature $s_5$.

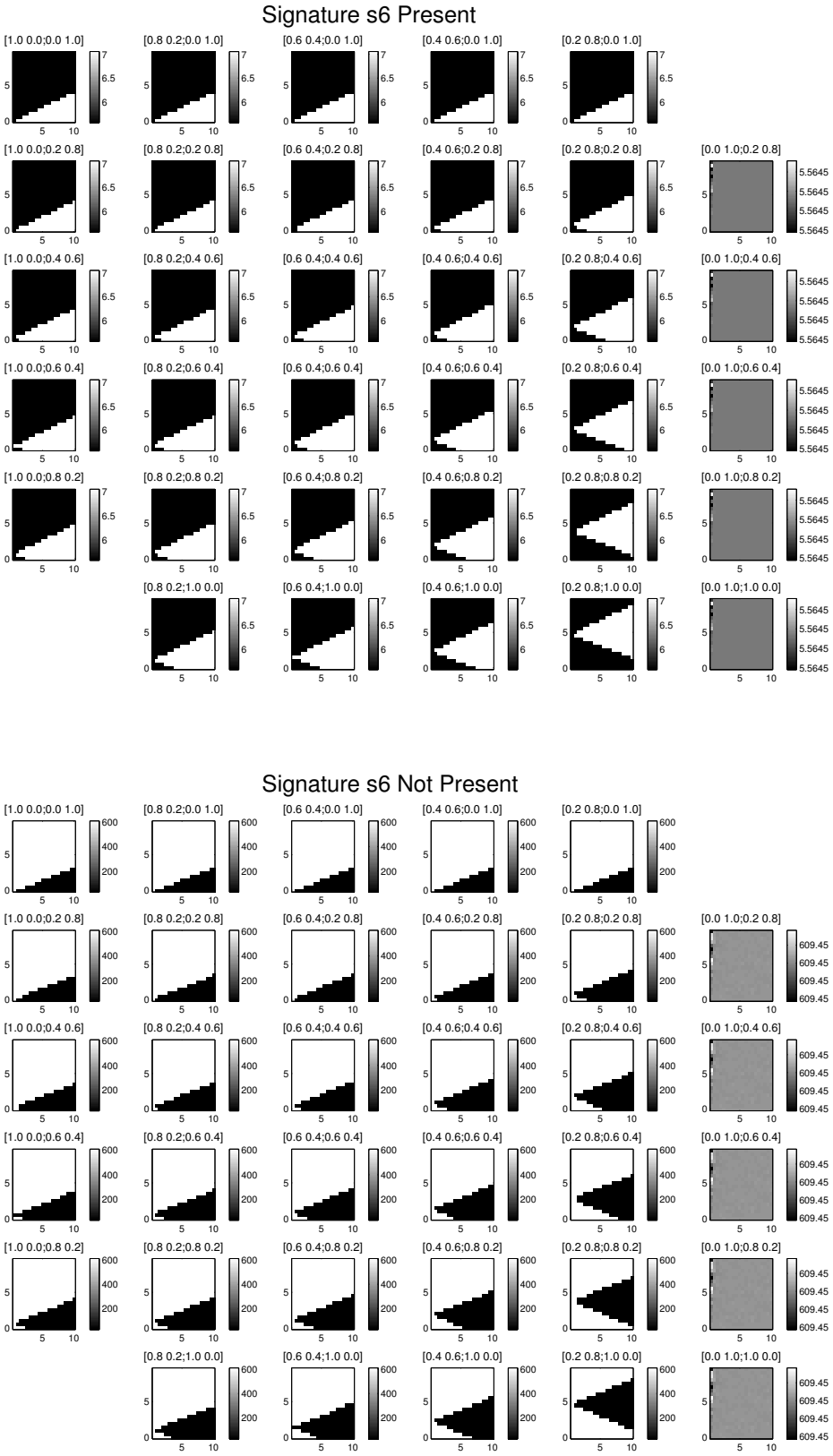



Figure A.30: Sought-after signature $s_6$.

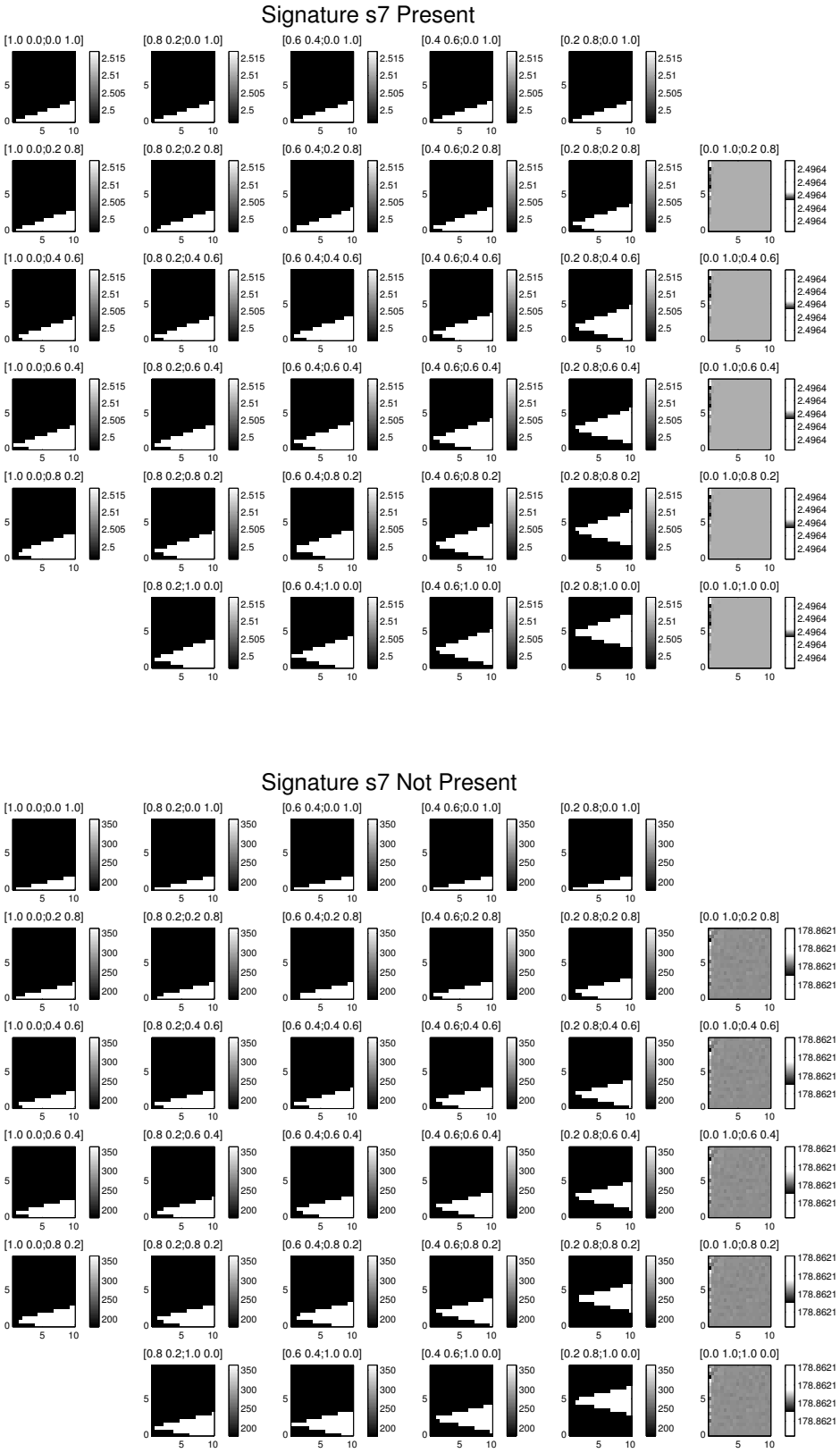



Figure A.31: Sought-after signature $s_7$.

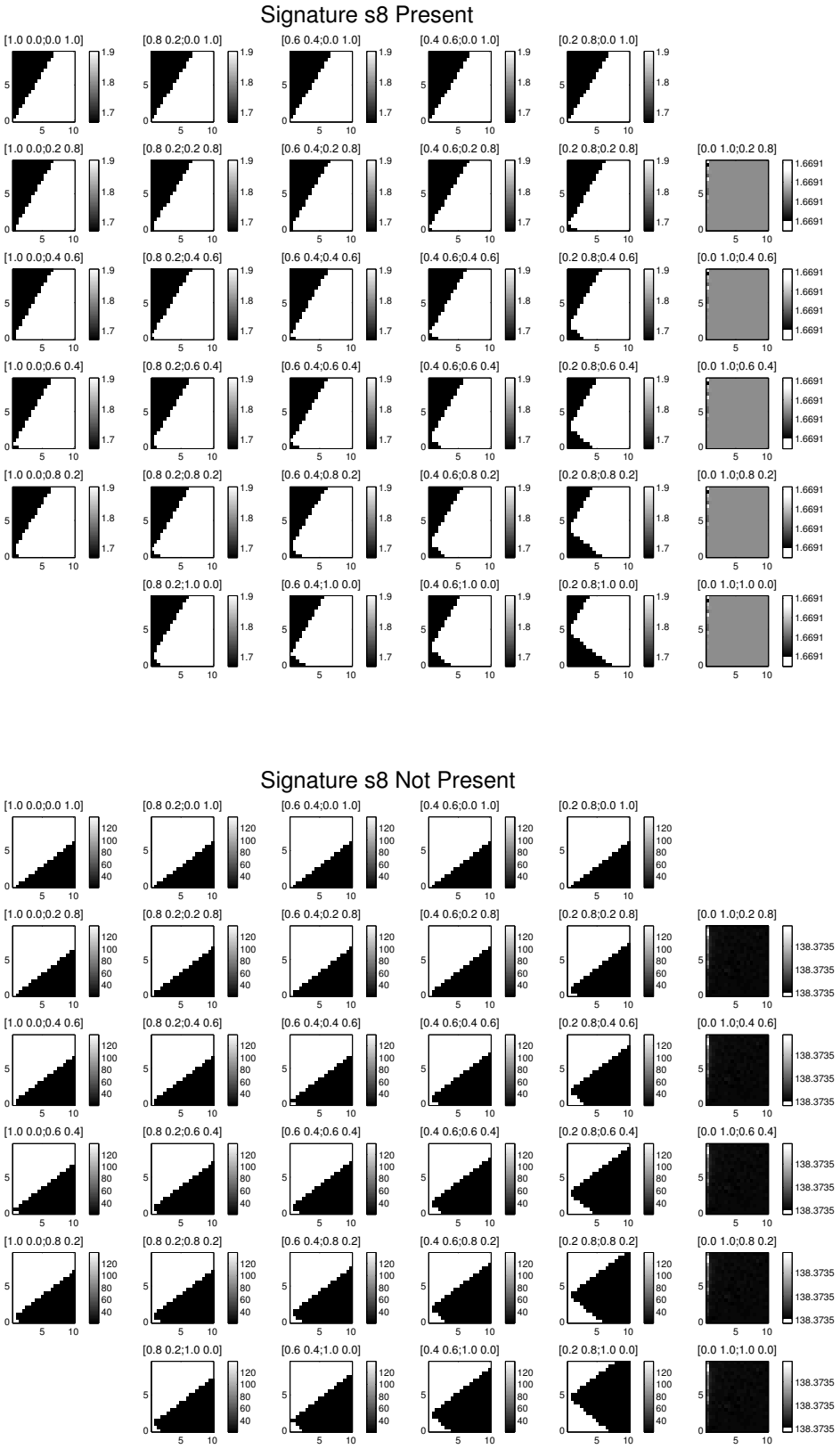



Figure A.32: Sought-after signature $s_8$.

## A.3 Effects of ICA Algorithm (Sec. 5.3.3)

Figures A.33 and A.34 show ratios obtained for the different signatures, using different FastICA configurations: 3 different $A_{init}$ matrices: $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ (FICA1); $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ (FICA2); and $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$ (FICA3); with both Gaussian (G) and $tanh()$ (T) nonlinearities. Notice, once again, the more stable results are obtained using the anti-diagonal initial estimates of the mixing matrix (FICA2 G and FICA2 T cases).

Figures A.35 and A.36 show the results obtained with other ICA algorithms. For comparison purposes, the first row shows results using FICA2 T. That FastICA variant was chosen given its stability. The next rows in the figures correspond to ITEFICA, JADE, KDICA and EFICA. A few other algorithms were tried as well (e.g. MILCA, RADICAL), but their execution times were too long, making them unappealing for most applications.

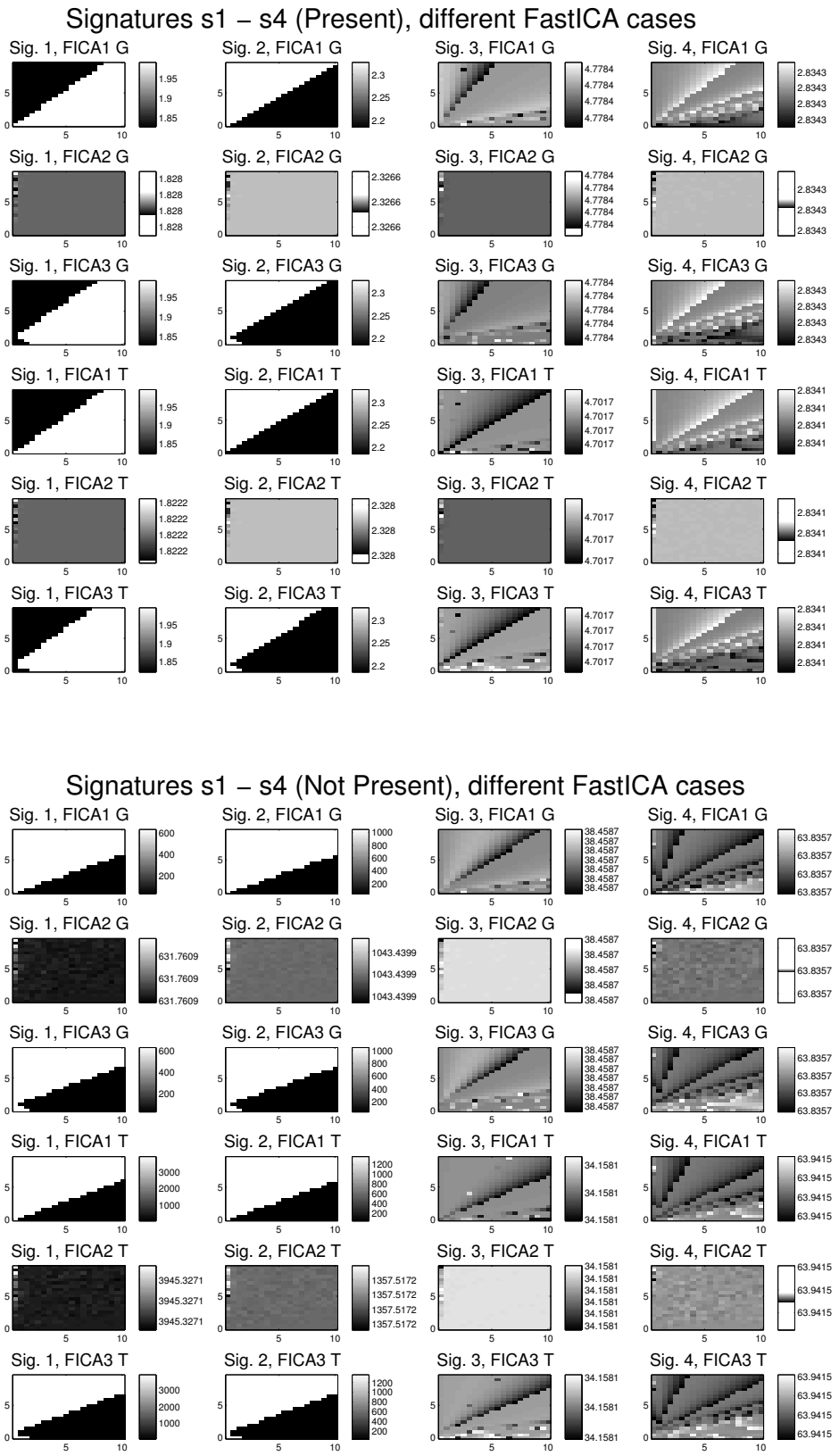



Figure A.33: Sought-after signatures $s_1 - s_4$.

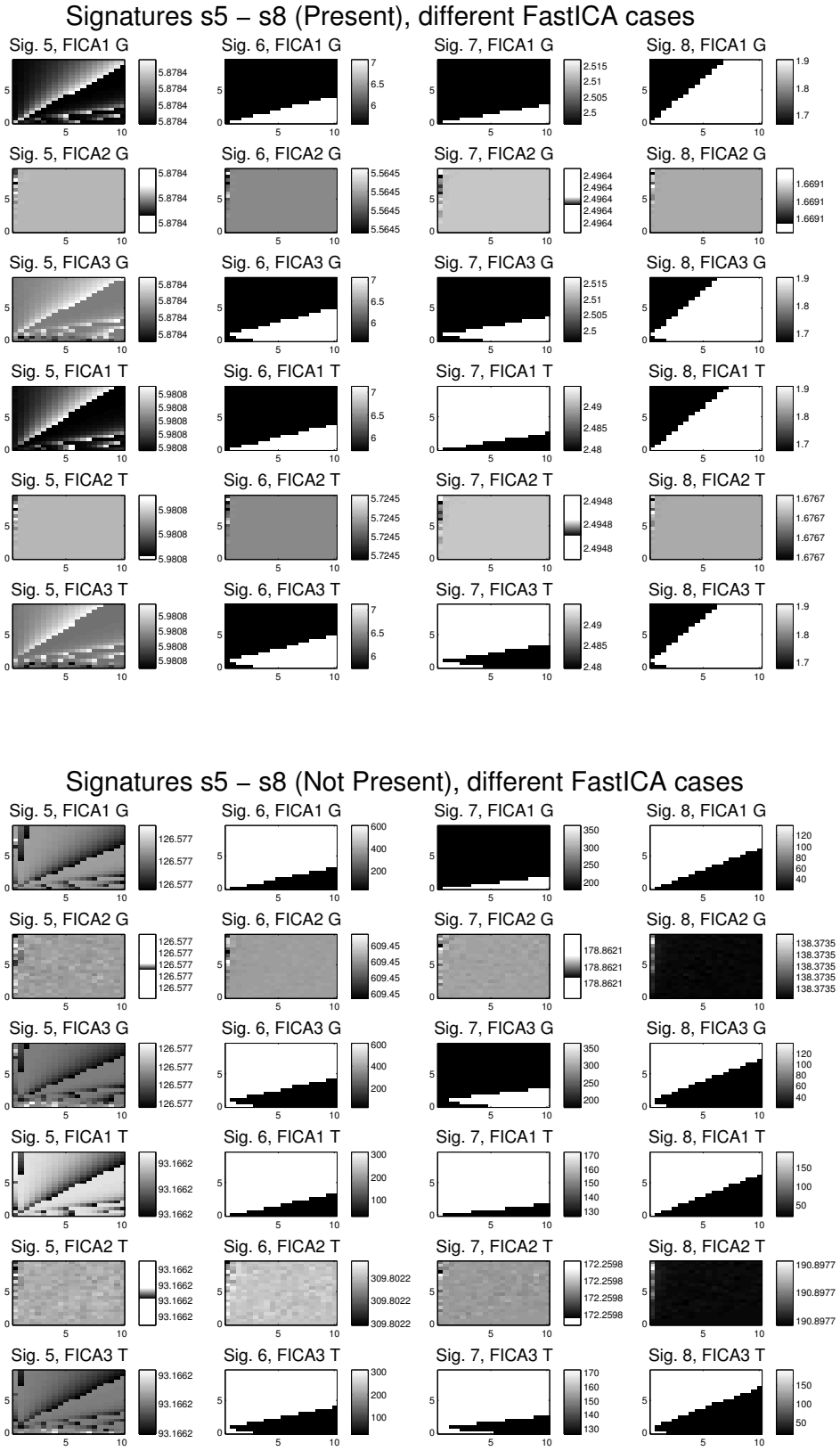



Figure A.34: Sought-after signatures $s_5 - s_8$.

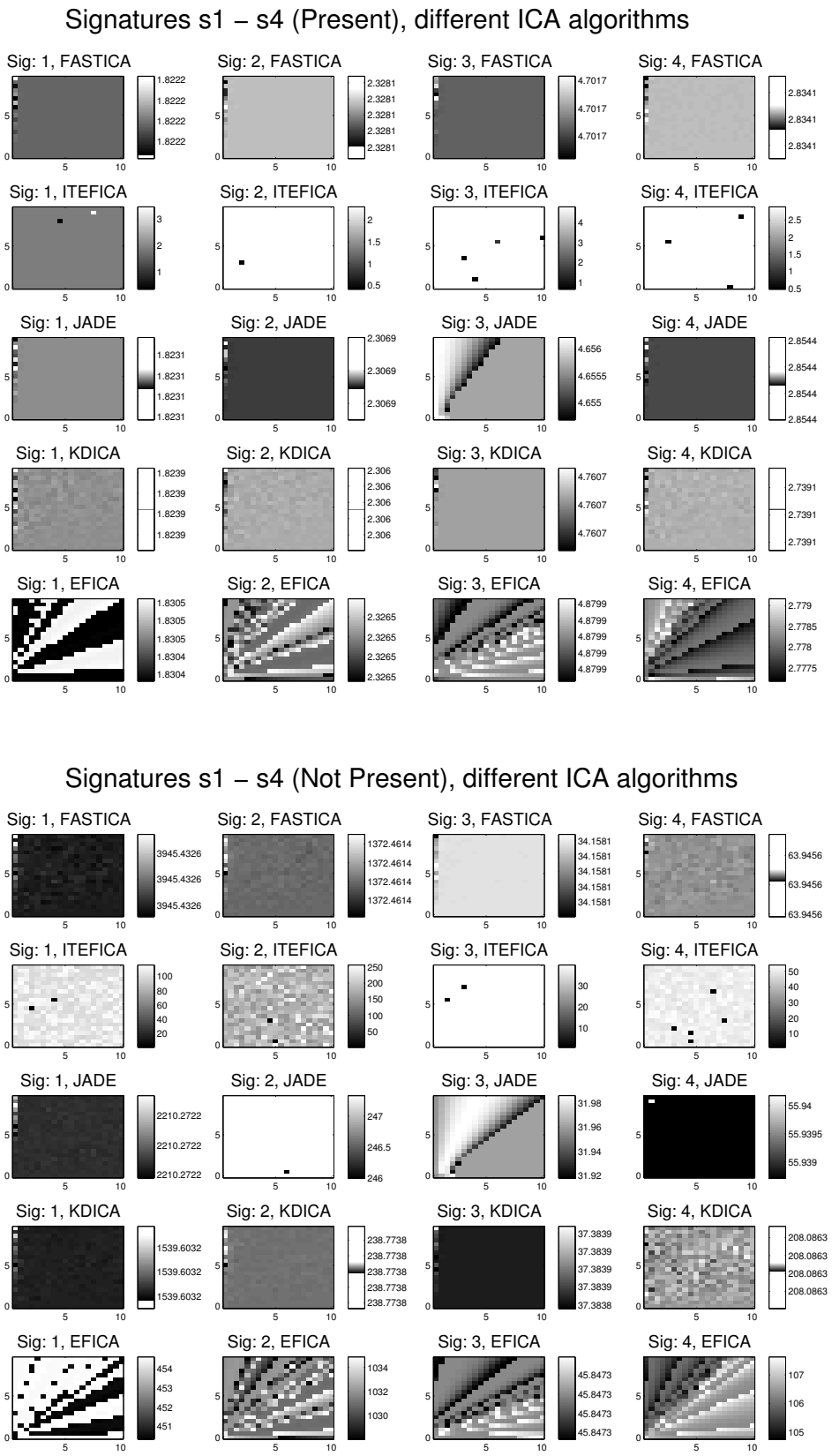



Figure A.35: Sought-after signatures $s_1 - s_4$.

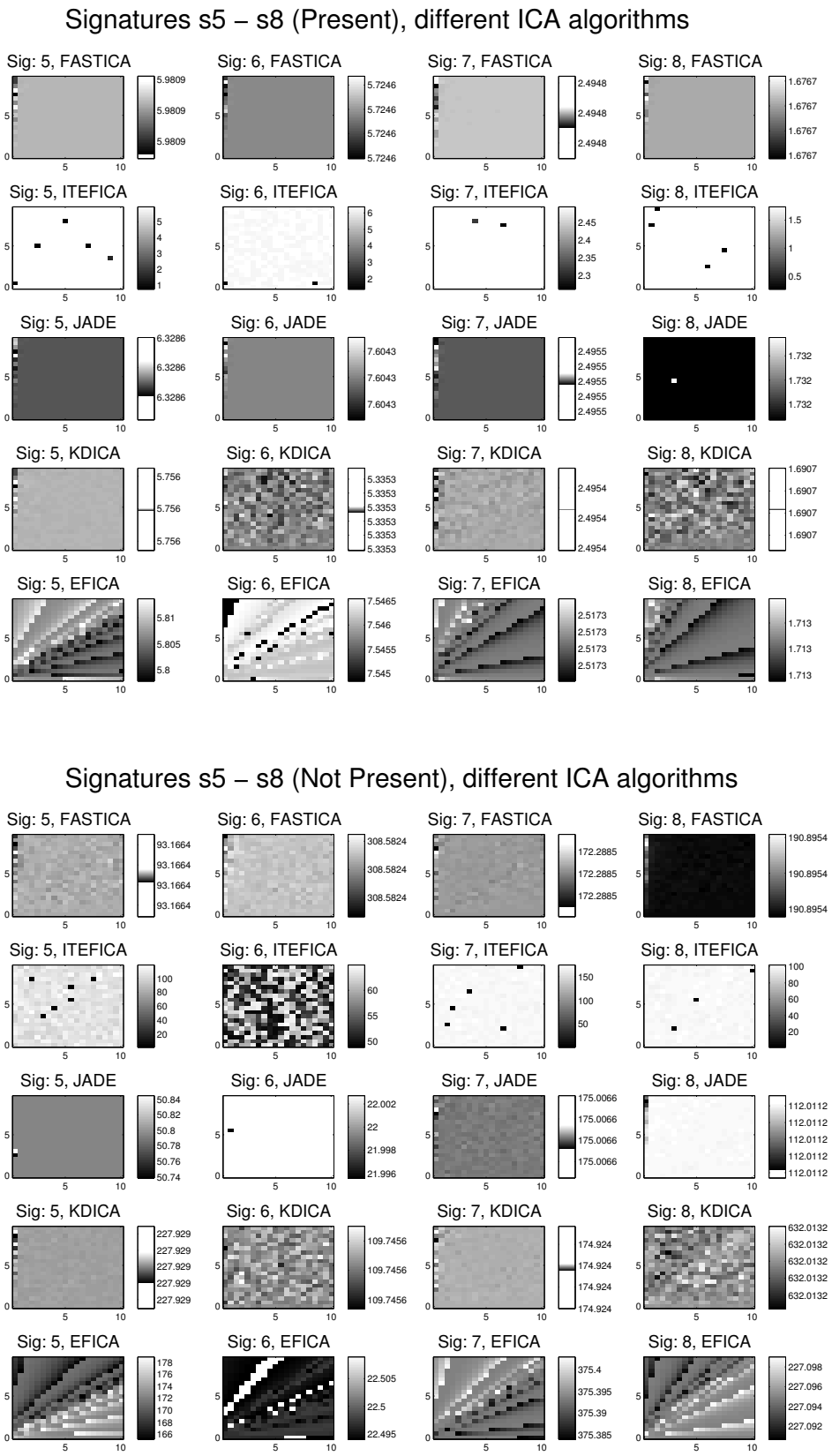



Figure A.36: Sought-after signatures $s_5 - s_8$.

## A.4 GUSSS Ratio vs. % of Signature Present in Mixtures (Sec. 5.3.4)

In this section we show figures as described in Section 5.3.4, for all 8 signatures, using FastICA and other ICA algorithms. Figures A.37 through A.52 correspond to zero noise ($\sigma = 0$). Figures A.53 through A.68 correspond to noise level $\sigma = 0.01$; Figures A.69 through A.84, to noise level $\sigma = 0.05$; and Figures A.85 through A.100, to noise level $\sigma = 0.1$.

Every Figure shows plots for four combinations of weights: $w_1 = 1$, $w_p = 1$; $w_1 = 1$, $w_p = 10$;$w_1 = 10$, $w_p = 1$; and $w_1 = 10$, $w_p = 10$. Ratios corresponding to $c_p \in [-0.01, 0.1]$ are shown in logarithmic scale (to the left), due to the large ratio values observed for $c_p$ close to zero. Ratios corresponding to $c_p \in [0.1, 1]$ are shown in linear scale (to the right).

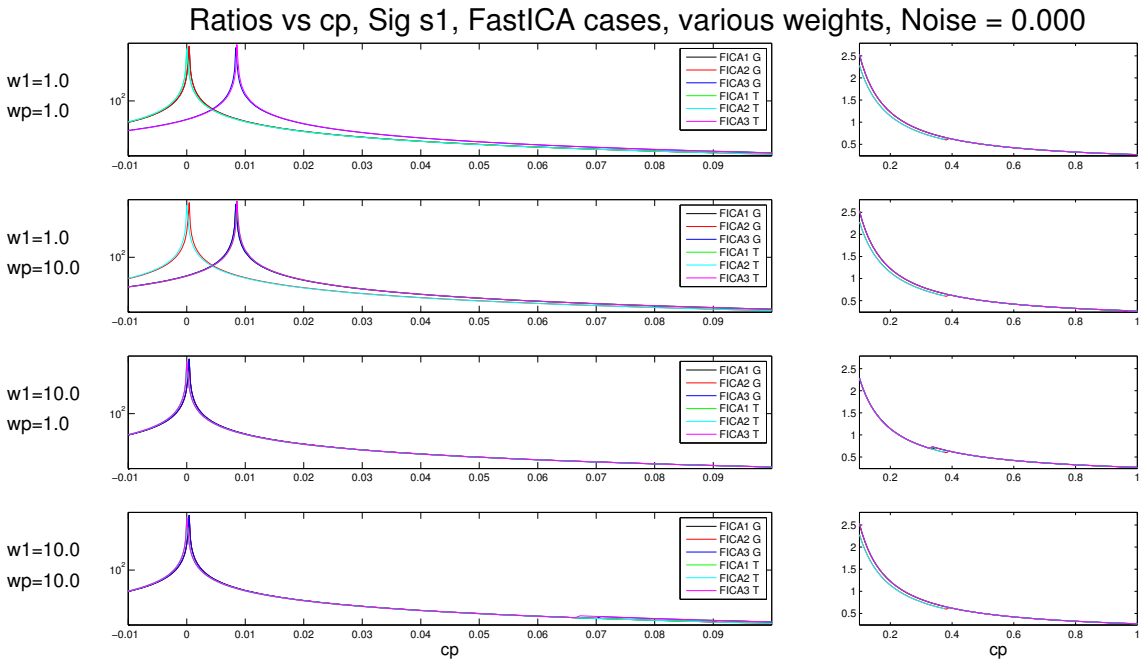



Figure A.37: Ratios vs. $c_p$. FastICA cases. Signature $s_1$. No noise added to the mixtures.

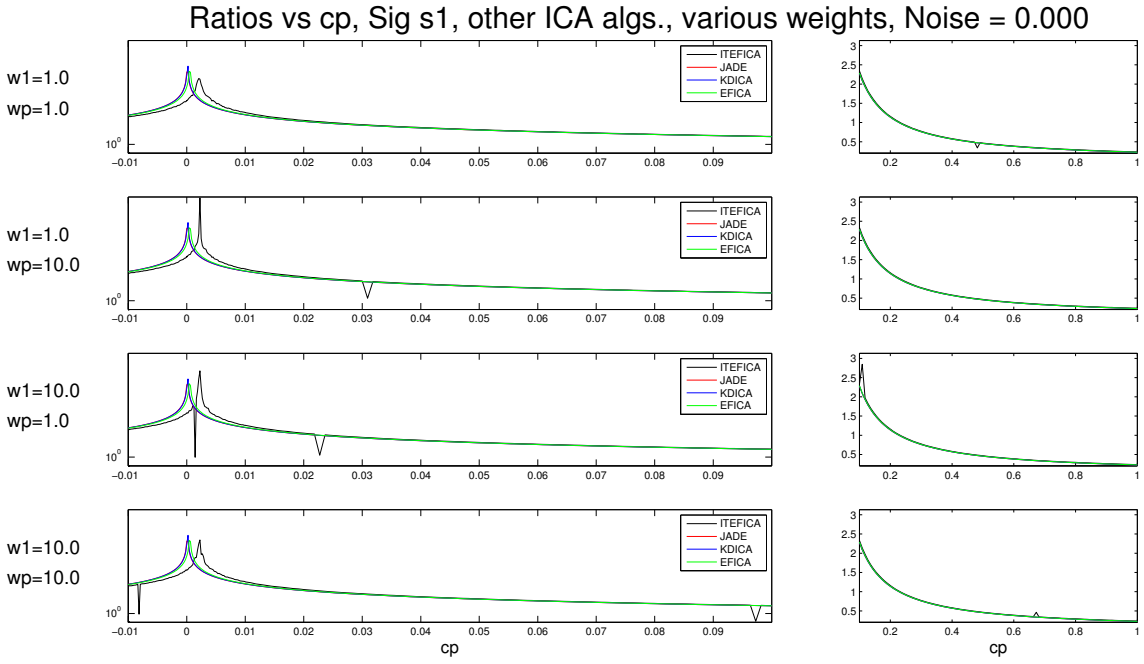



Figure A.38: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_1$. No noise added to the mixtures.

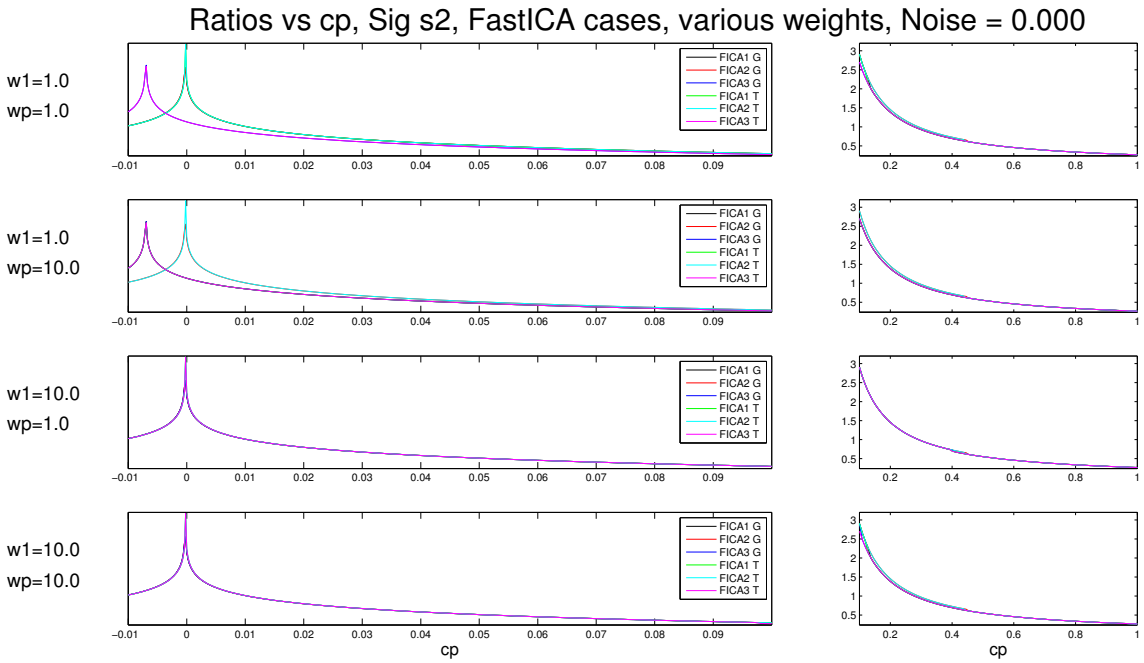



Figure A.39: Ratios vs. $c_p$. FastICA cases. Signature $s_2$. No noise added to the mixtures.

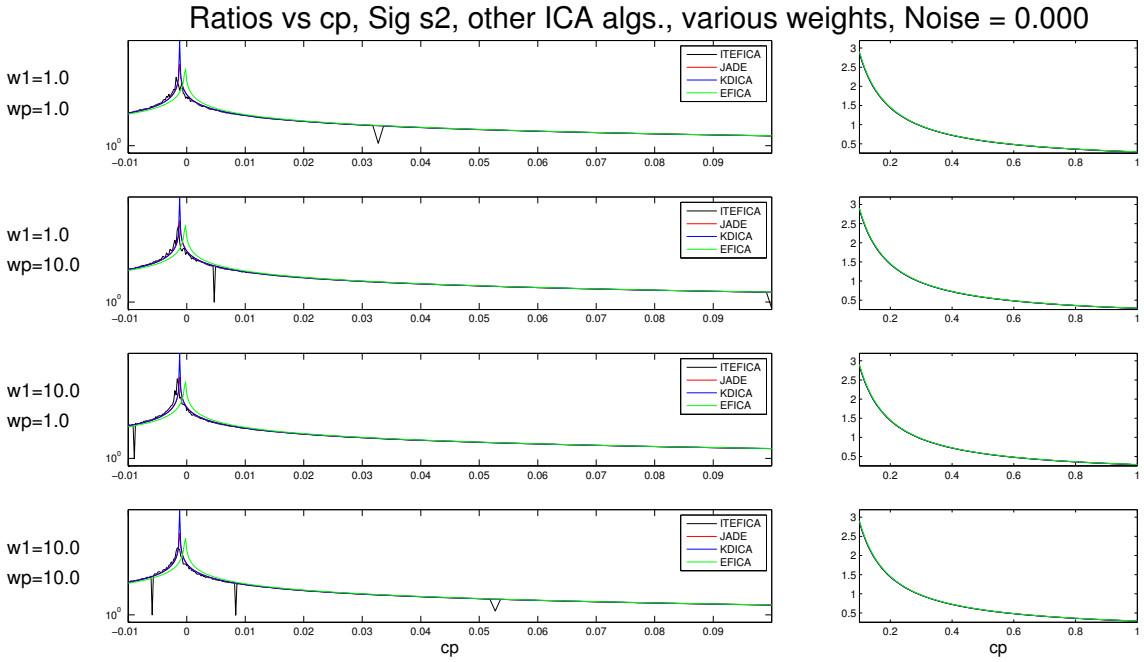



Figure A.40: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_2$. No noise added to the mixtures.

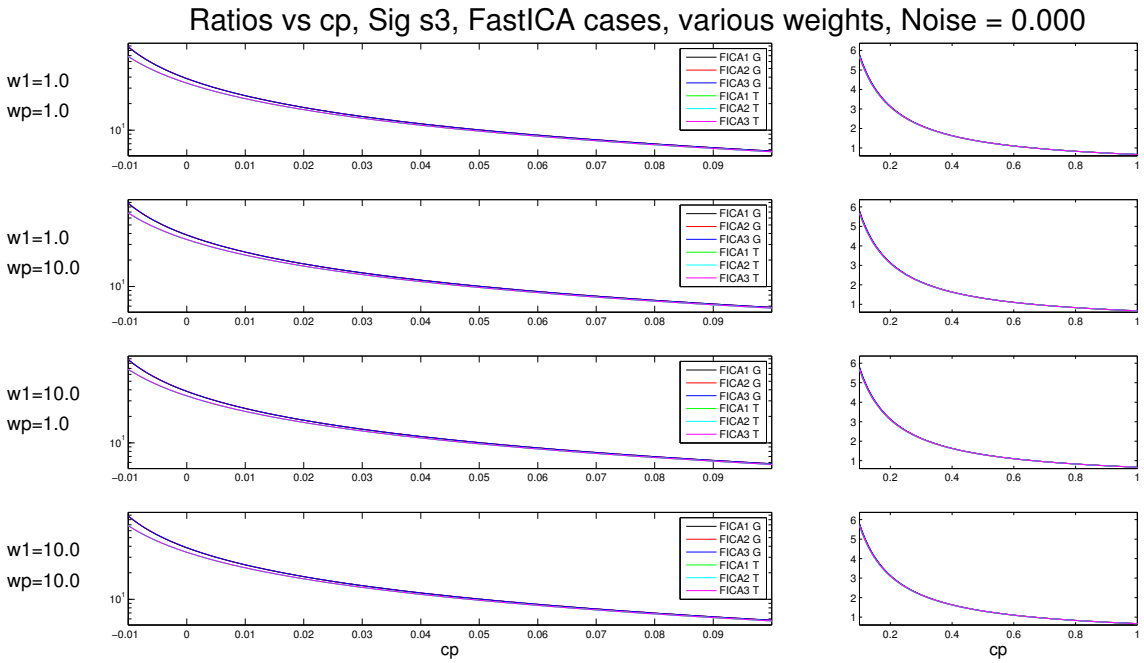



Figure A.41: Ratios vs. $c_p$. FastICA cases. Signature $s_3$. No noise added to the mixtures.

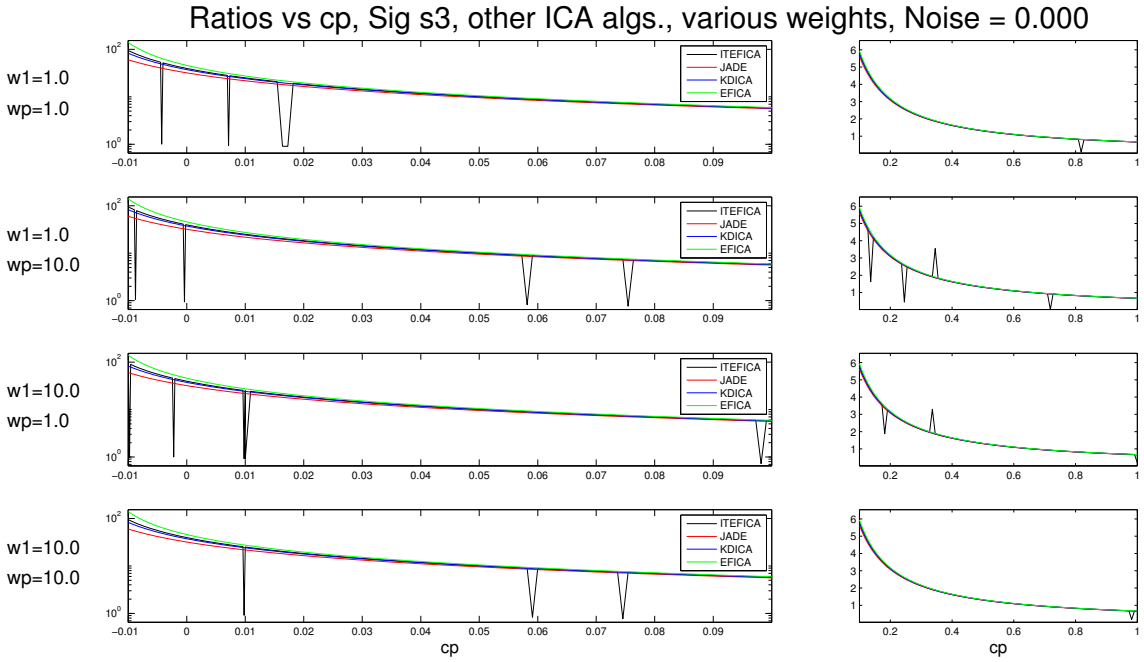



Figure A.42: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_3$. No noise added to the mixtures.

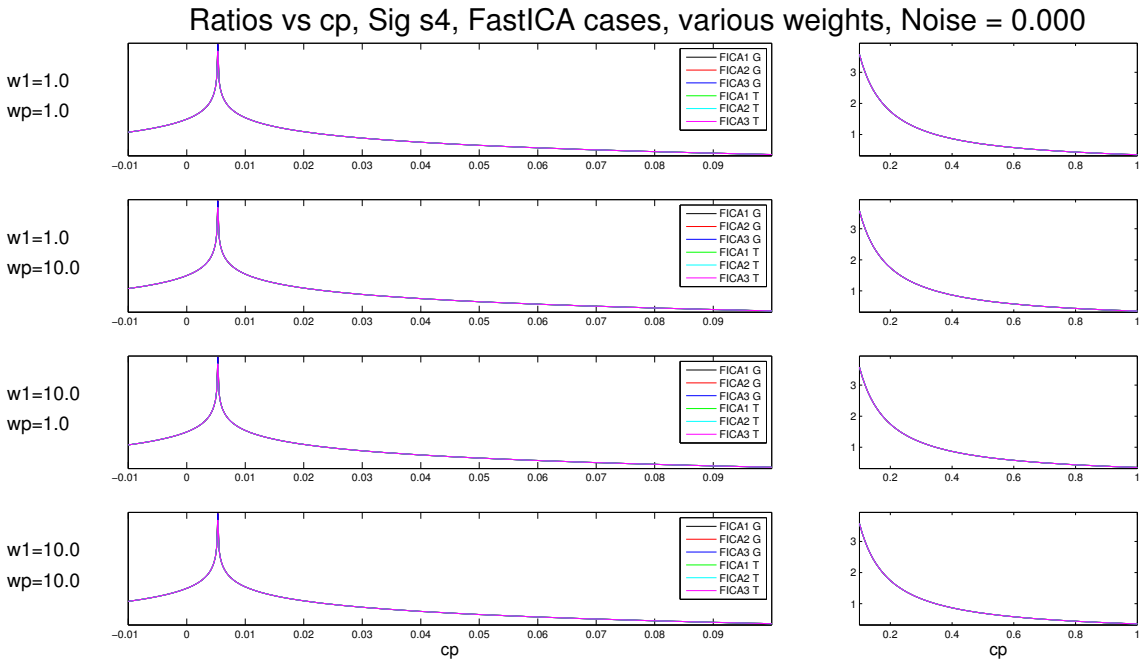



Figure A.43: Ratios vs. $c_p$. FastICA cases. Signature $s_4$. No noise added to the mixtures.

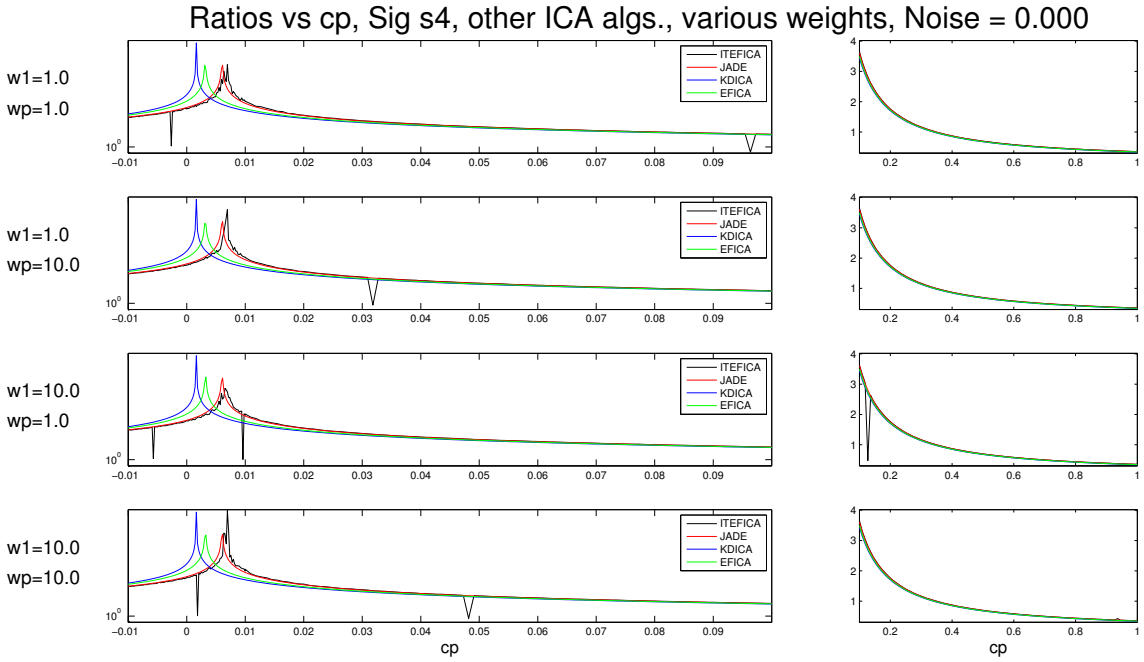



Figure A.44: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_4$. No noise added to the mixtures.

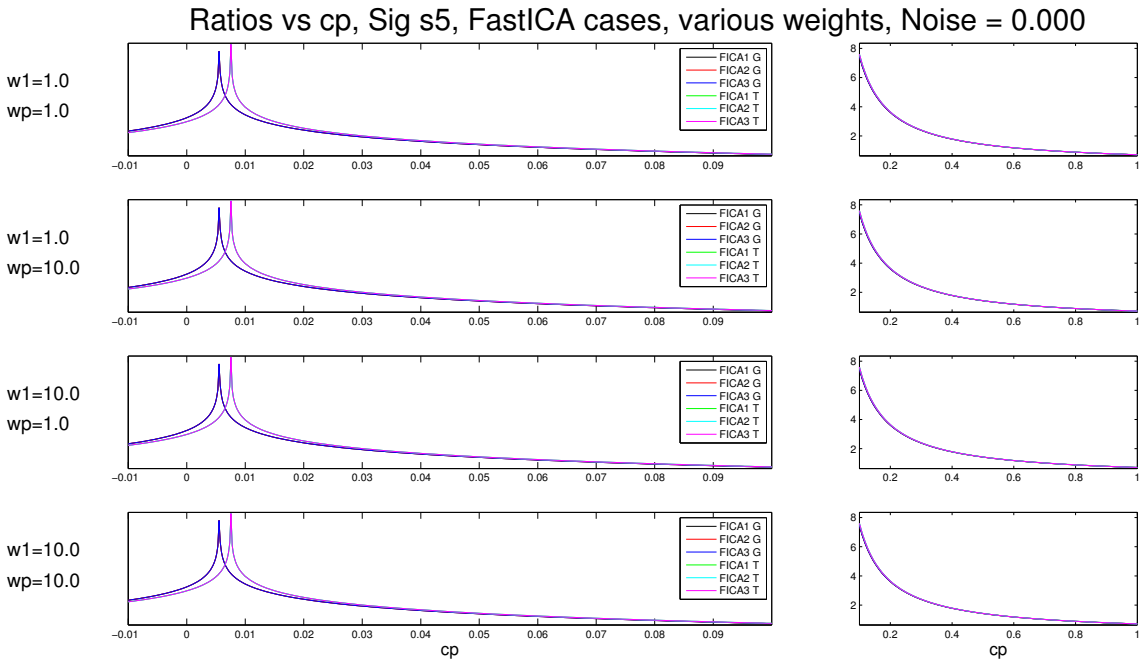



Figure A.45: Ratios vs. $c_p$. FastICA cases. Signature $s_5$. No noise added to the mixtures.

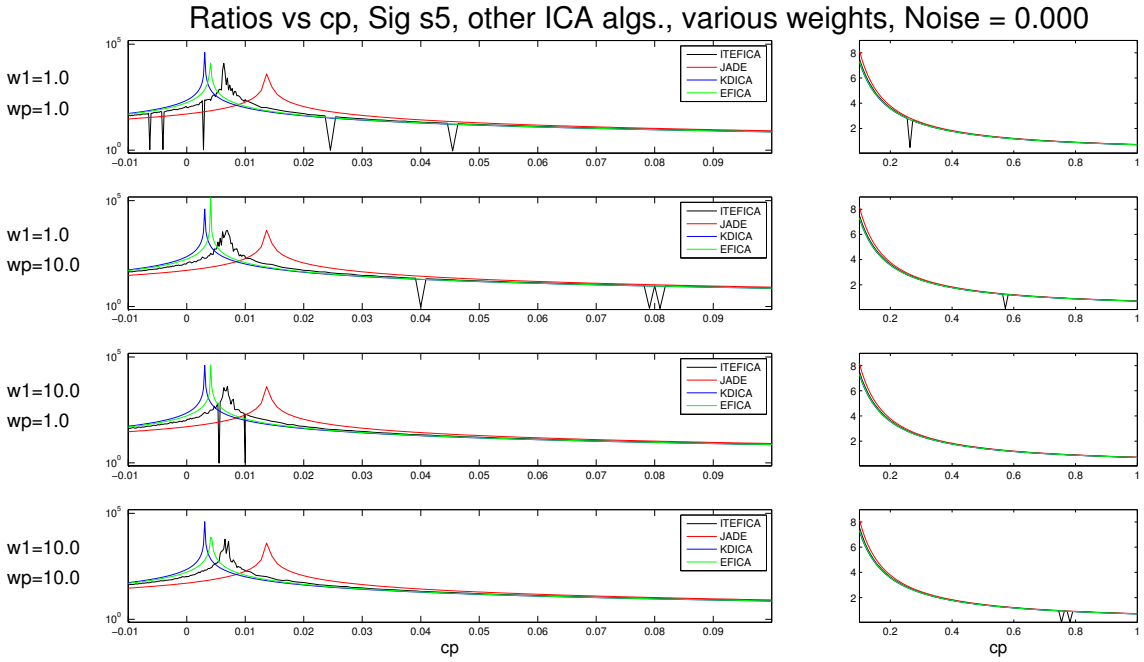



Figure A.46: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_5$. No noise added to the mixtures.

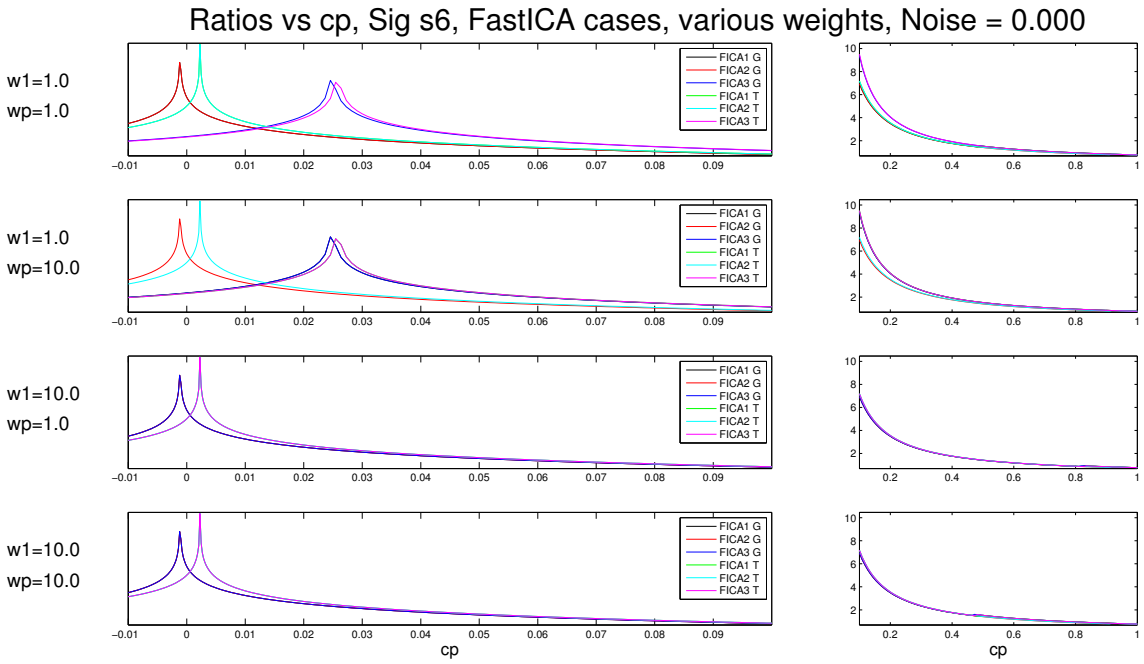



Figure A.47: Ratios vs. $c_p$. FastICA cases. Signature $s_6$. No noise added to the mixtures.

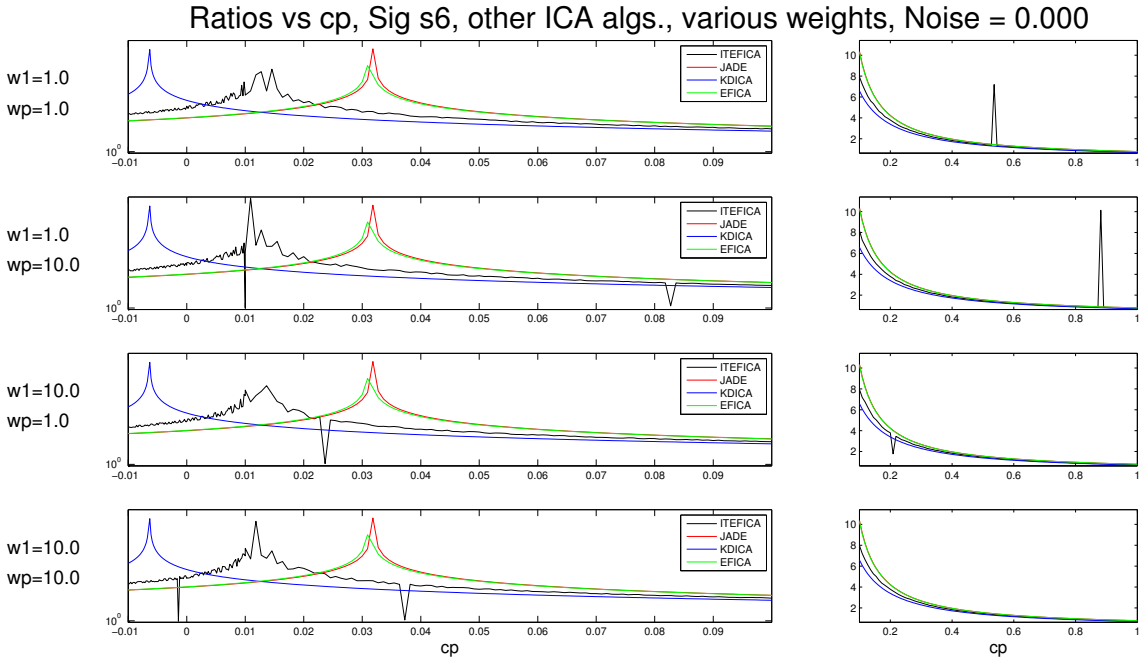



Figure A.48: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_6$. No noise added to the mixtures.

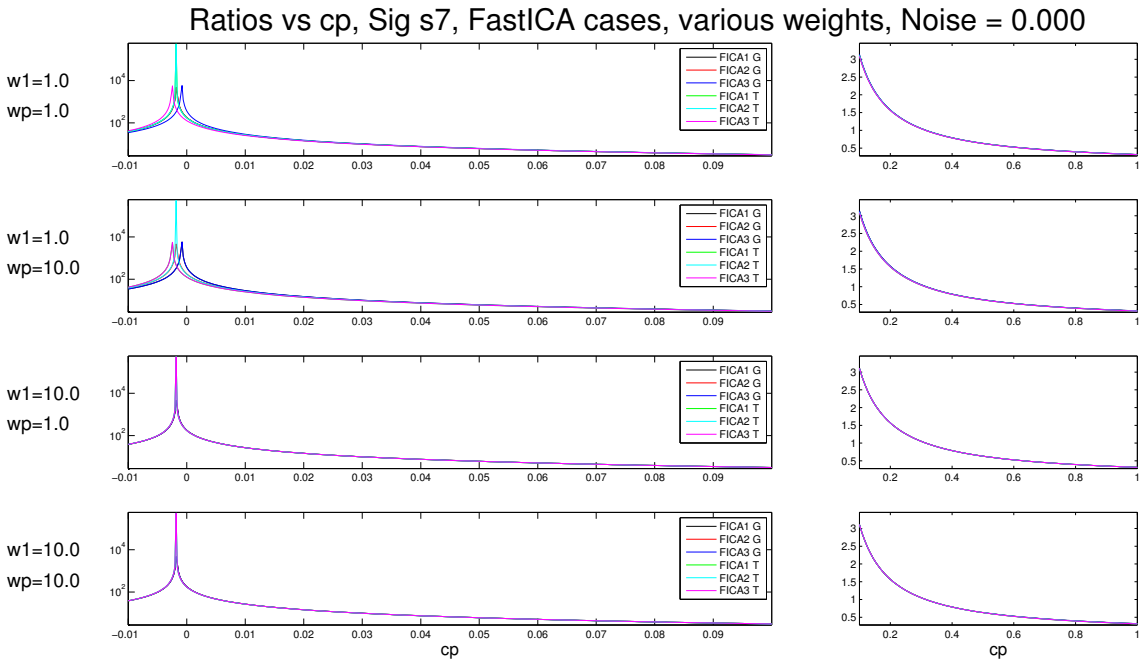



Figure A.49: Ratios vs. $c_p$. FastICA cases. Signature $s_7$. No noise added to the mixtures.

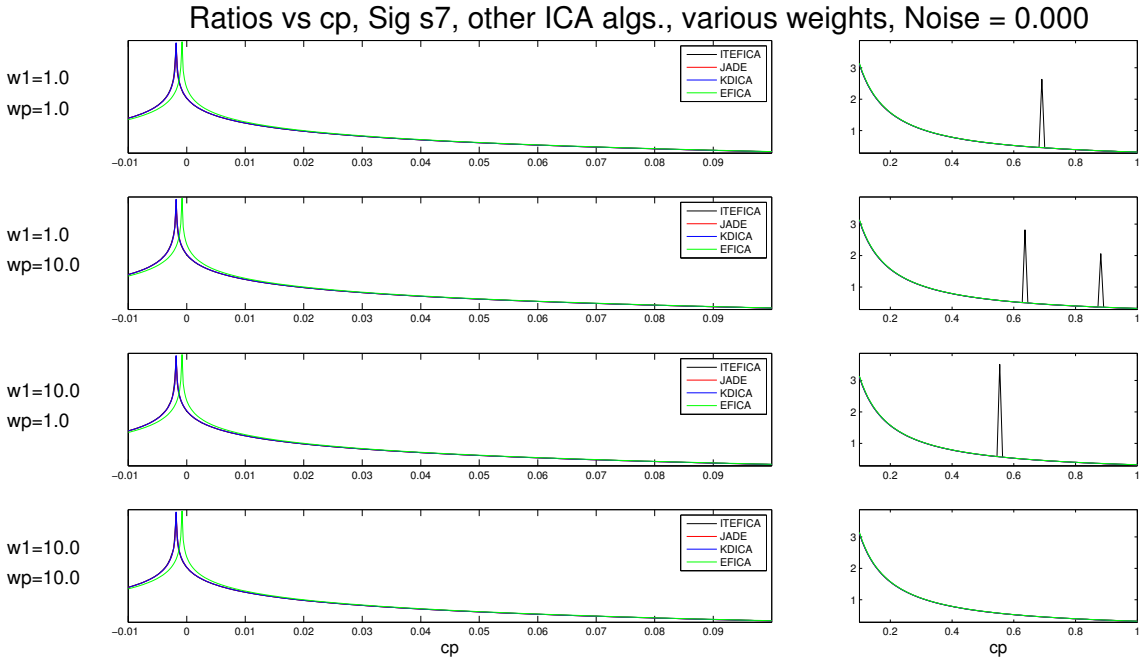



Figure A.50: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_7$. No noise added to the mixtures.

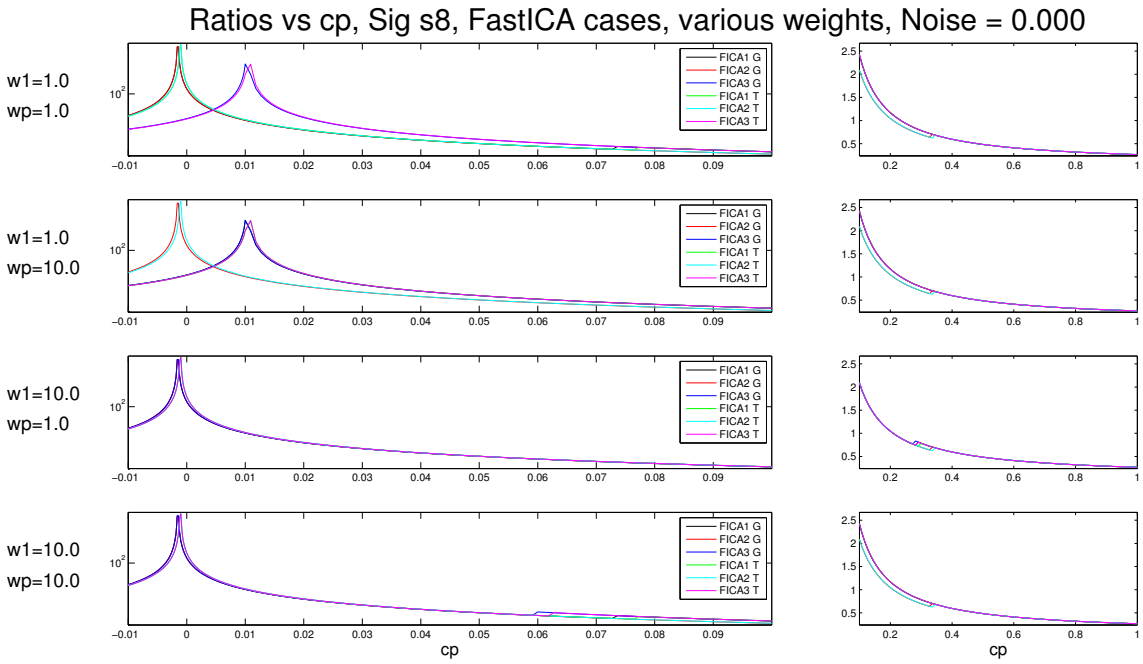



Figure A.51: Ratios vs. $c_p$. FastICA cases. Signature $s_8$. No noise added to the mixtures.

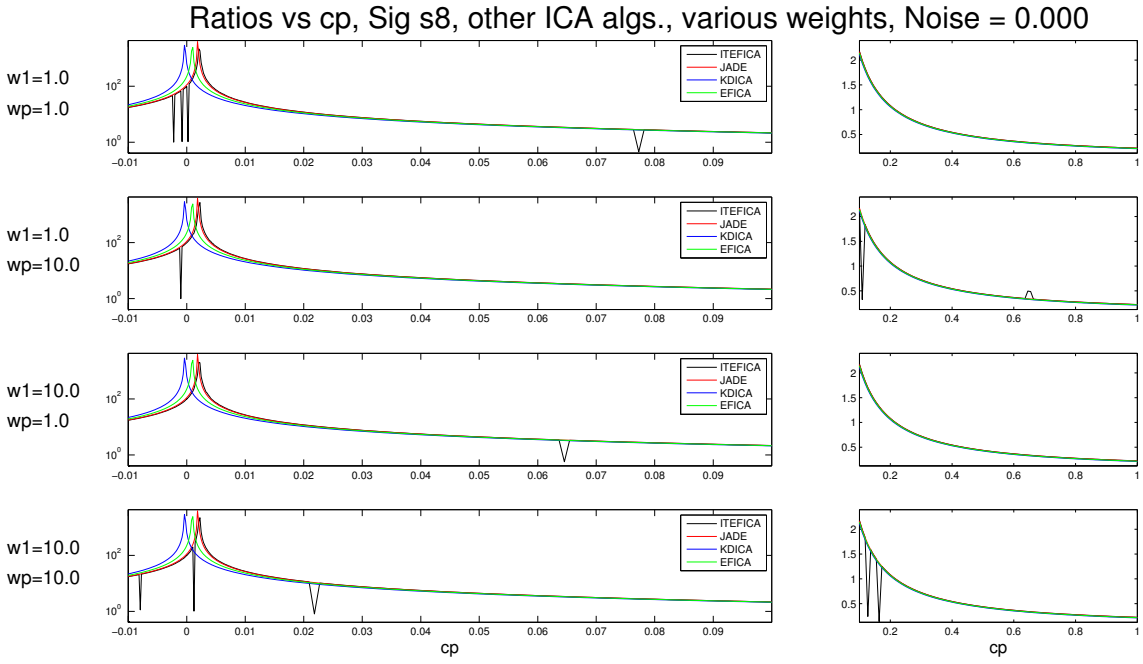



Figure A.52: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_8$. No noise added to the mixtures.

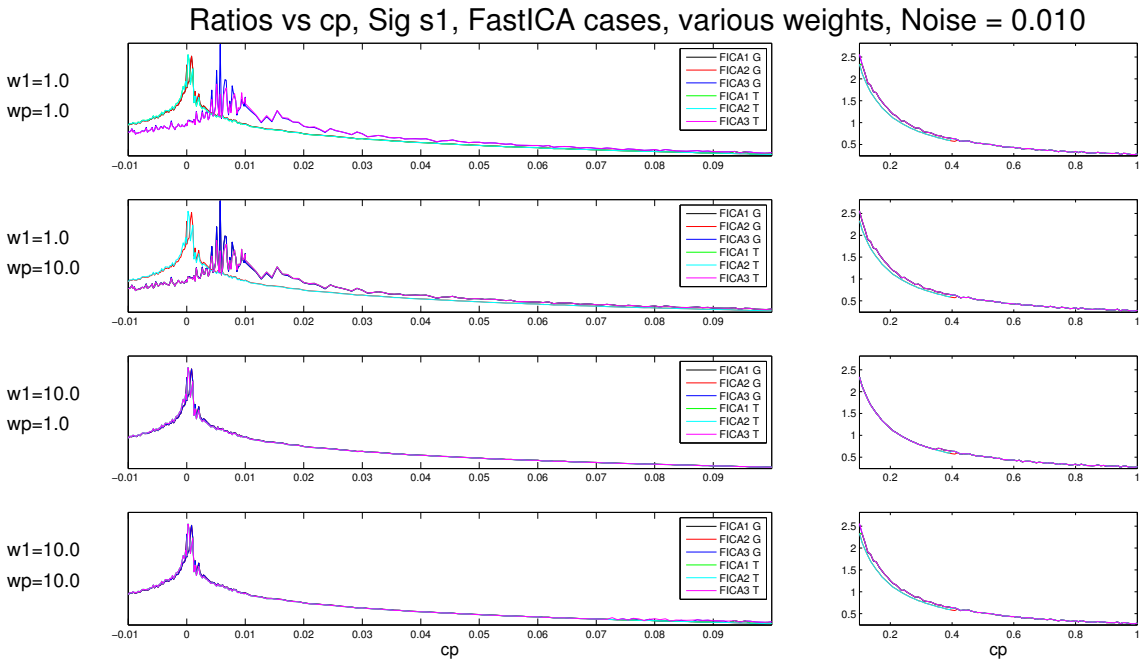



Figure A.53: Ratios vs. $c_p$. FastICA cases. Signature $s_1$. Noise added to the mixtures ($\sigma = 0.01$).

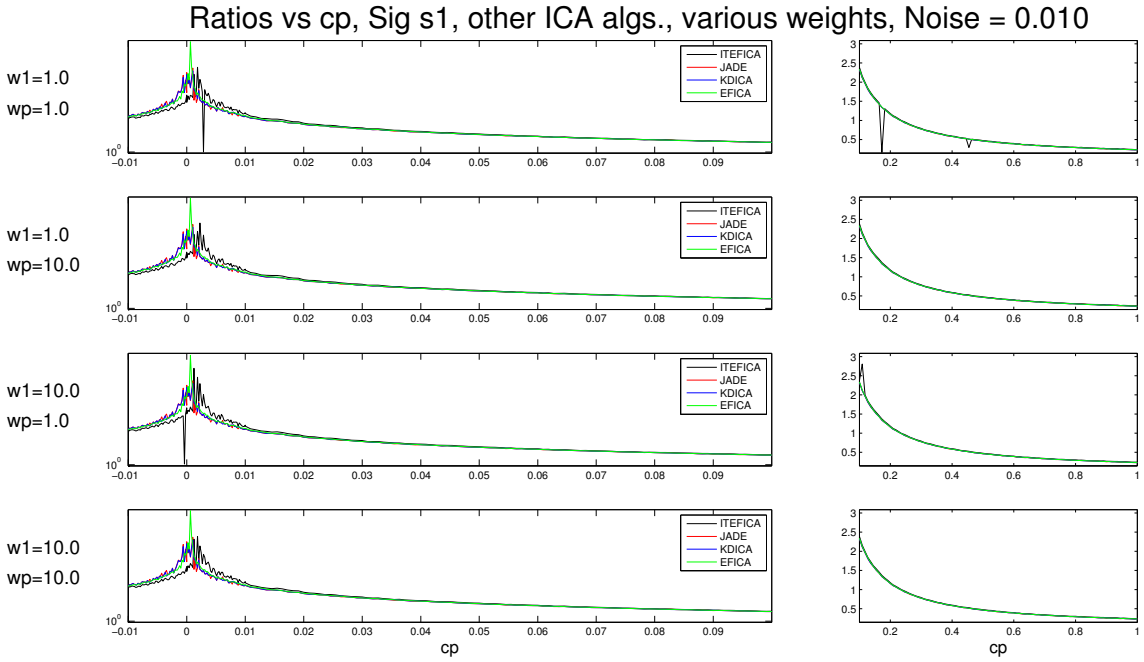



Figure A.54: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_1$. Noise added to the mixtures ($\sigma = 0.01$).

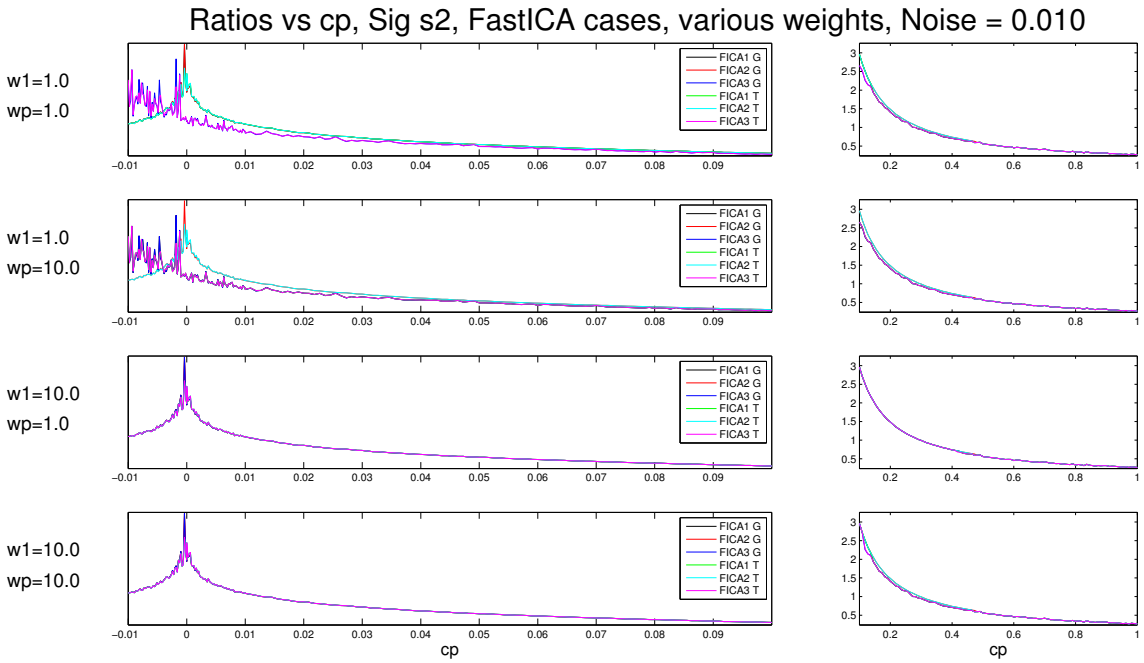



Figure A.55: Ratios vs. $c_p$. FastICA cases. Signature $s_2$. Noise added to the mixtures ($\sigma = 0.01$).

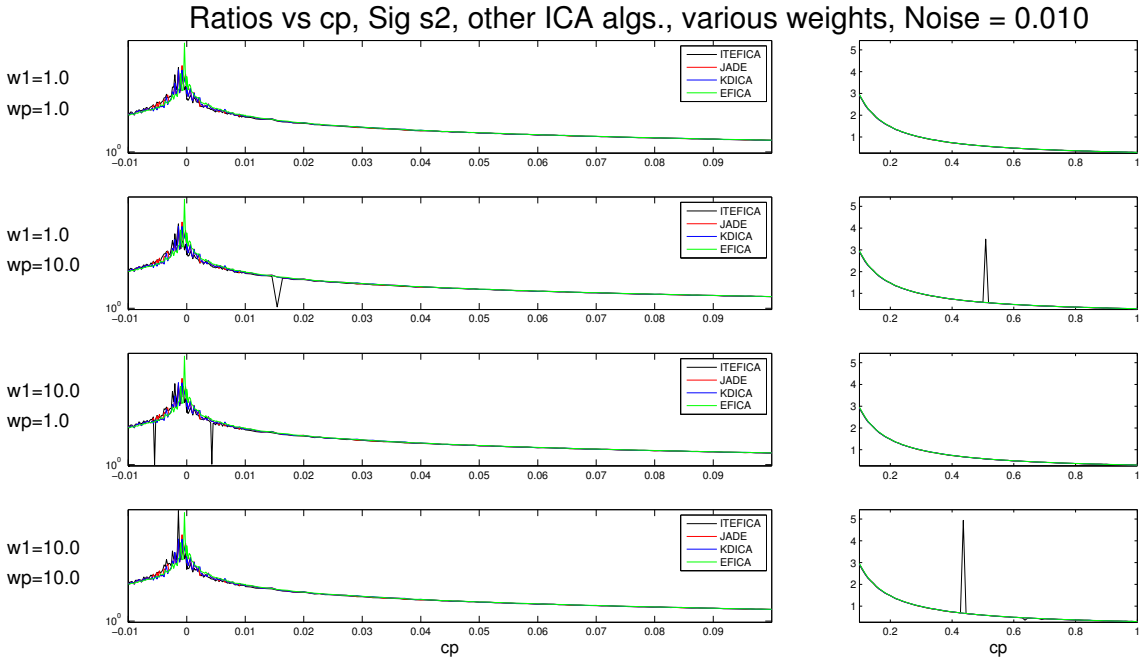



Figure A.56: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_2$. Noise added to the mixtures ($\sigma = 0.01$).

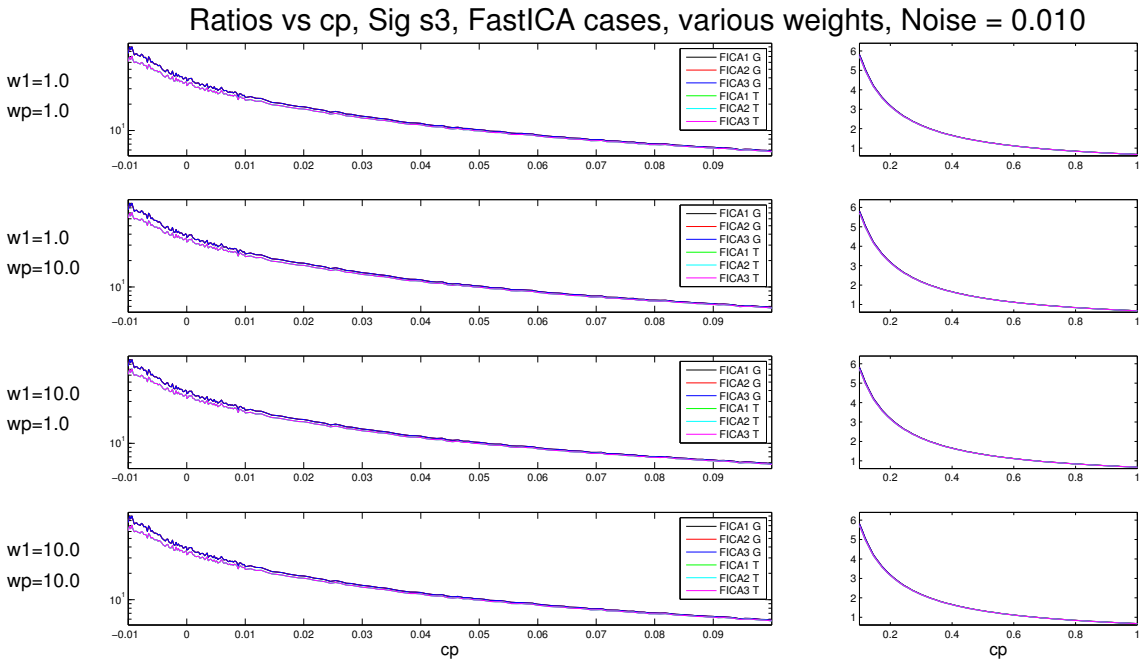



Figure A.57: Ratios vs. $c_p$. FastICA cases. Signature $s_3$. Noise added to the mixtures ($\sigma = 0.01$).

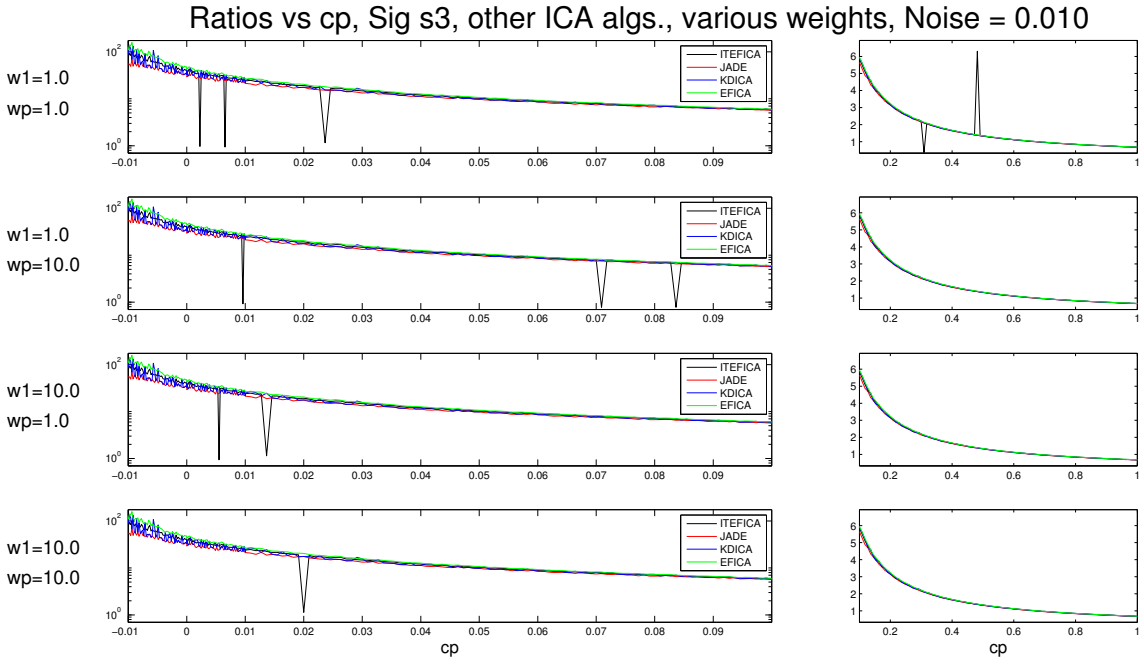



Figure A.58: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_3$. Noise added to the mixtures ($\sigma = 0.01$).

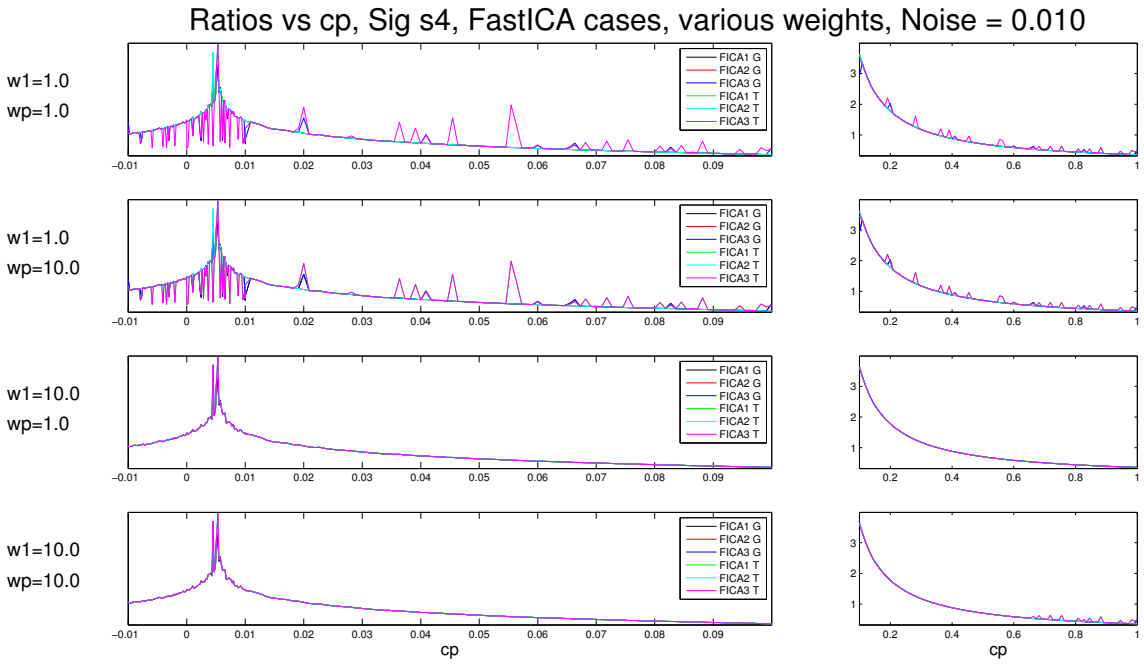



Figure A.59: Ratios vs. $c_p$. FastICA cases. Signature $s_4$. Noise added to the mixtures ($\sigma = 0.01$).

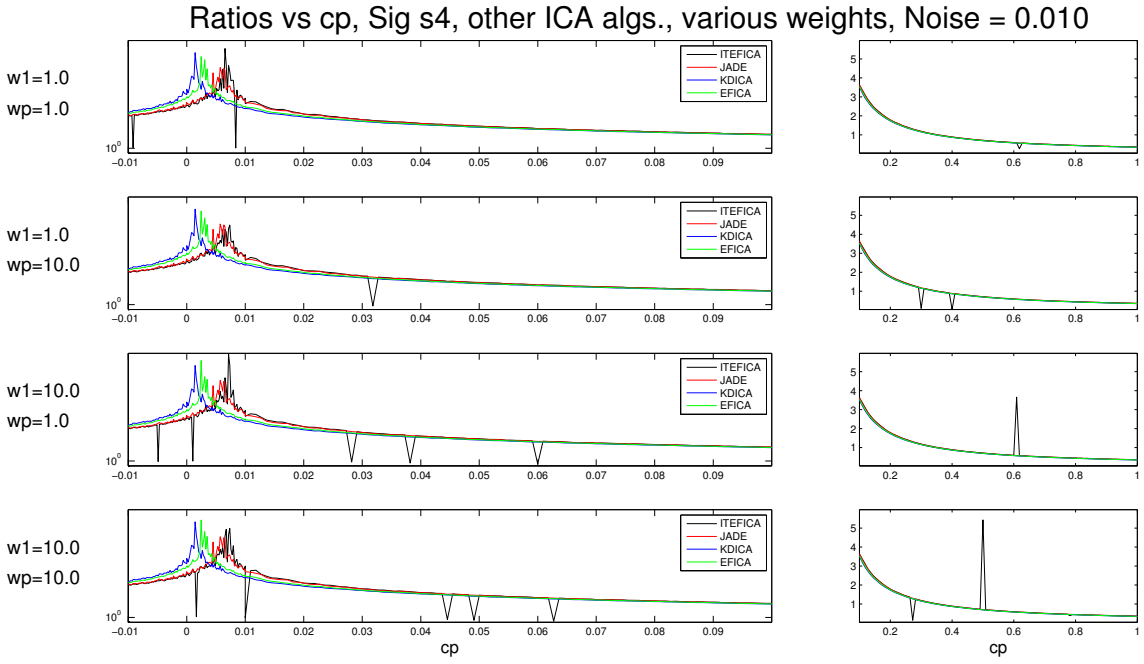



Figure A.60: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_4$. Noise added to the mixtures ($\sigma = 0.01$).

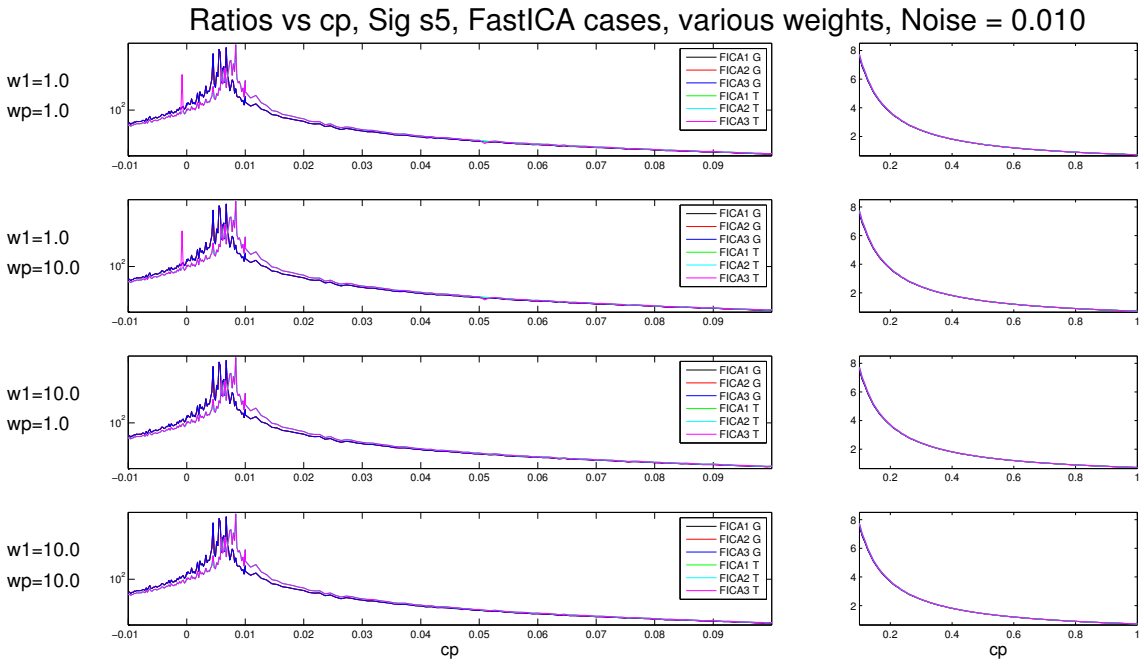



Figure A.61: Ratios vs. $c_p$. FastICA cases. Signature $s_5$. Noise added to the mixtures ($\sigma = 0.01$).

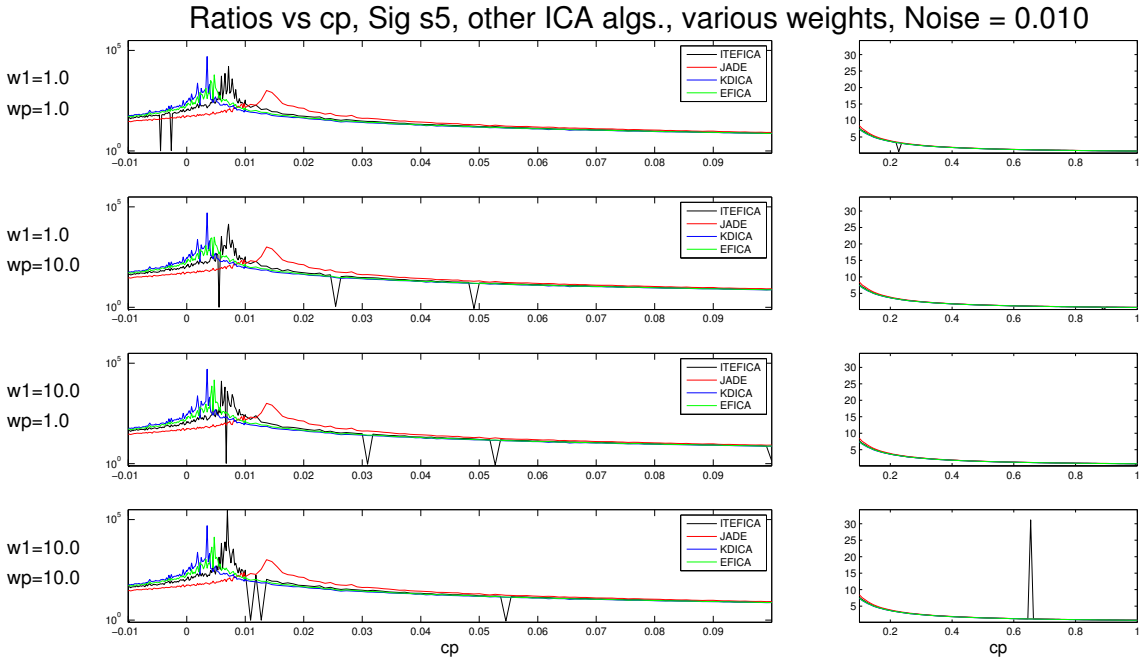



Figure A.62: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_5$. Noise added to the mixtures ($\sigma = 0.01$).

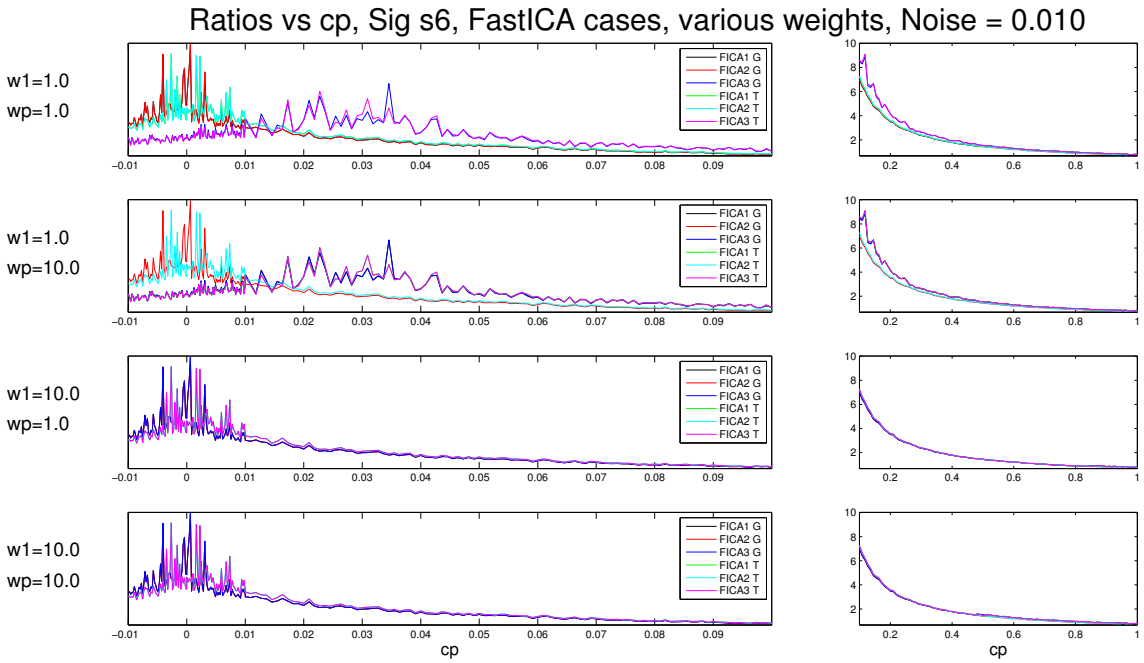



Figure A.63: Ratios vs. $c_p$. FastICA cases. Signature $s_6$. Noise added to the mixtures ($\sigma = 0.01$).

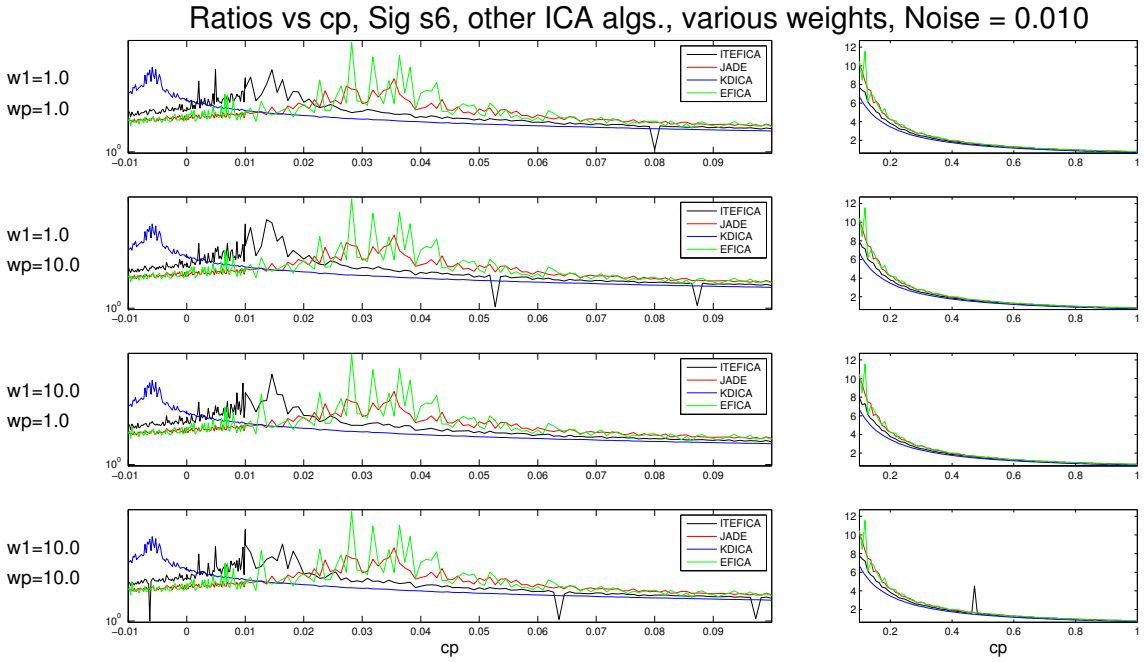



Figure A.64: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_6$. Noise added to the mixtures ($\sigma = 0.01$).

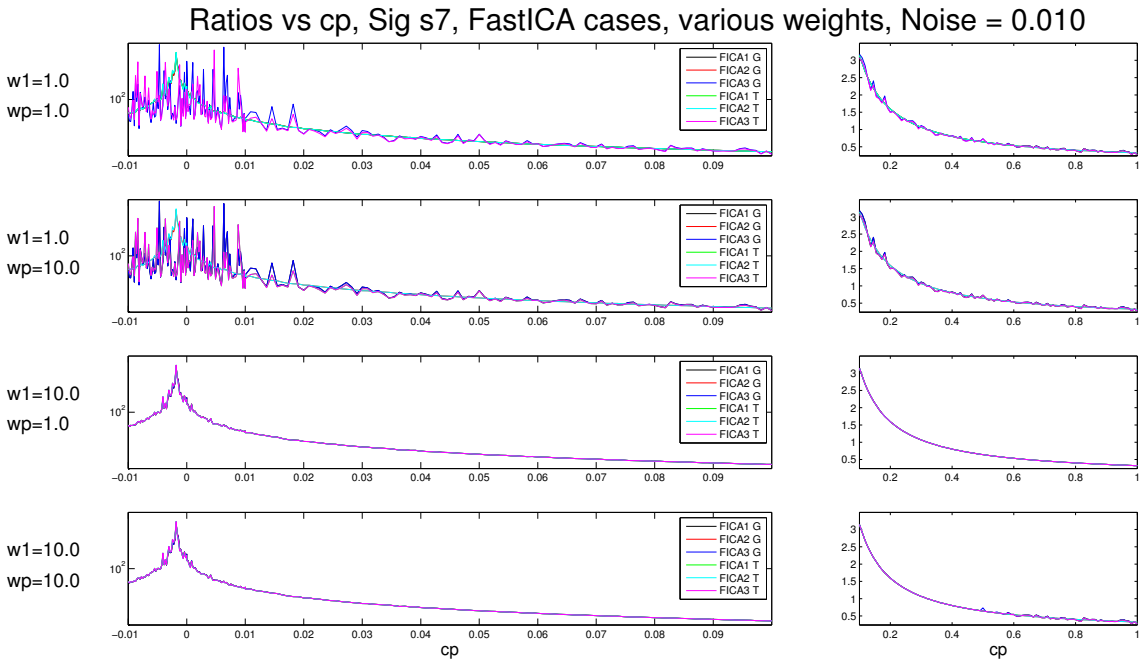



Figure A.65: Ratios vs. $c_p$. FastICA cases. Signature $s_7$. Noise added to the mixtures ($\sigma = 0.01$).

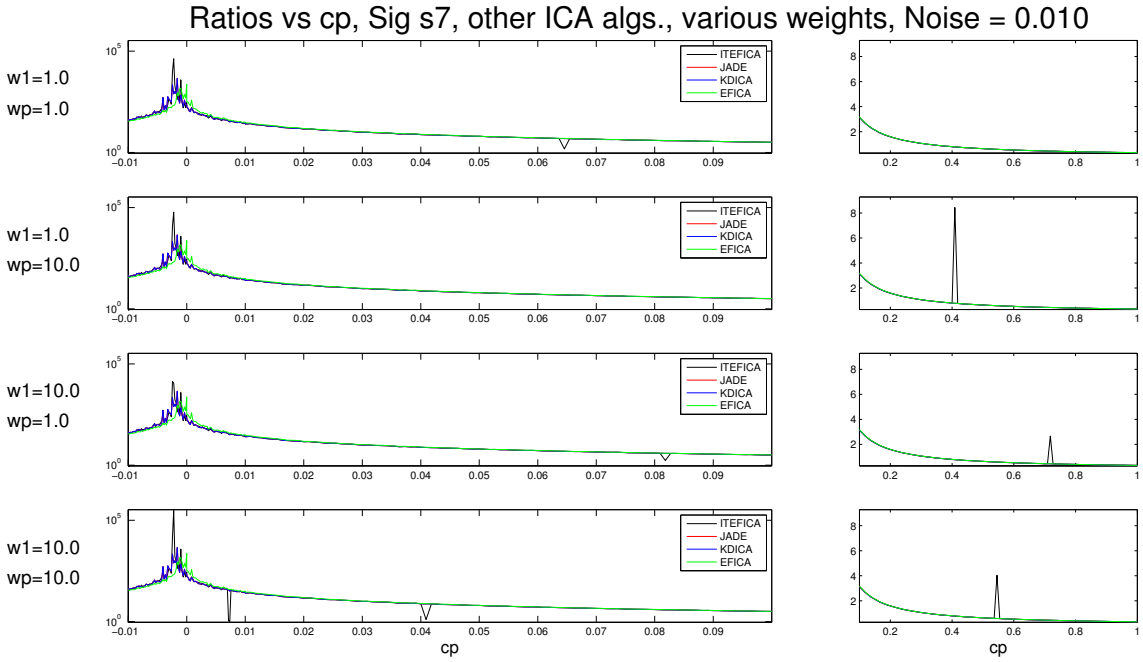



Figure A.66: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_7$. Noise added to the mixtures ($\sigma = 0.01$).

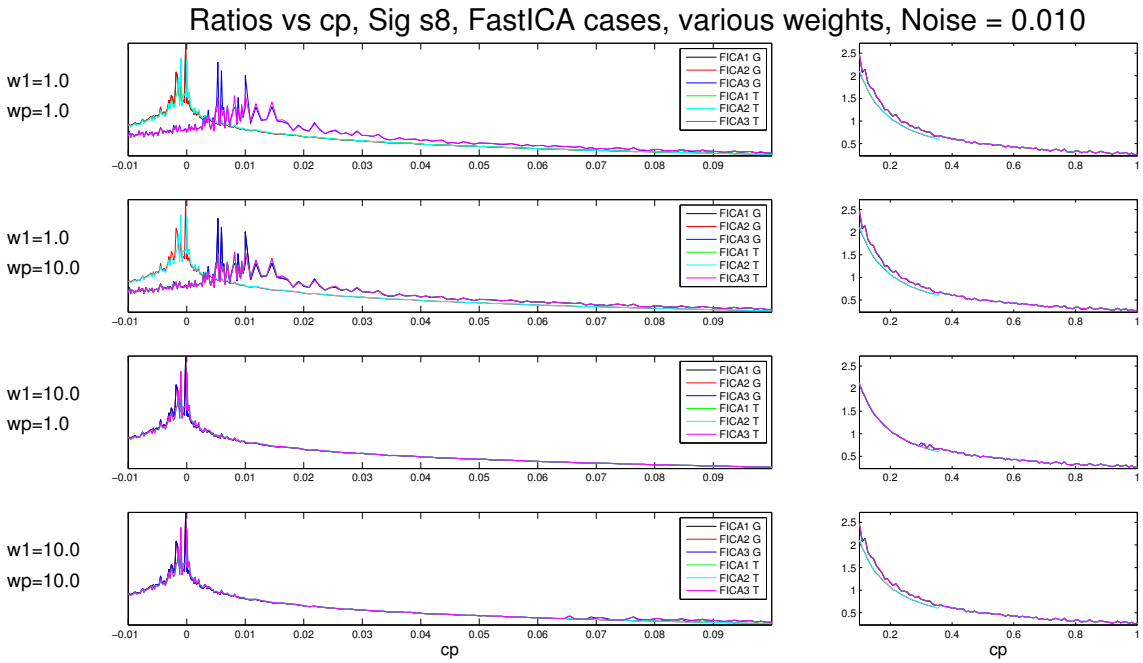



Figure A.67: Ratios vs. $c_p$. FastICA cases. Signature $s_8$. Noise added to the mixtures ($\sigma = 0.01$).

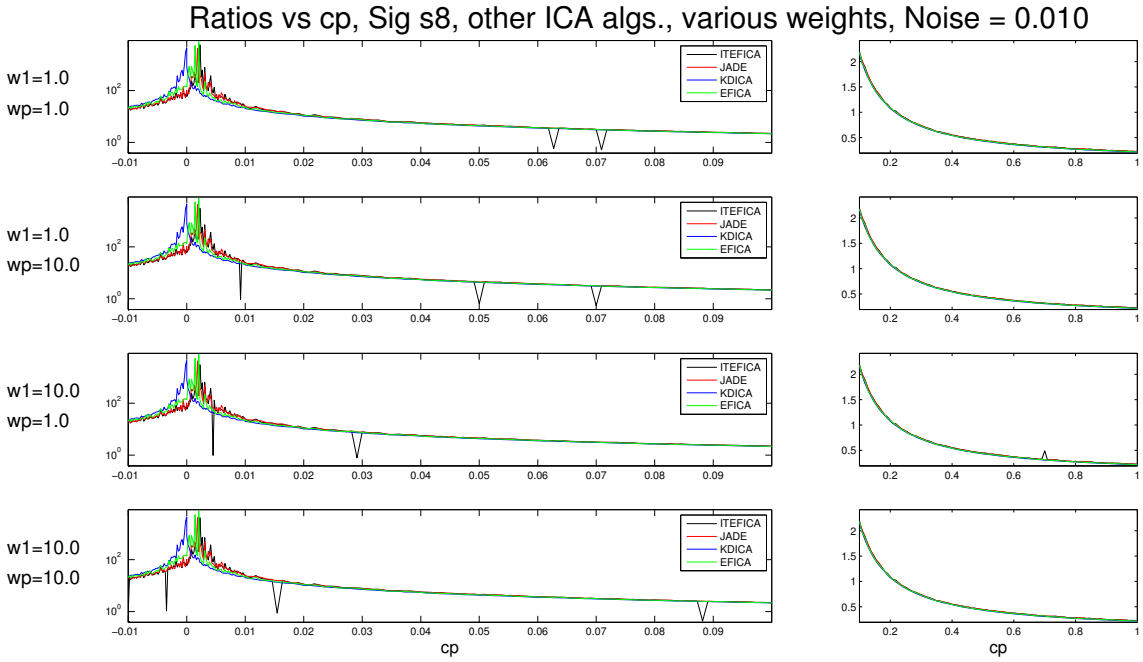



Figure A.68: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_8$. Noise added to the mixtures ($\sigma = 0.01$).

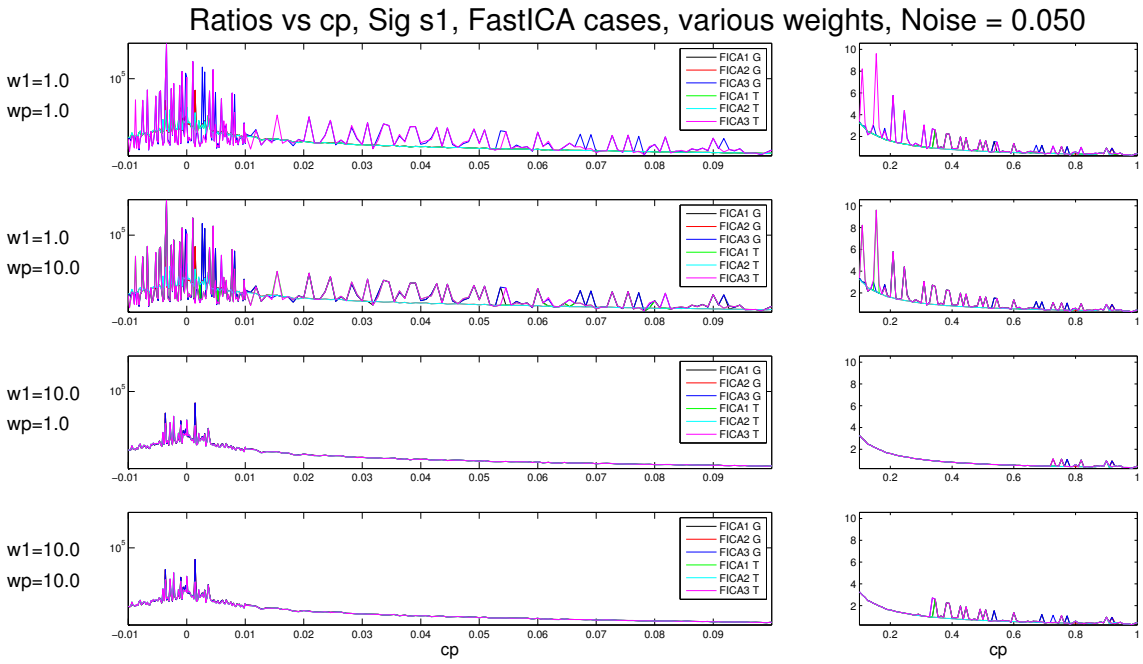



Figure A.69: Ratios vs. $c_p$. FastICA cases. Signature $s_1$. Noise added to the mixtures ($\sigma = 0.05$).

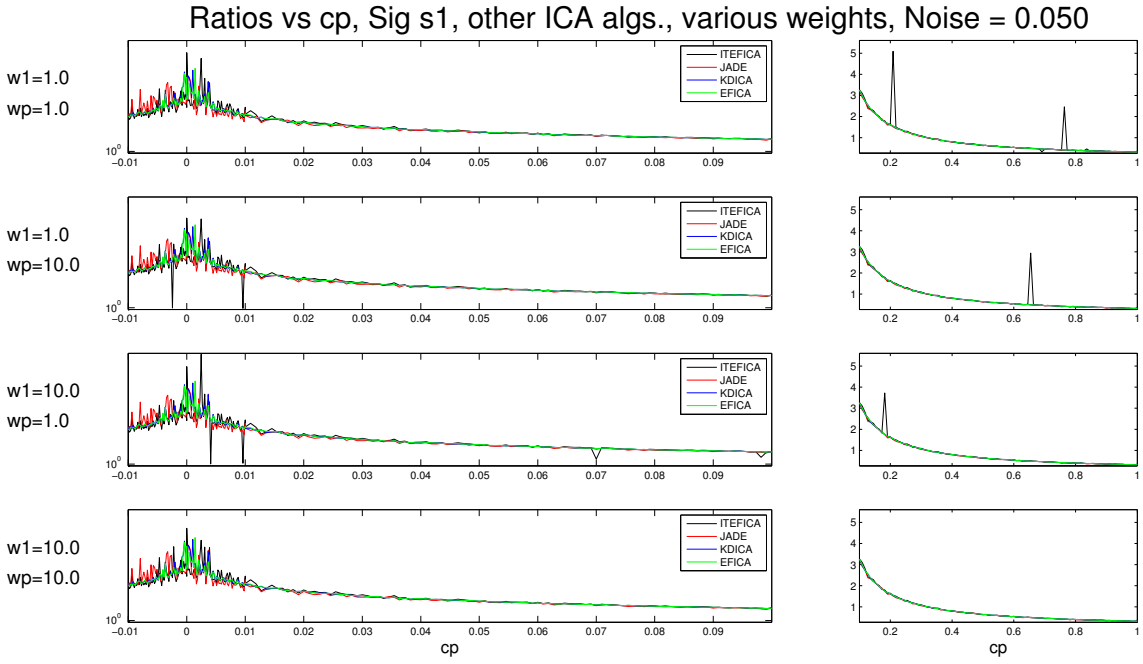



Figure A.70: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_1$. Noise added to the mixtures ($\sigma = 0.05$).

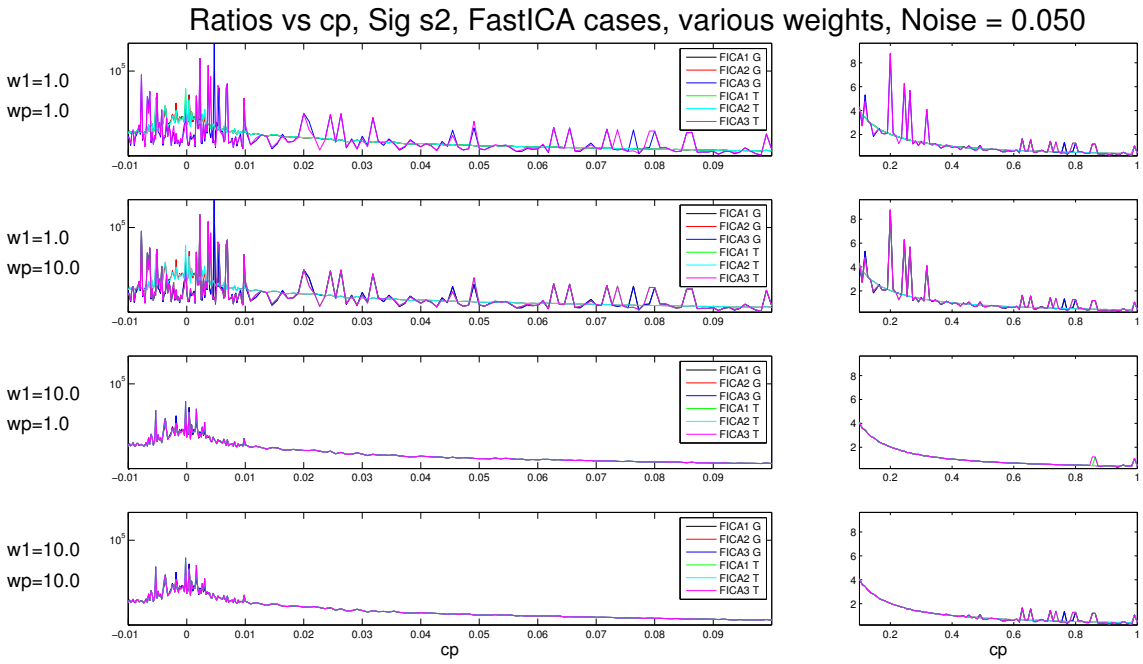



Figure A.71: Ratios vs. $c_p$. FastICA cases. Signature $s_2$. Noise added to the mixtures ($\sigma = 0.05$).

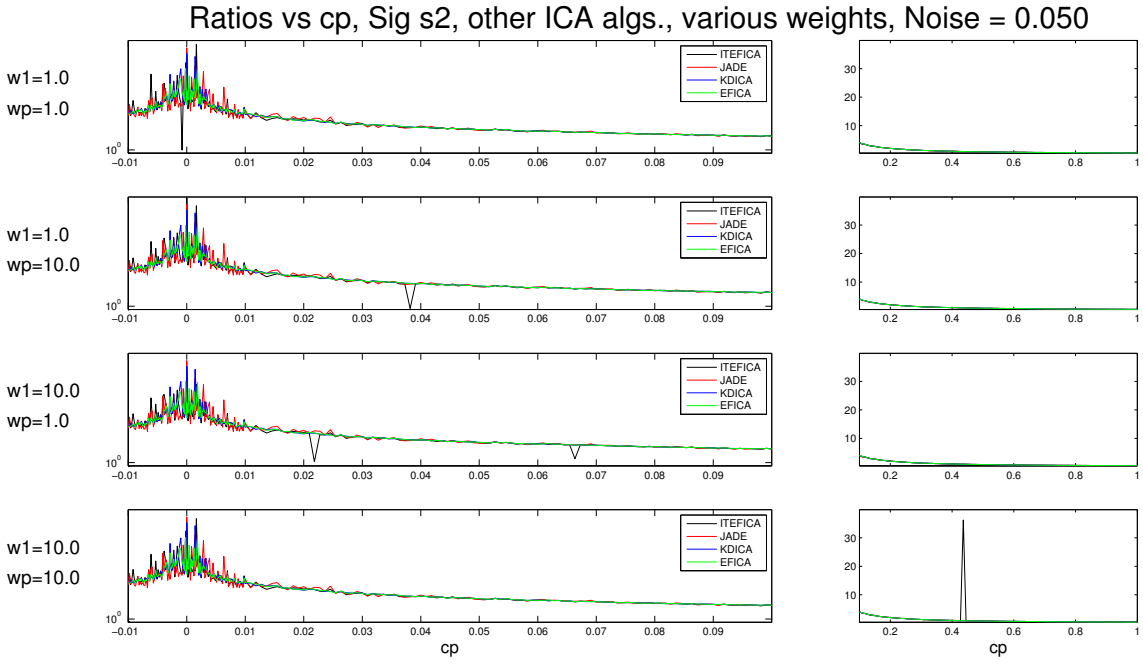



Figure A.72: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_2$. Noise added to the mixtures ($\sigma = 0.05$).

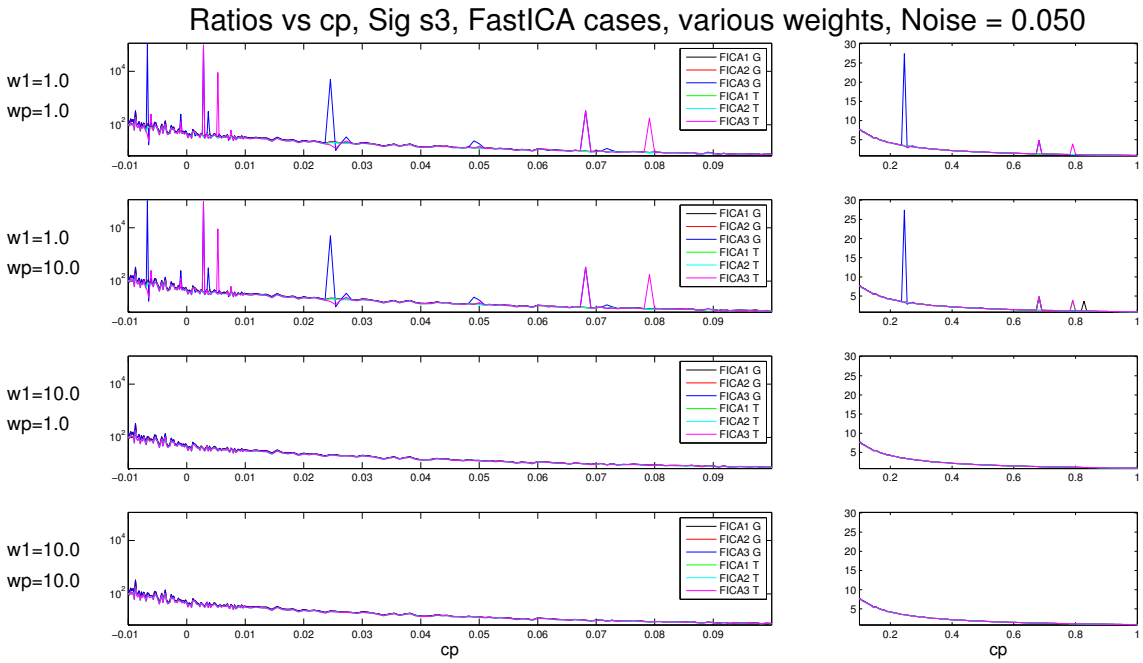



Figure A.73: Ratios vs. $c_p$. FastICA cases. Signature $s_3$. Noise added to the mixtures ($\sigma = 0.05$).

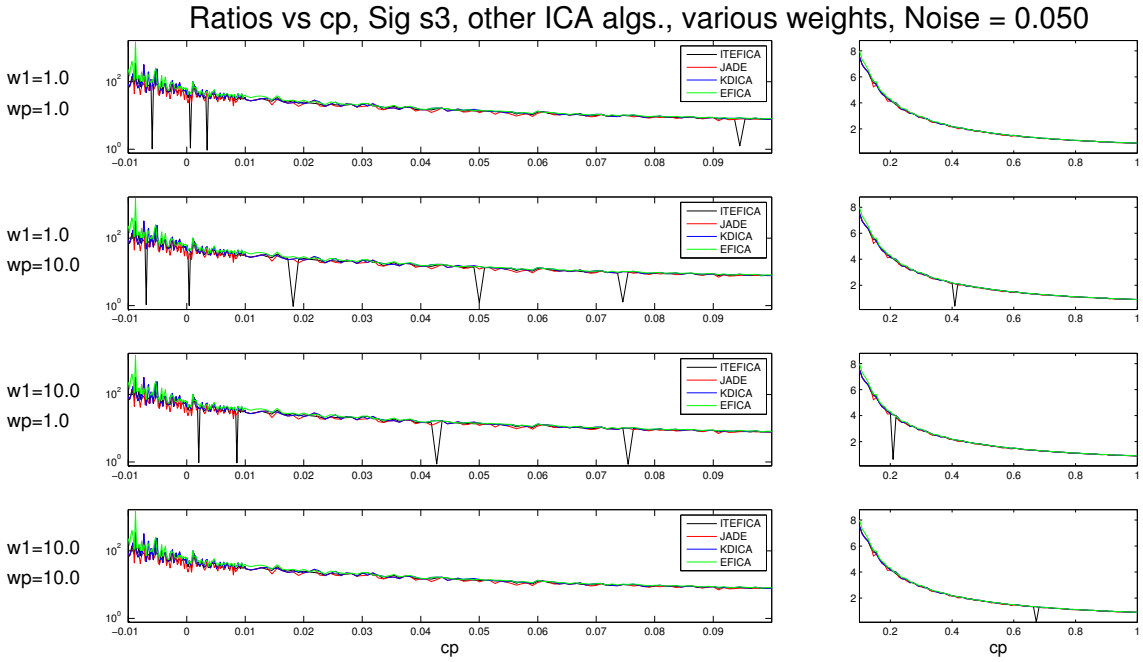



Figure A.74: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_3$. Noise added to the mixtures ($\sigma = 0.05$).

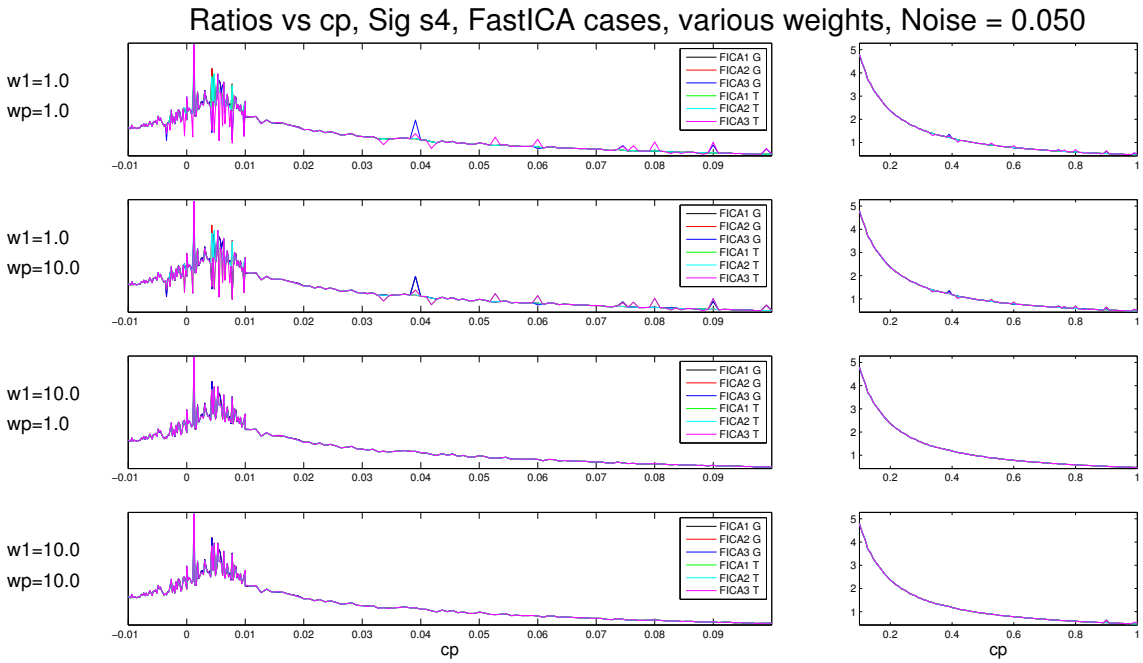



Figure A.75: Ratios vs. $c_p$. FastICA cases. Signature $s_4$. Noise added to the mixtures ($\sigma = 0.05$).

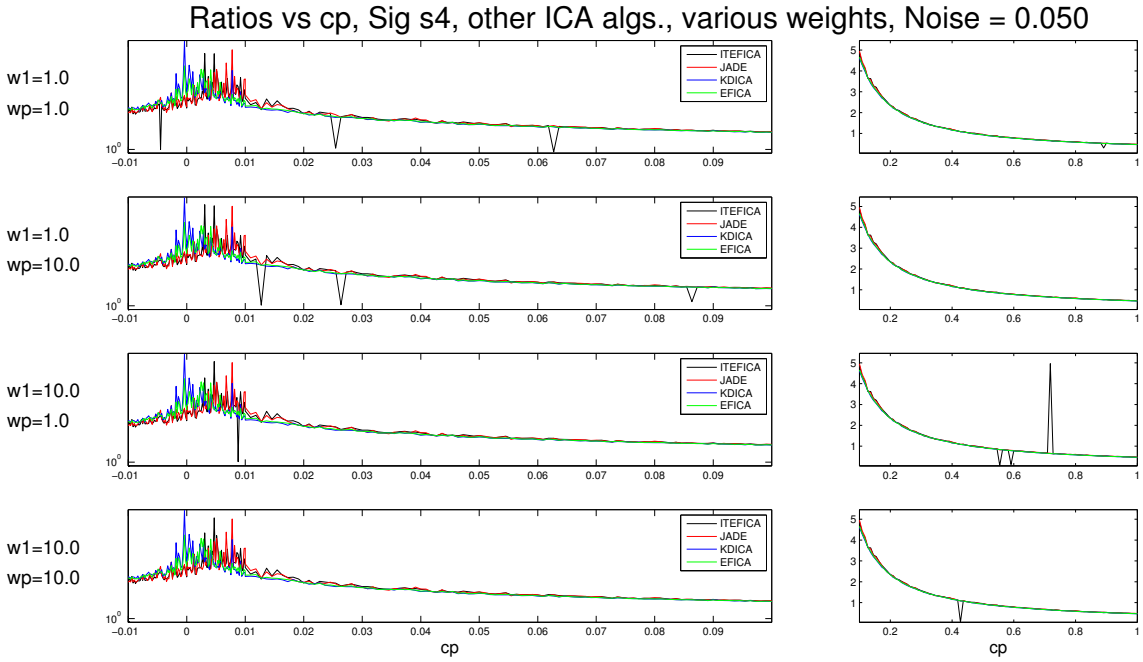



Figure A.76: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_4$. Noise added to the mixtures ($\sigma = 0.05$).

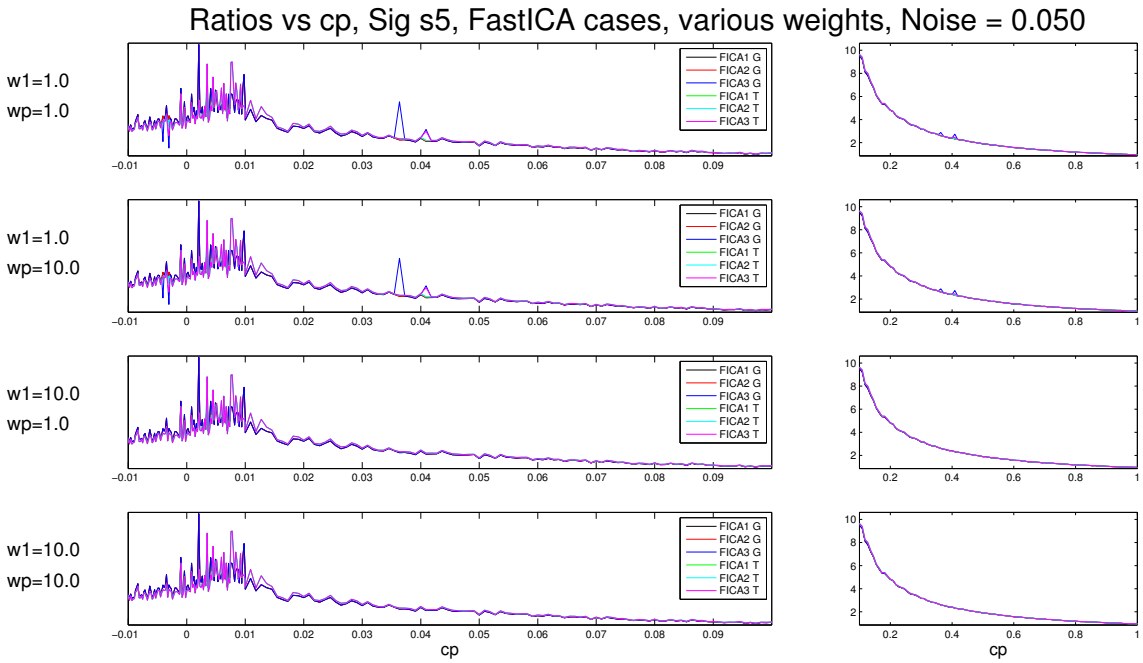



Figure A.77: Ratios vs. $c_p$. FastICA cases. Signature $s_5$. Noise added to the mixtures ($\sigma = 0.05$).

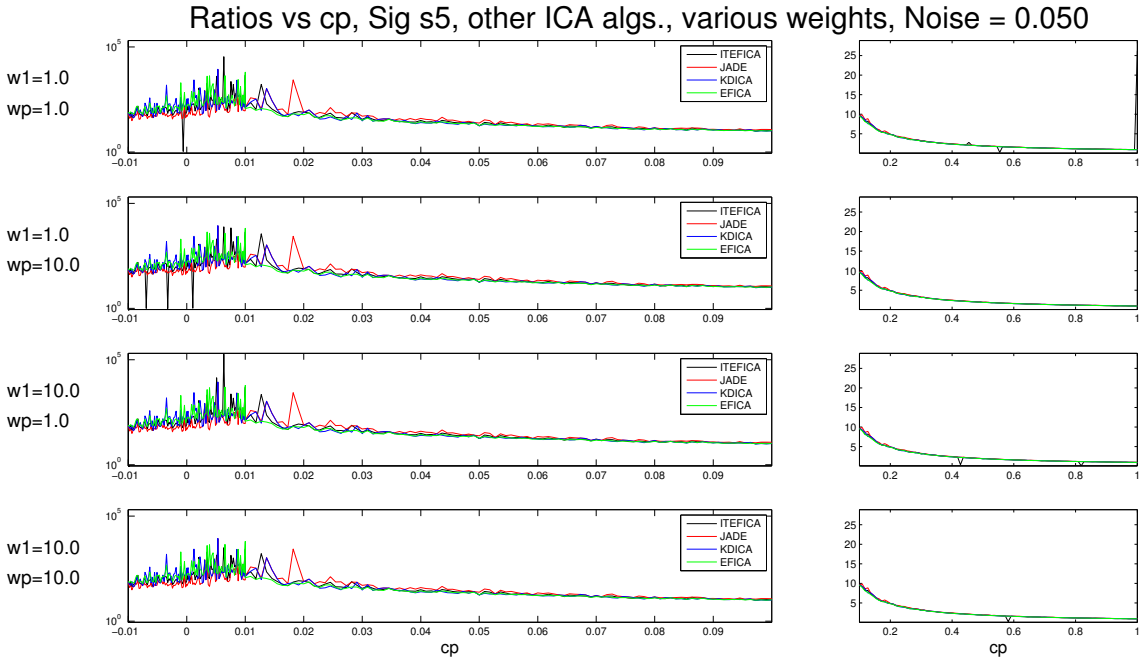



Figure A.78: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_5$. Noise added to the mixtures ($\sigma = 0.05$).

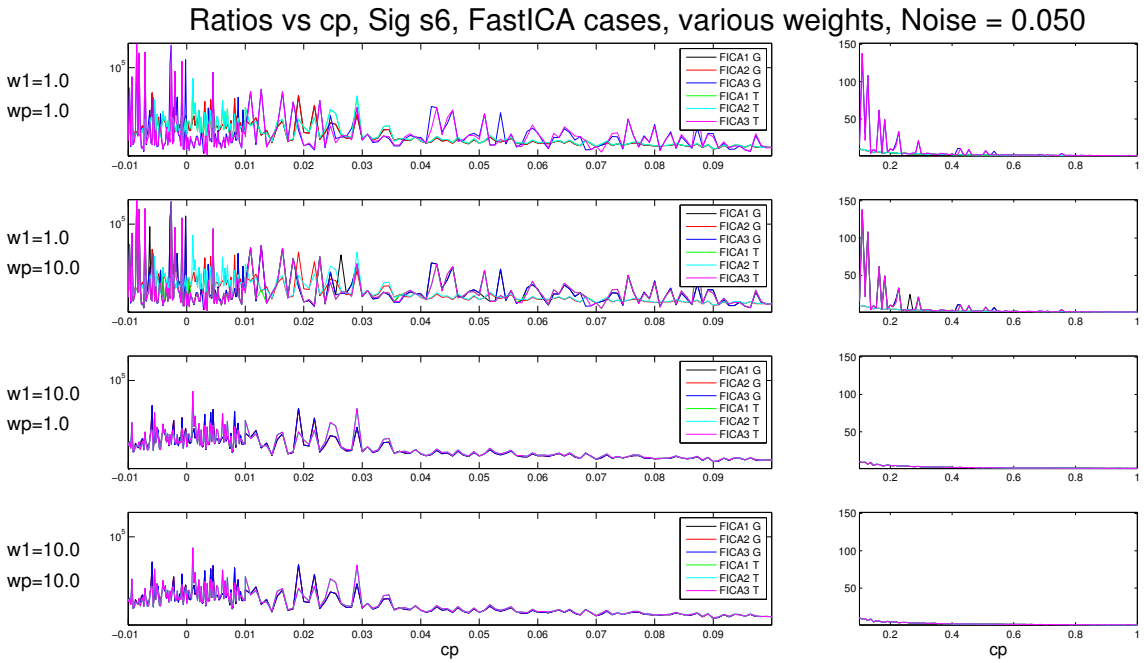



Figure A.79: Ratios vs. $c_p$. FastICA cases. Signature $s_6$. Noise added to the mixtures ($\sigma = 0.05$).

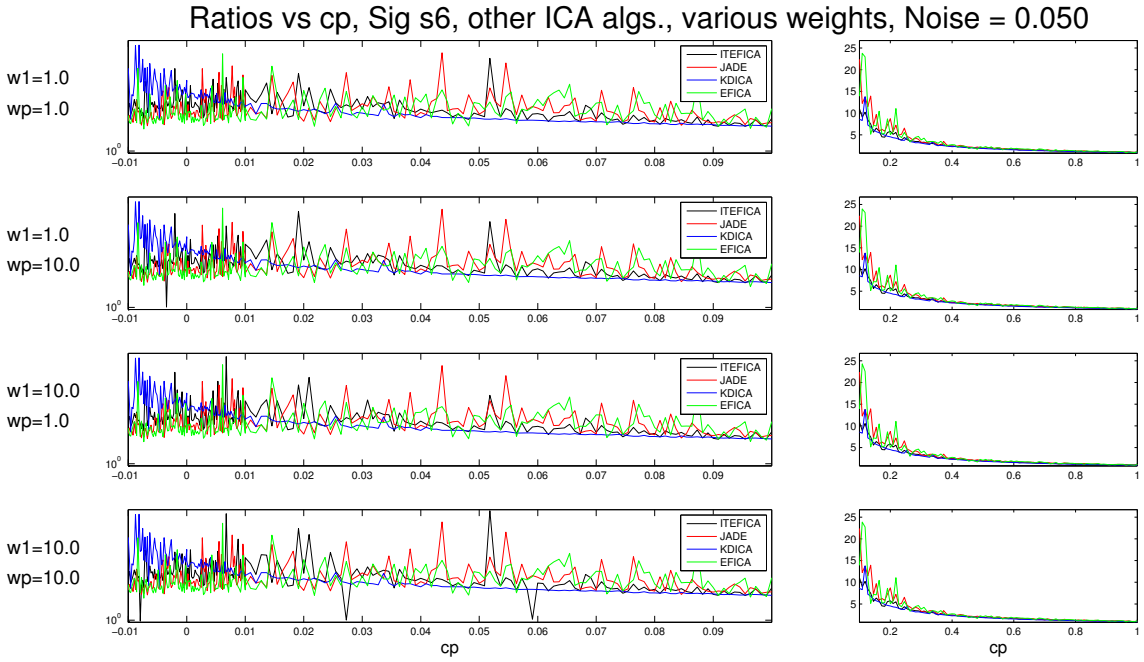



Figure A.80: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_6$. Noise added to the mixtures ($\sigma = 0.05$).

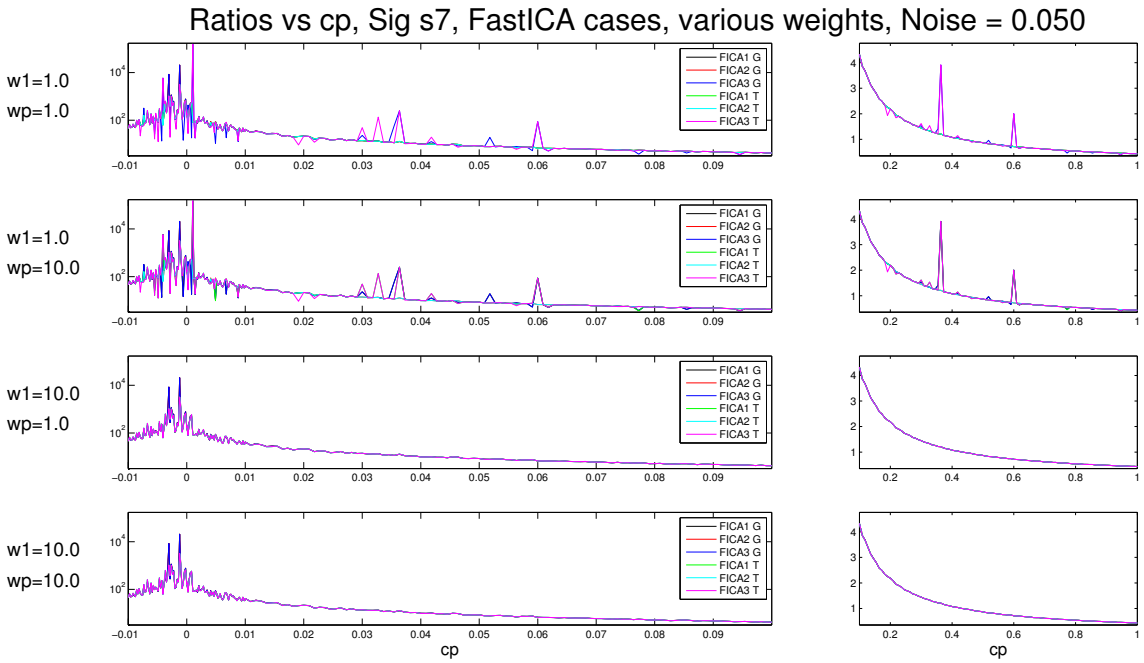



Figure A.81: Ratios vs. $c_p$. FastICA cases. Signature $s_7$. Noise added to the mixtures ($\sigma = 0.05$).

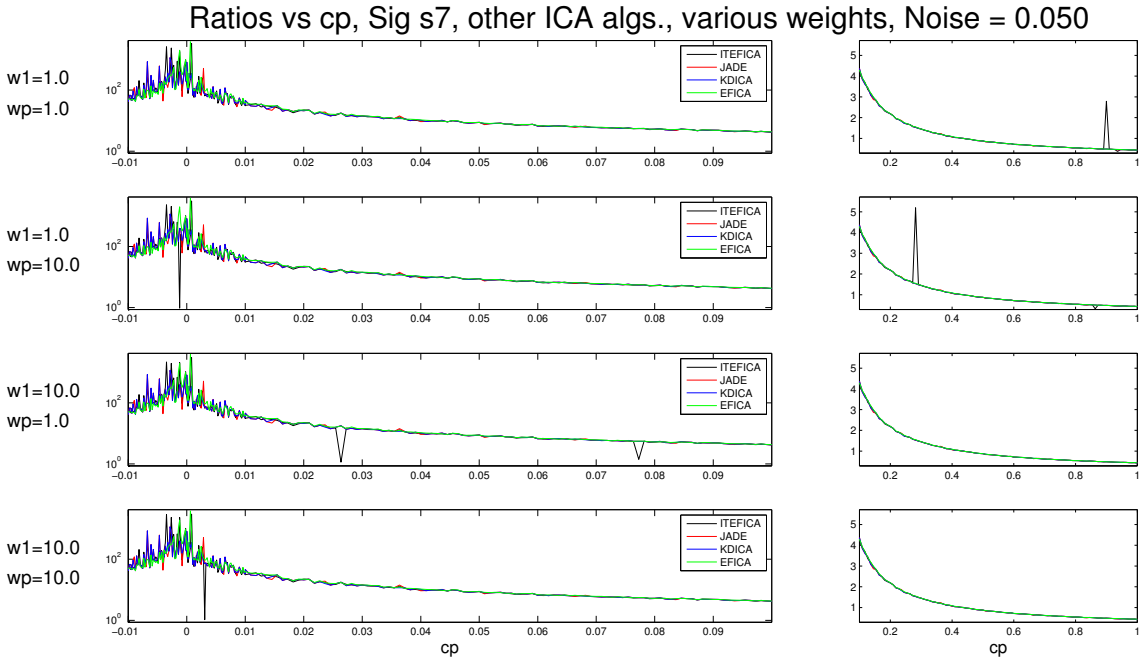



Figure A.82: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_7$. Noise added to the mixtures ($\sigma = 0.05$).

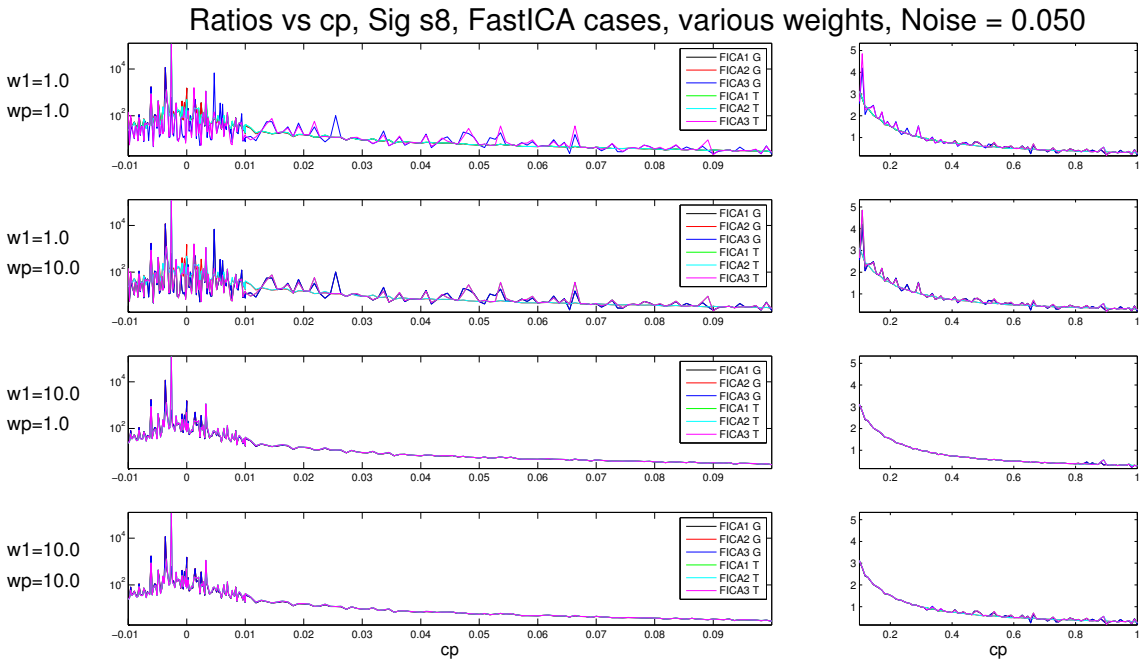



Figure A.83: Ratios vs. $c_p$. FastICA cases. Signature $s_8$. Noise added to the mixtures ($\sigma = 0.05$).

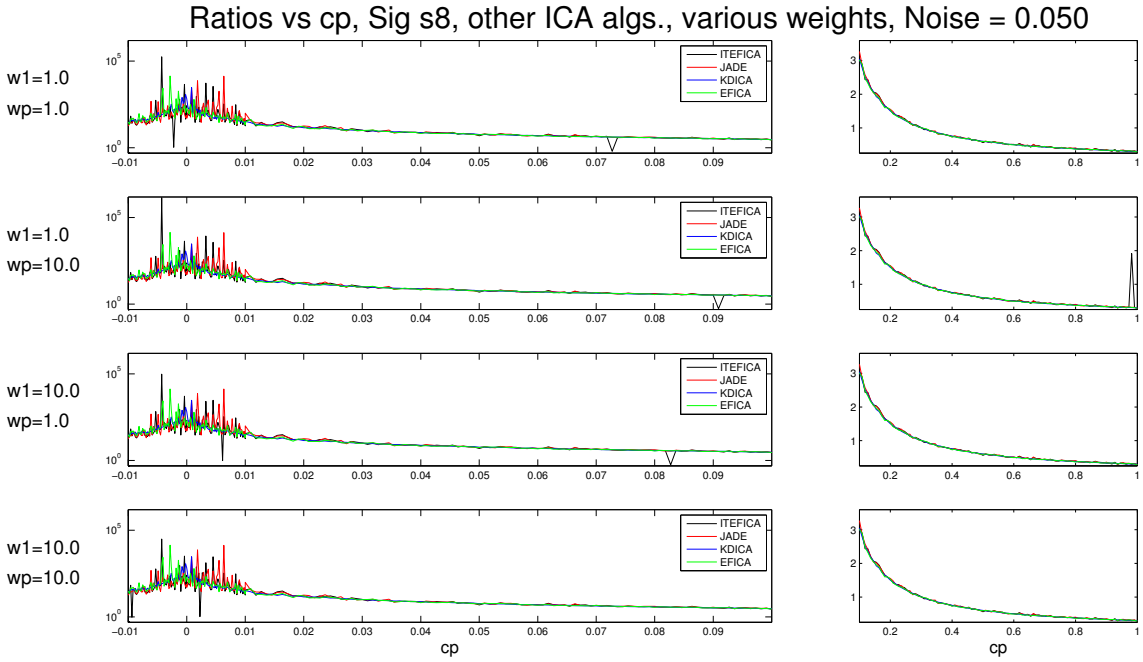



Figure A.84: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_8$. Noise added to the mixtures ($\sigma = 0.05$).

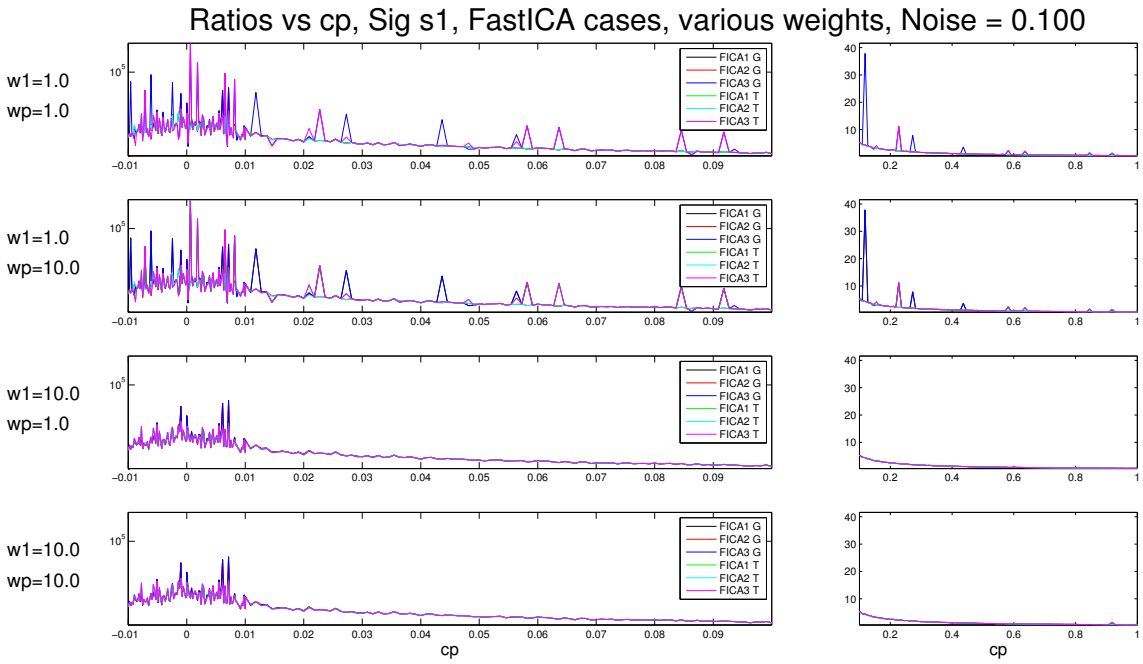



Figure A.85: Ratios vs. $c_p$. FastICA cases. Signature $s_1$. Noise added to the mixtures ($\sigma = 0.1$).

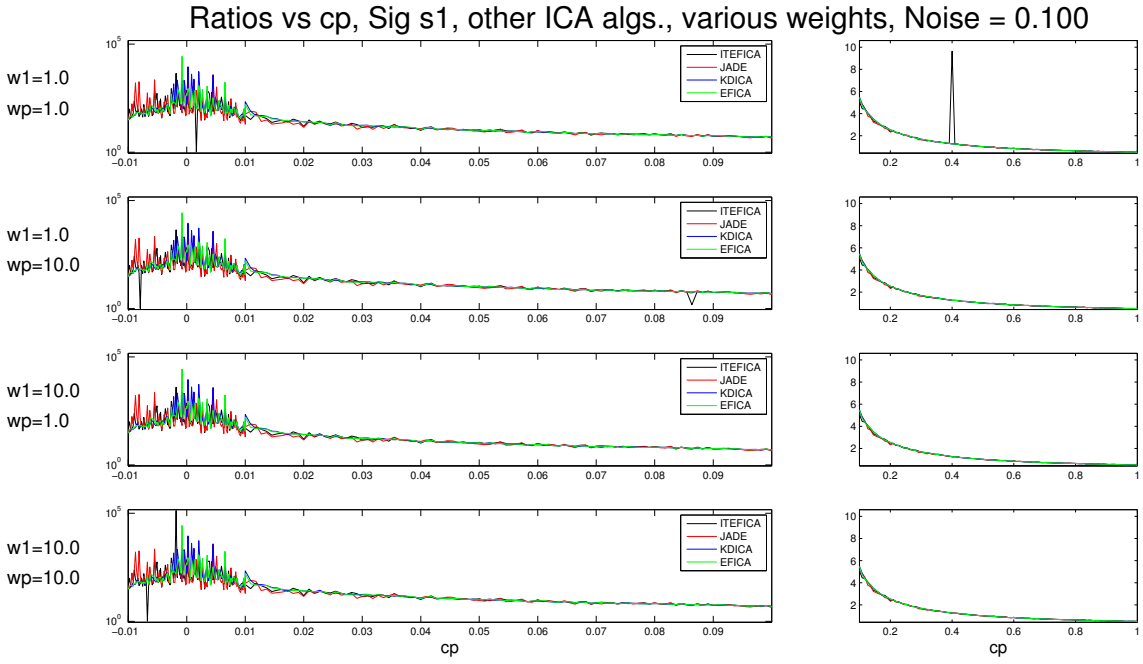



Figure A.86: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_1$. Noise added to the mixtures ($\sigma = 0.1$).

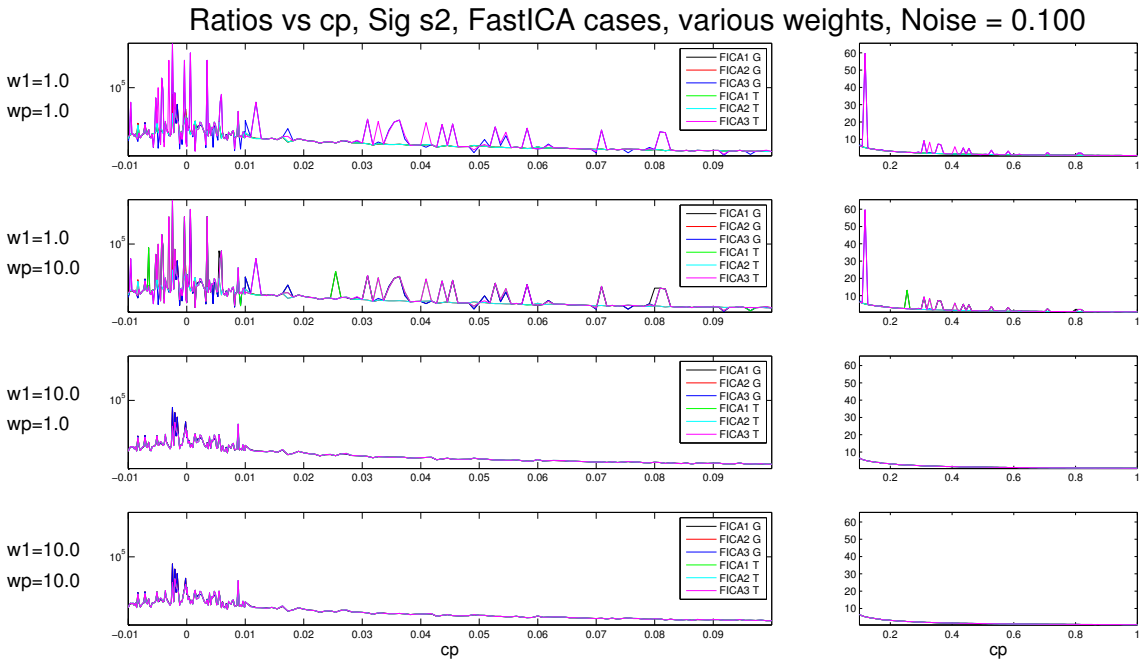



Figure A.87: Ratios vs. $c_p$. FastICA cases. Signature $s_2$. Noise added to the mixtures ($\sigma = 0.1$).

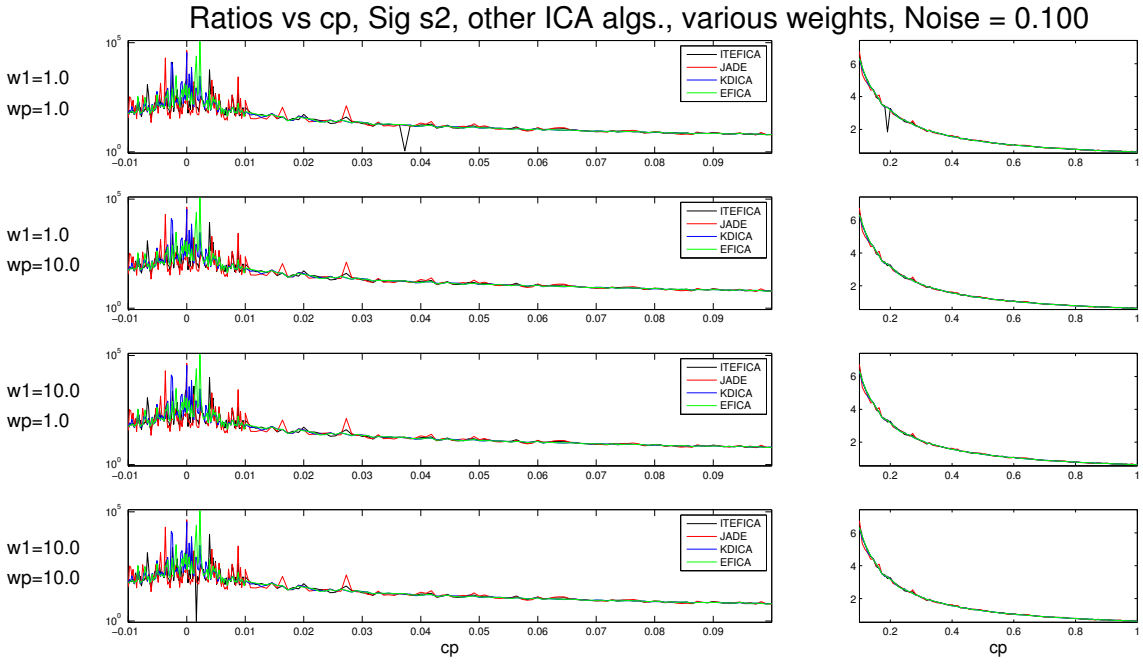



Figure A.88: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_2$. Noise added to the mixtures ($\sigma = 0.1$).

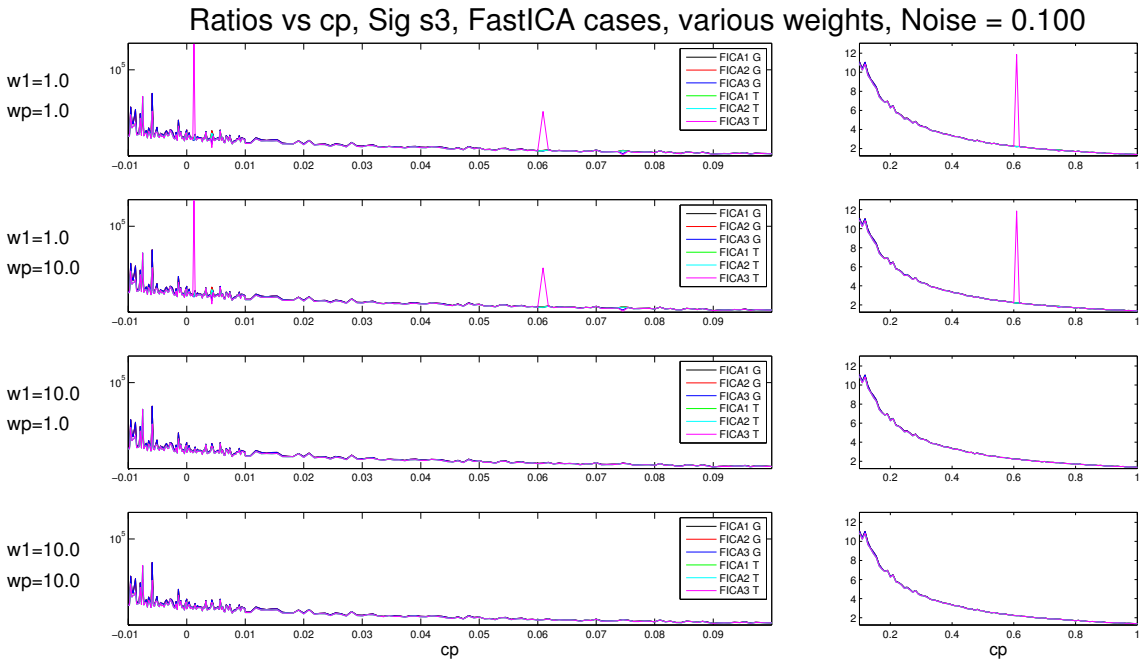



Figure A.89: Ratios vs. $c_p$. FastICA cases. Signature $s_3$. Noise added to the mixtures ($\sigma = 0.1$).

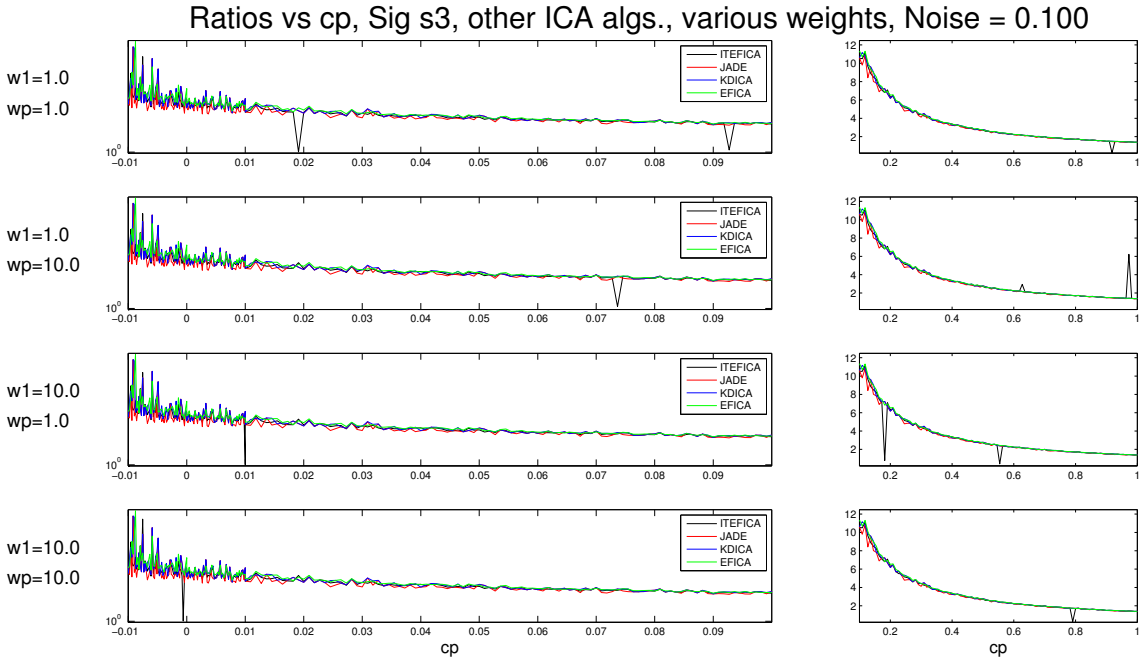



Figure A.90: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_3$. Noise added to the mixtures ($\sigma = 0.1$).

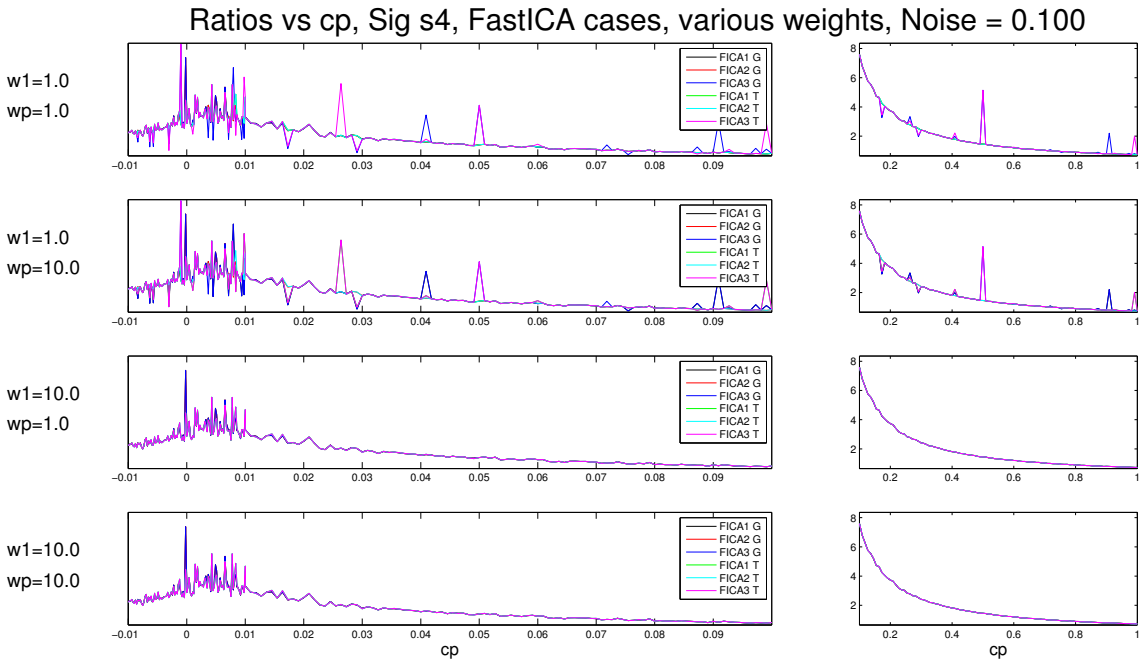



Figure A.91: Ratios vs. $c_p$. FastICA cases. Signature $s_4$. Noise added to the mixtures ($\sigma = 0.1$).

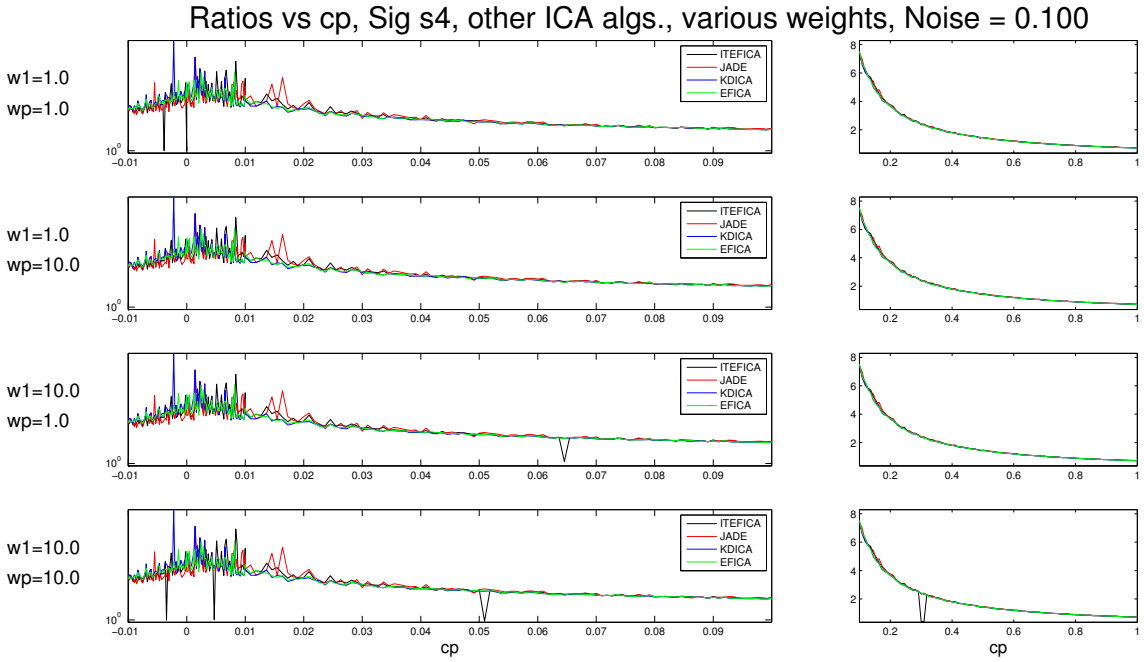



Figure A.92: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_4$. Noise added to the mixtures ($\sigma = 0.1$).

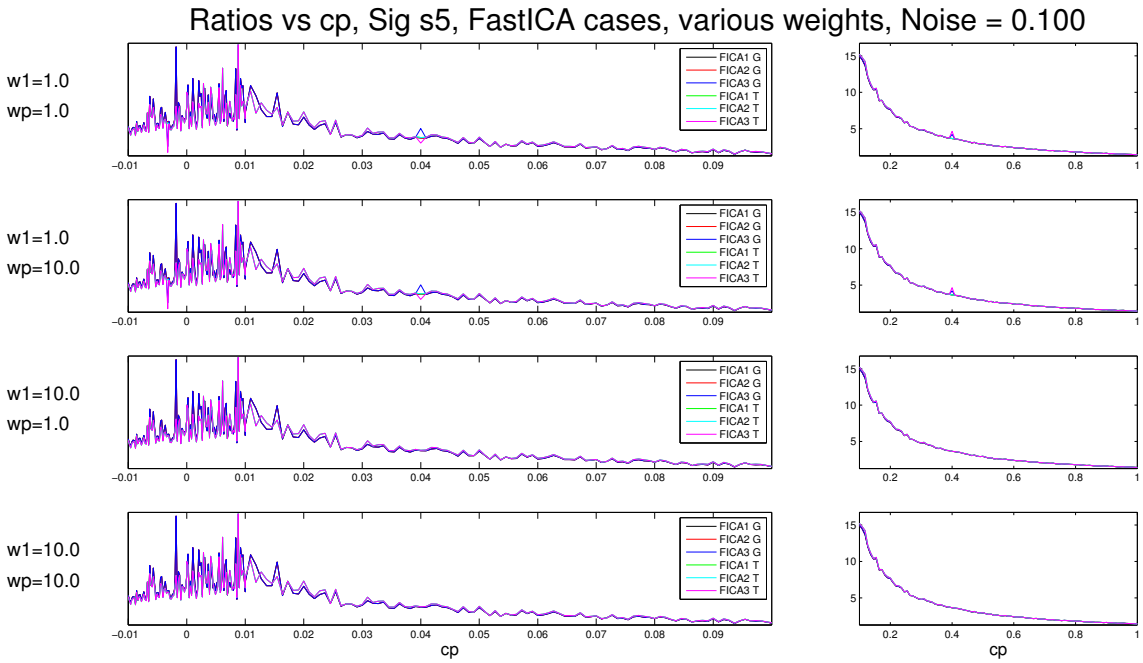



Figure A.93: Ratios vs. $c_p$. FastICA cases. Signature $s_5$. Noise added to the mixtures ($\sigma = 0.1$).

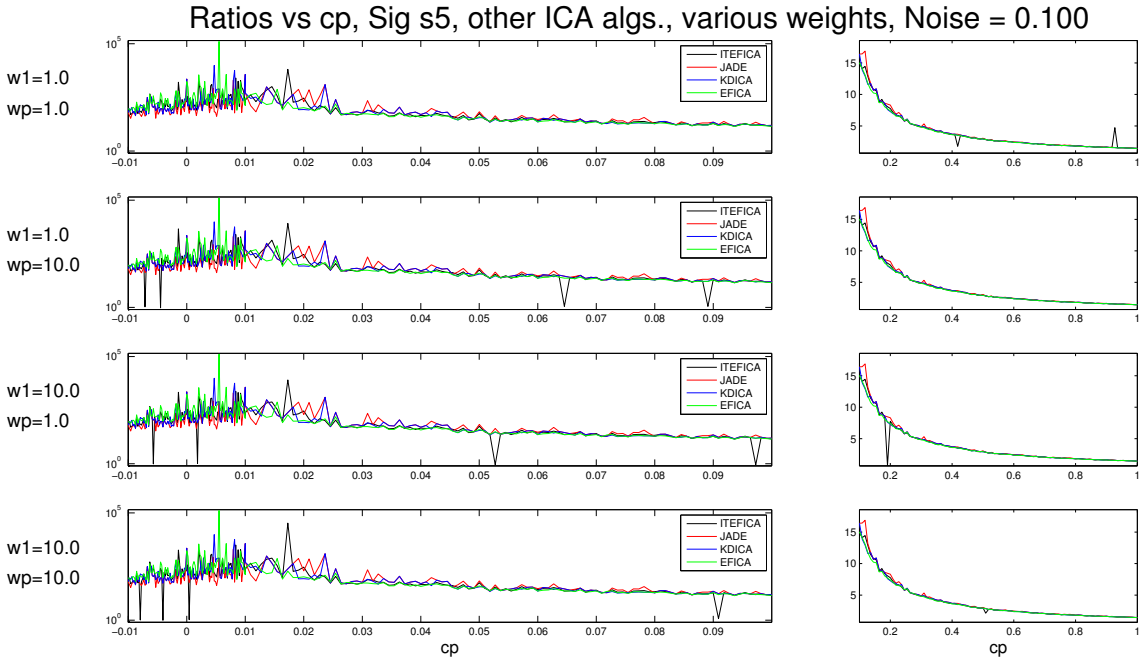



Figure A.94: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_5$. Noise added to the mixtures ($\sigma = 0.1$).

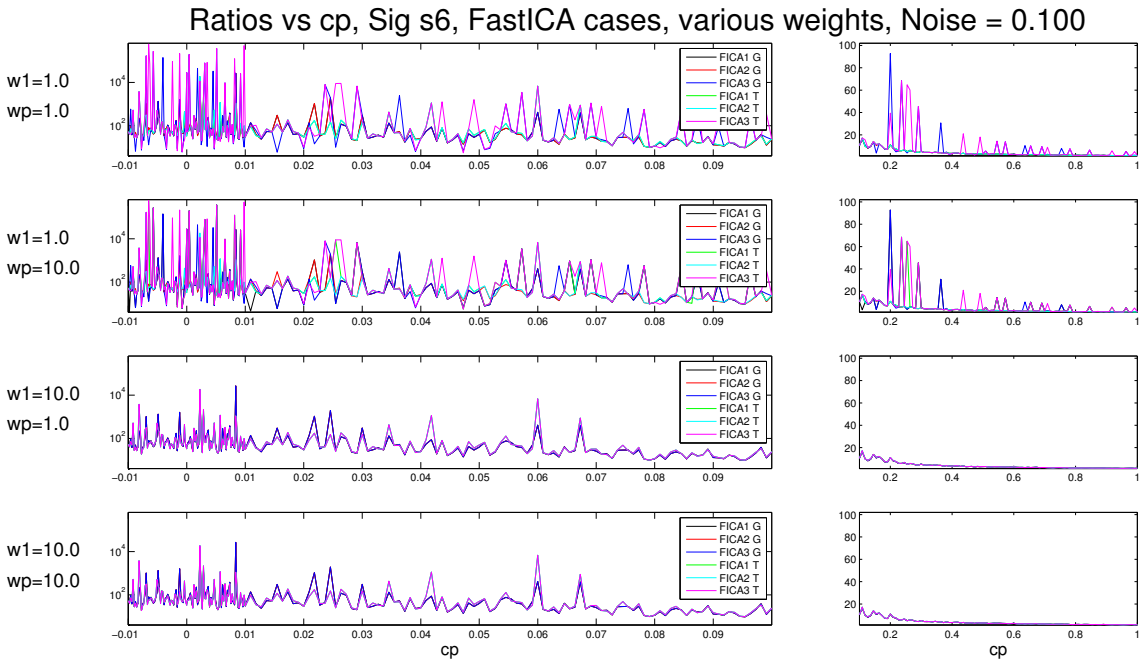



Figure A.95: Ratios vs. $c_p$. FastICA cases. Signature $s_6$. Noise added to the mixtures ($\sigma = 0.1$).

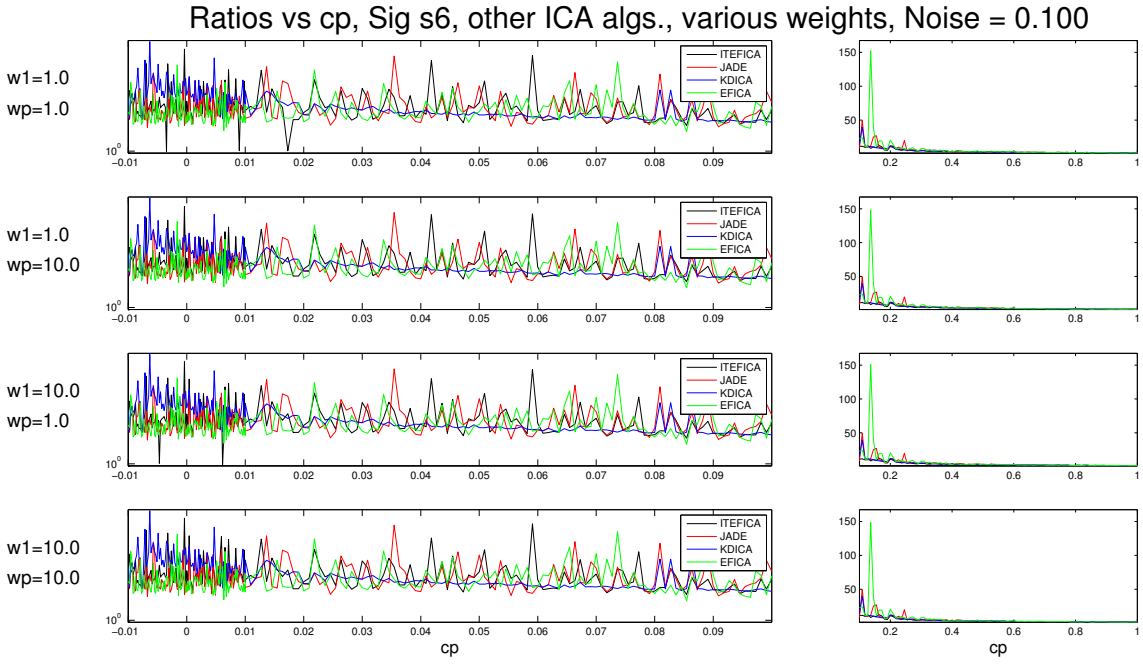



Figure A.96: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_6$. Noise added to the mixtures ($\sigma = 0.1$).

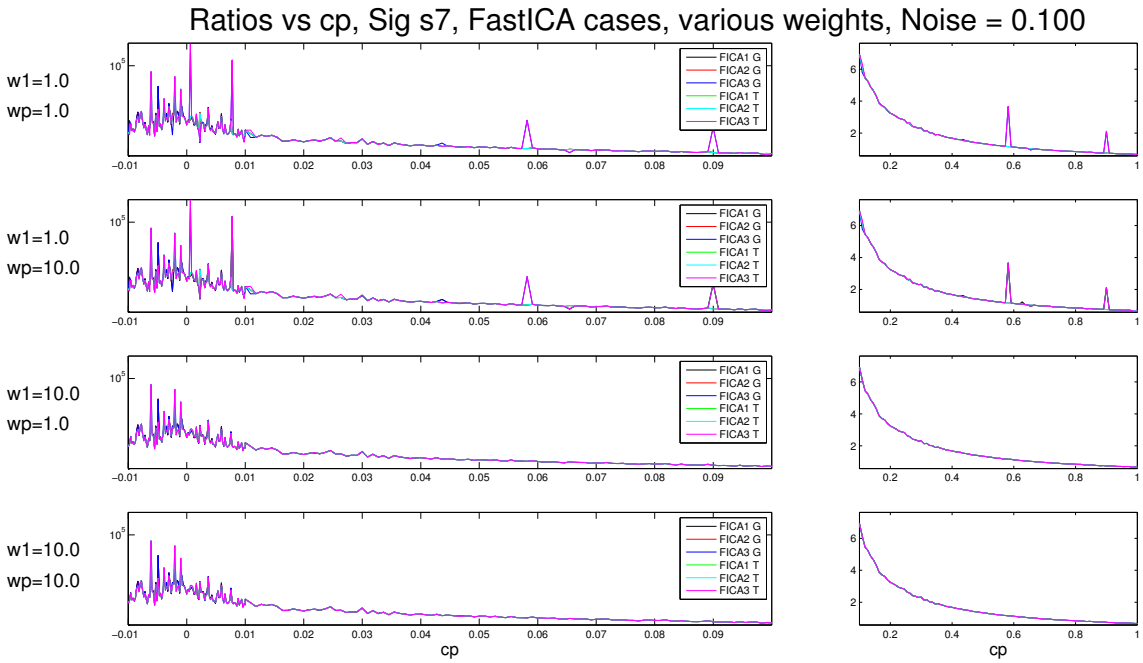



Figure A.97: Ratios vs. $c_p$. FastICA cases. Signature $s_7$. Noise added to the mixtures ($\sigma = 0.1$).

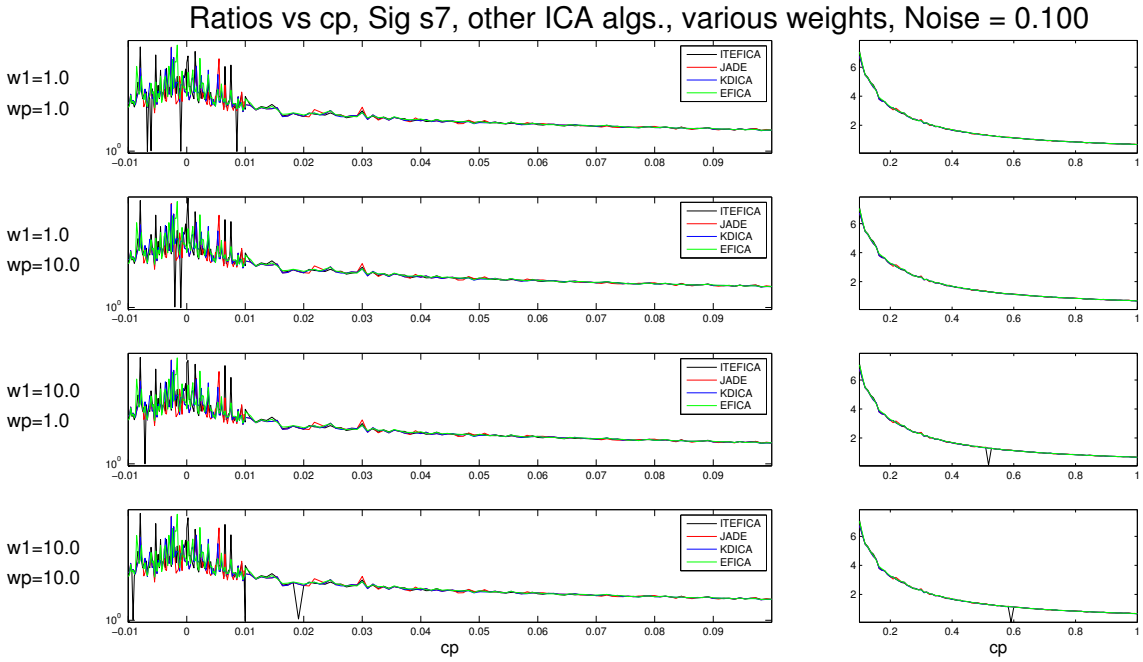



Figure A.98: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_7$. Noise added to the mixtures ($\sigma = 0.1$).

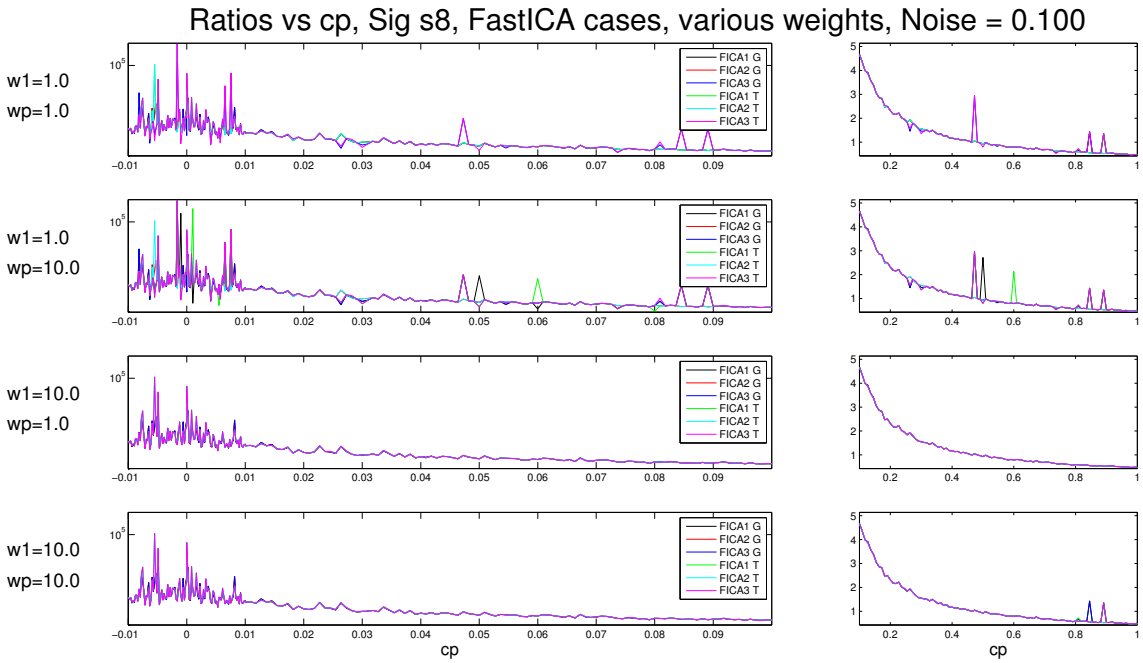



Figure A.99: Ratios vs. $c_p$. FastICA cases. Signature $s_8$. Noise added to the mixtures ($\sigma = 0.1$).

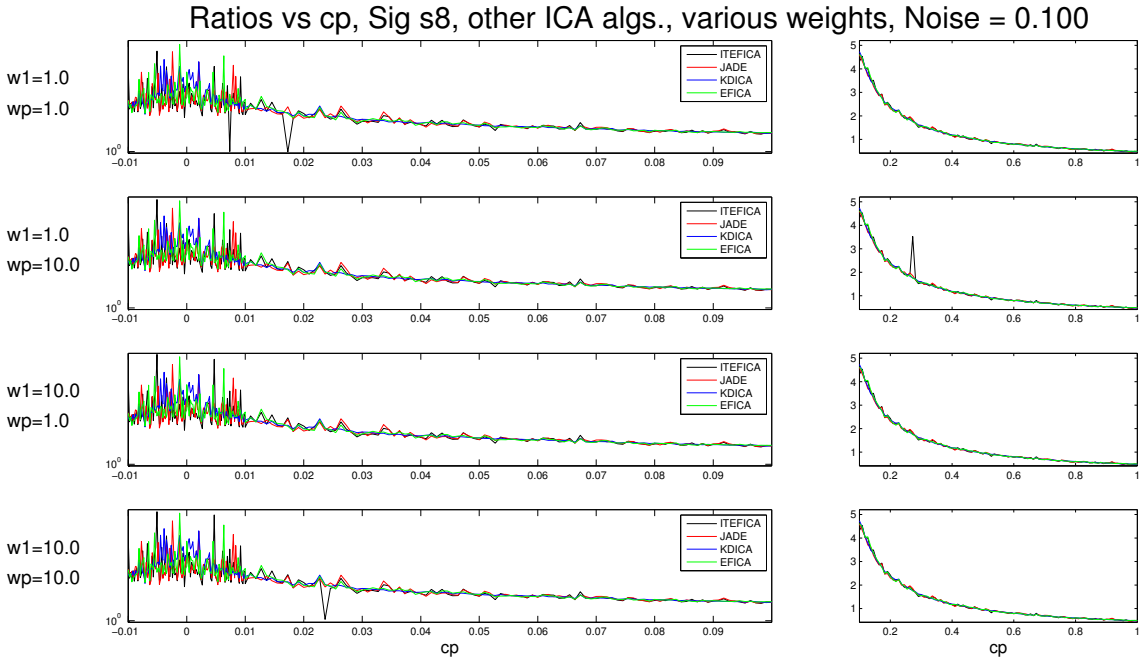



Figure A.100: Ratios vs. $c_p$. Other ICA algorithms. Signature $s_8$. Noise added to the mixtures ($\sigma = 0.1$).

# Appendix B

# Complete Set of Results - Independence Assumption

In this Appendix we present the results obtained from the experiments described in Section 5.4.

## B.1 Performance as a Function of Dependency (Sec. 5.4.2)

Here we show the results for all test subjects (Section 5.4.2).

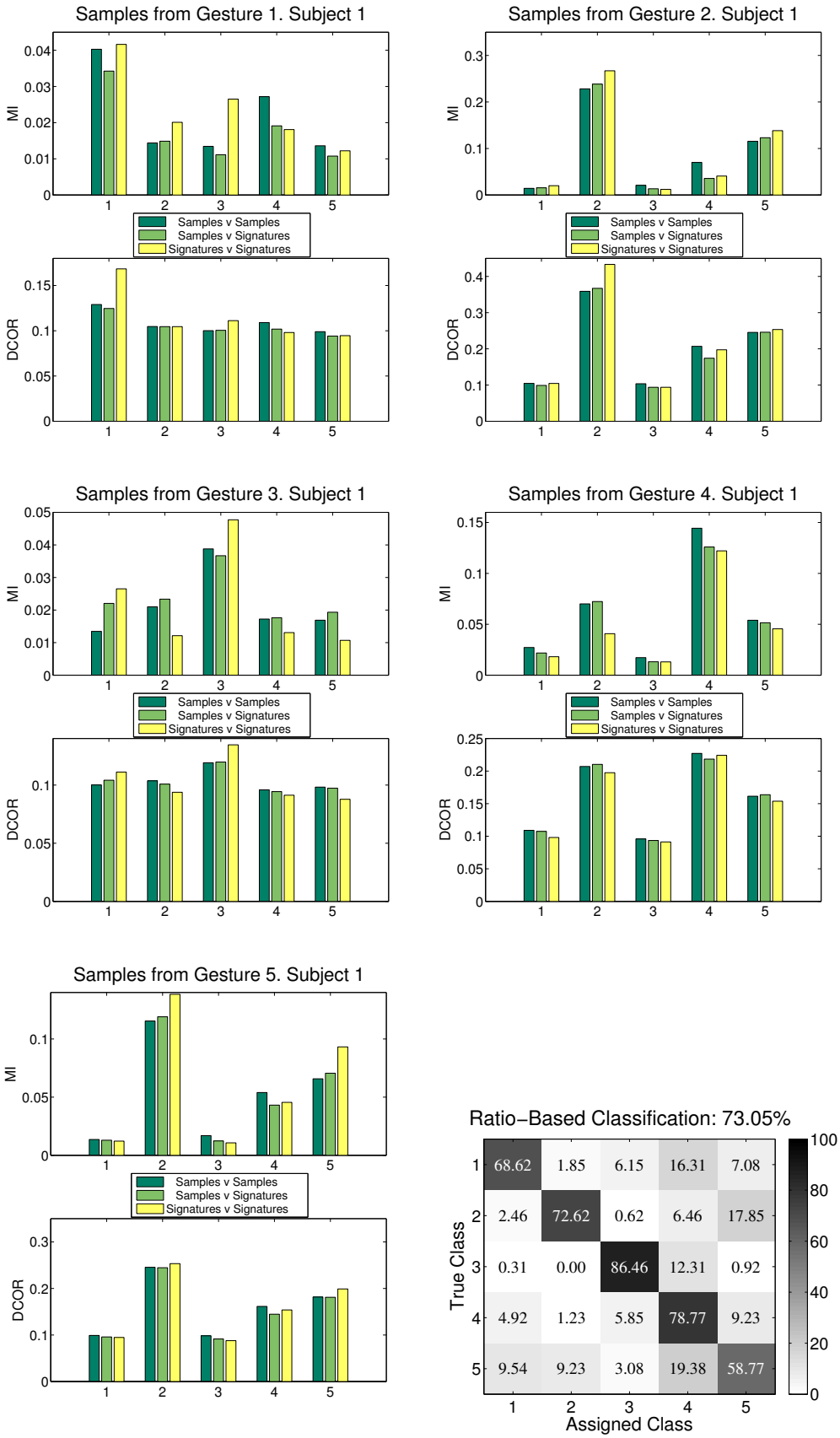



Figure B.1: MI and DCOR average values. 5 different gestures. Test subject 1.

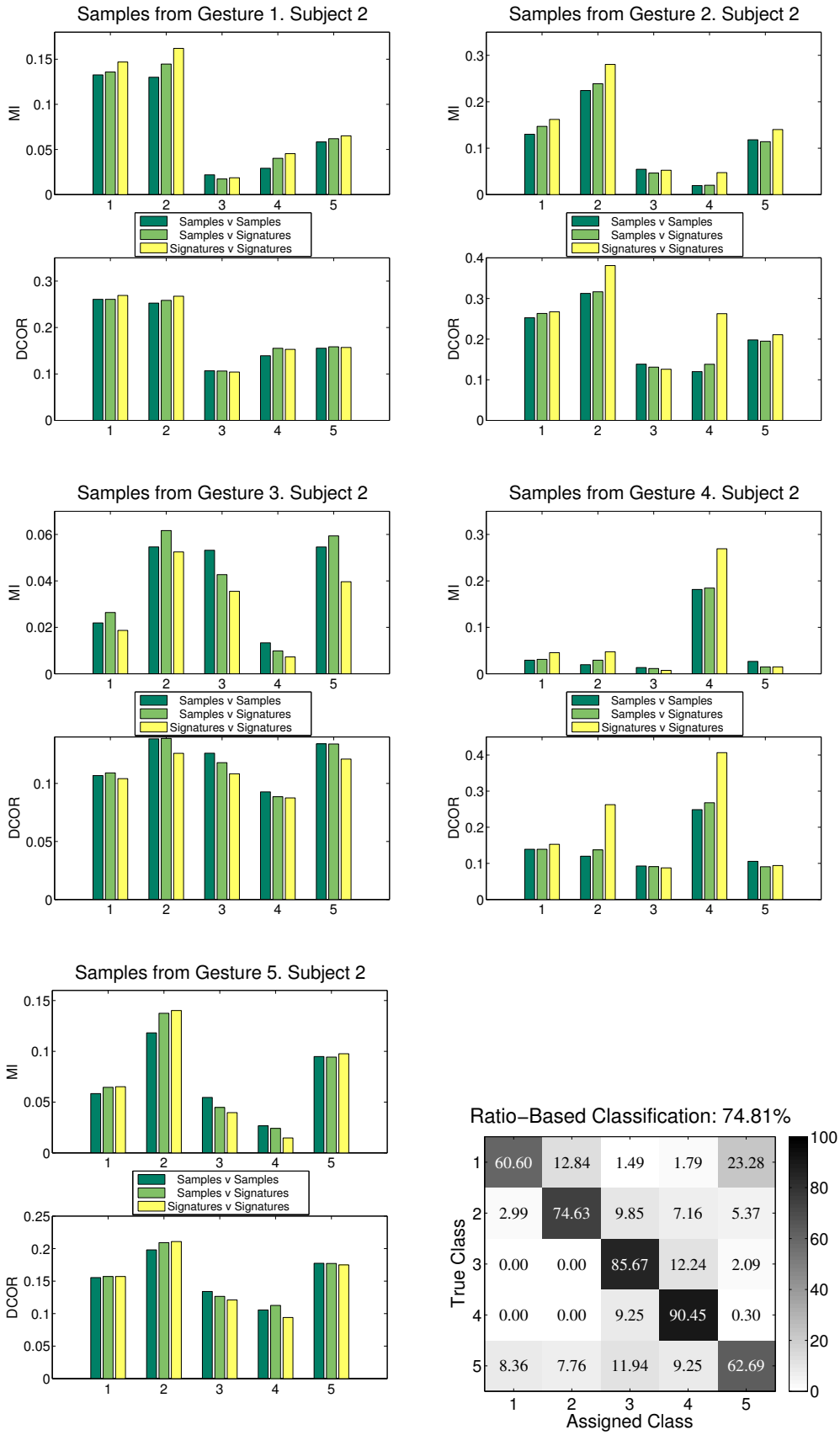



Figure B.2: MI and DCOR average values. 5 different gestures. Test subject 2.

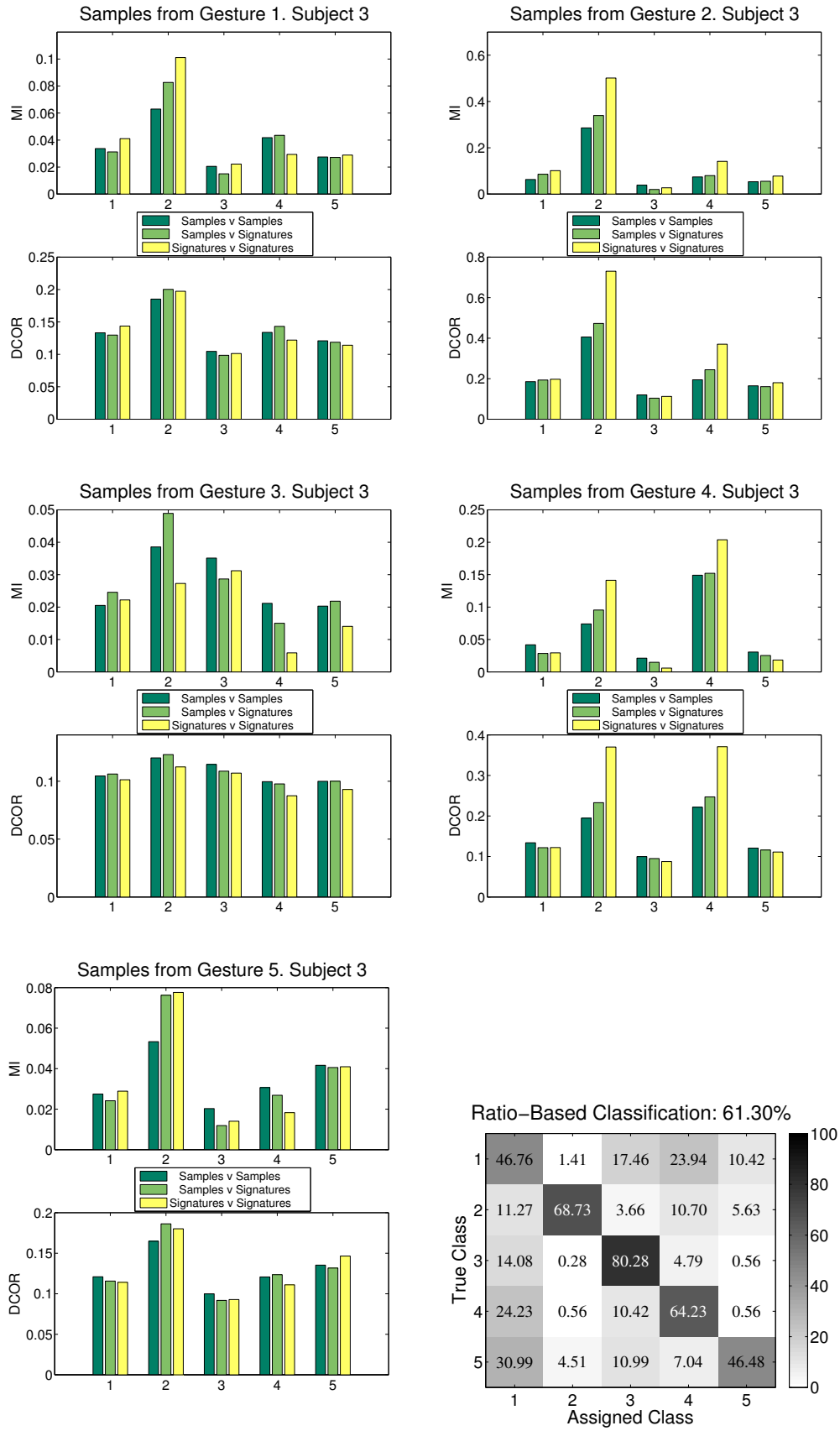



Figure B.3: MI and DCOR average values. 5 different gestures. Test subject 3.

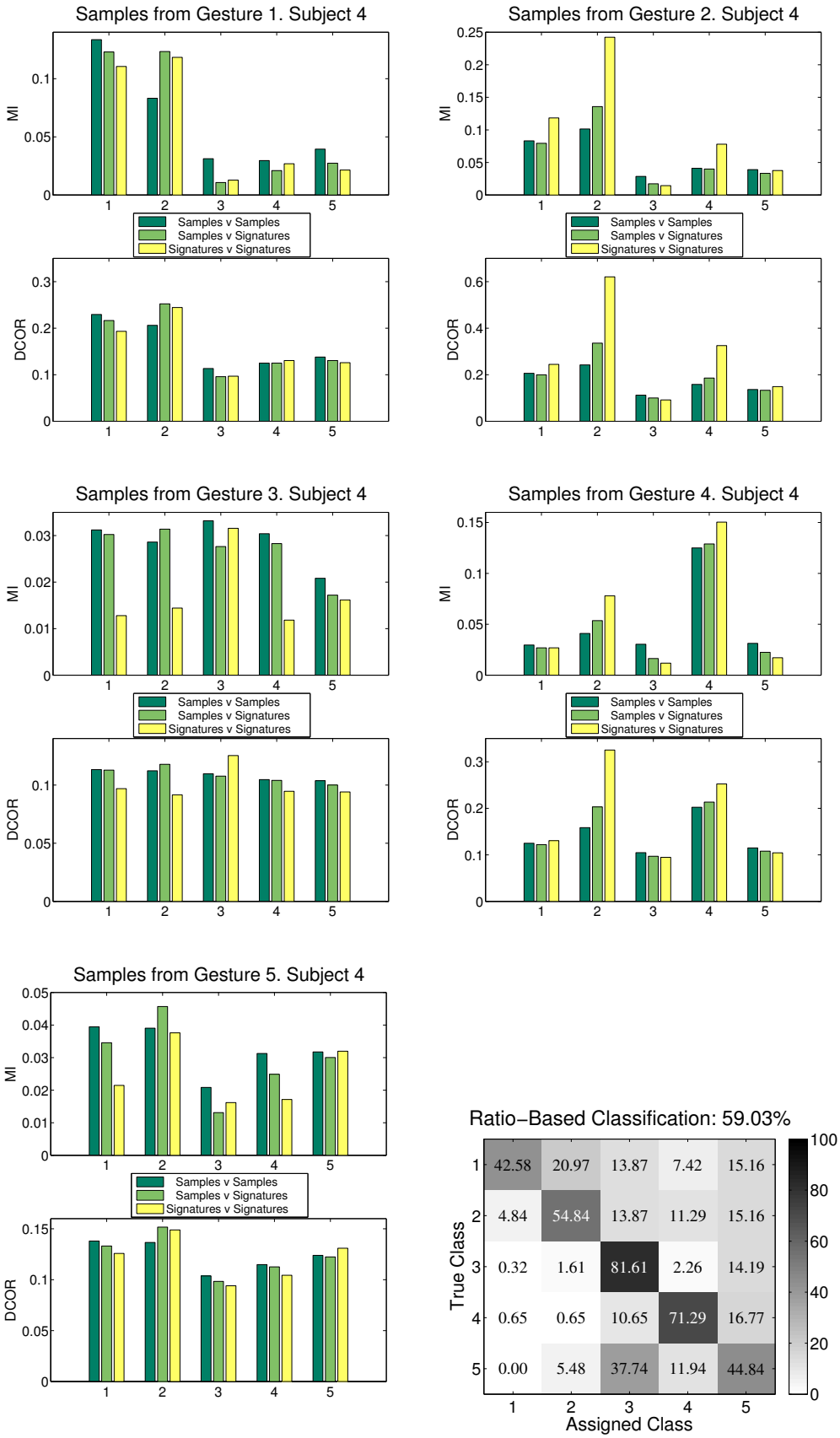



Figure B.4: MI and DCOR average values. 5 different gestures. Test subject 4.

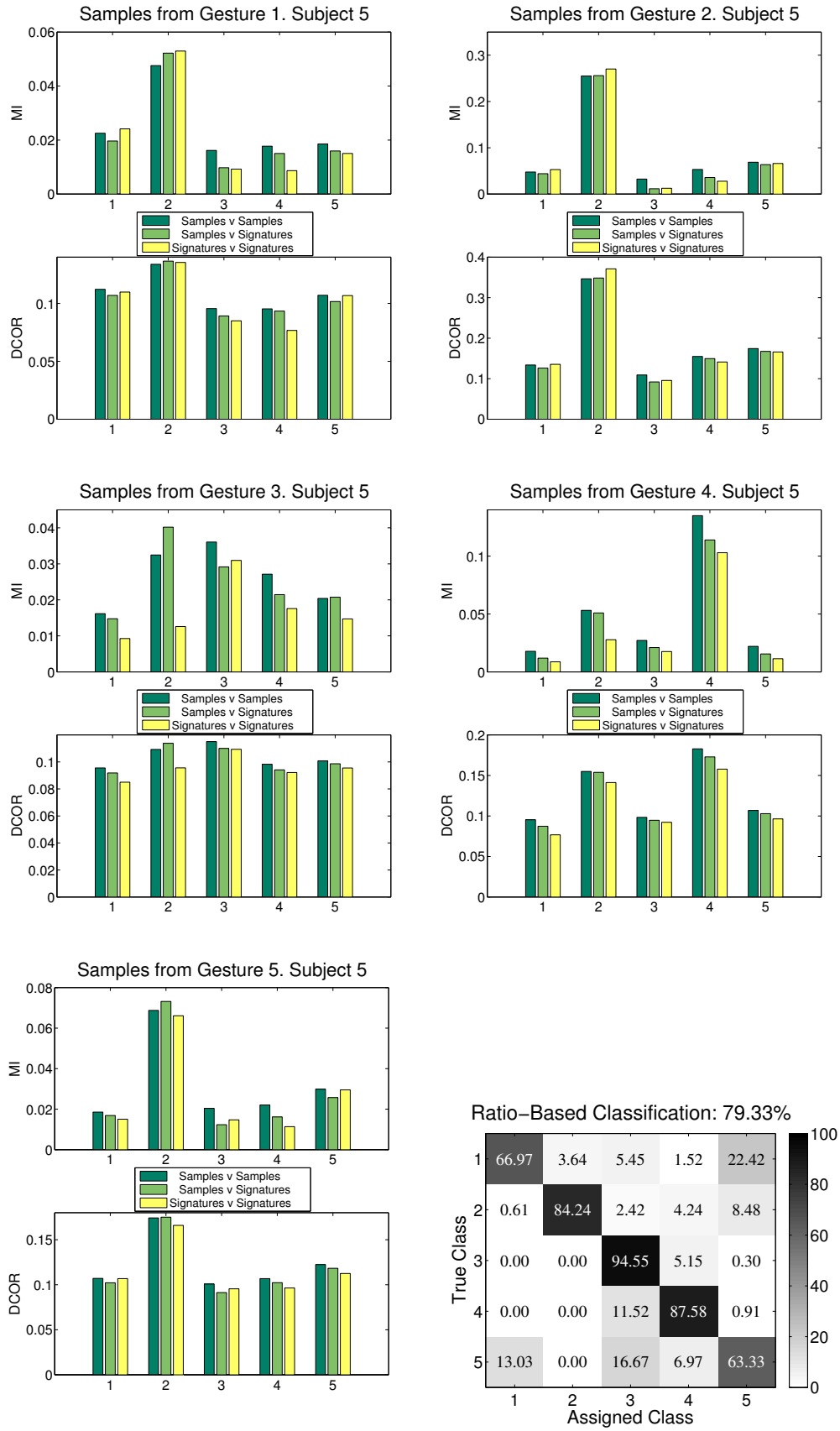



Figure B.5: MI and DCOR average values. 5 different gestures. Test subject 5.

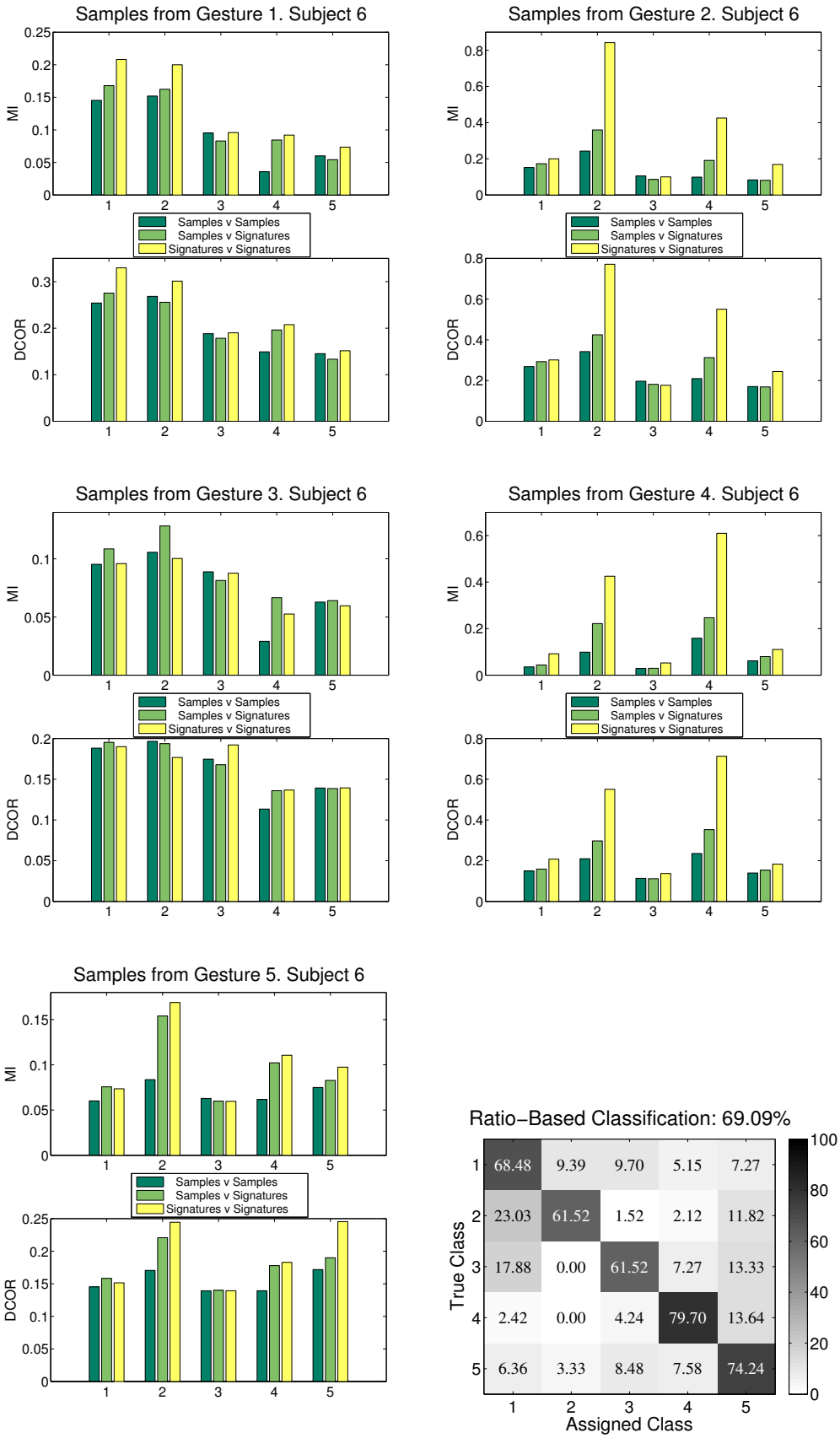



Figure B.6: MI and DCOR average values. 5 different gestures. Test subject 6.

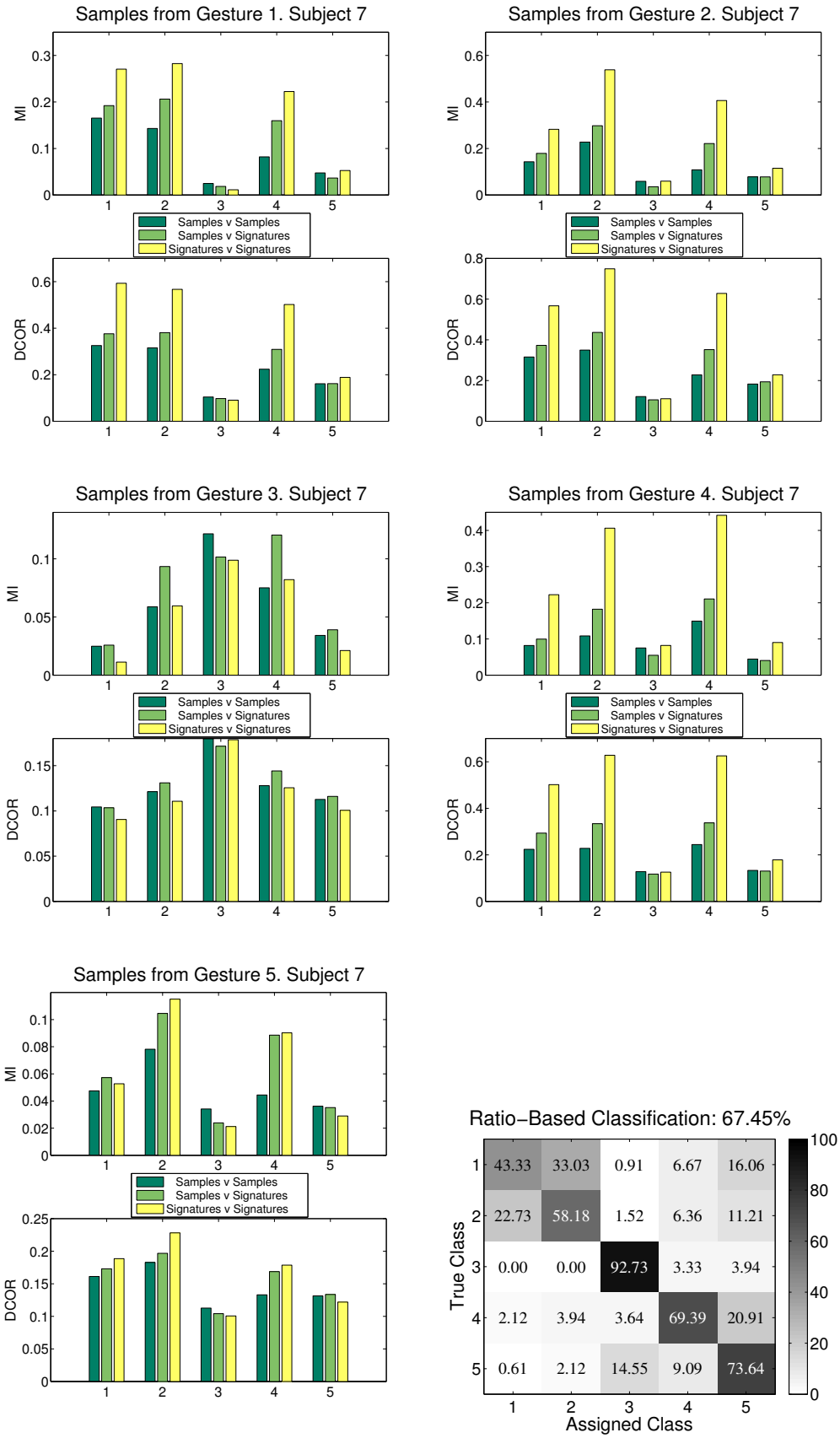



Figure B.7: MI and DCOR average values. 5 different gestures. Test subject 7.

# Appendix C

# Complete Set of Results - Learning Signatures

In this Appendix we present the results obtained from the experiments described in Section 5.5.

## C.1 sEMG Results, All Test Subjects (Sec. 5.5)

The following tables show confusion matrices with average percentages over the 5-fold tests. The average correct classification percentages are also presented at the bottom of each table.

Legend: M-B - Mean-Based signature; C.E - Clustering-based signature using Euclidean distance; C.C - Clustering-based signature using Correlation distance; ICA All - ICA decomposition based signature using all training signals combined to get the basic components; ICA Per - ICA decomposition based signature using training signals per class separately.

Table C.1: 3 gestures. Test subjects 1 (left) and 2 (right).

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 94.8 | 0.0 | 5.2 |
| | 2 | 0.0 | 99.4 | 0.6 |
| | 3 | 0.9 | 0.0 | 99.1 |
| Correct classification: 97.7% | | | | |

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.8 | 1.2 | 0.0 |
| | 2 | 0.0 | 99.4 | 0.6 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.4% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 96.0 | 0.0 | 4.0 |
| | 2 | 0.0 | 99.7 | 0.3 |
| | 3 | 0.9 | 0.0 | 99.1 |
| Correct classification: 98.3% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.8 | 1.2 | 0.0 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.6% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 95.1 | 0.0 | 4.9 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.9 | 0.0 | 99.1 |
| Correct classification: 98.1% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.5 | 1.5 | 0.0 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 93.2 | 0.0 | 6.8 |
| | 2 | 0.0 | 99.7 | 0.3 |
| | 3 | 0.9 | 0.0 | 99.1 |
| Correct classification: 97.3% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 97.9 | 2.1 | 0.0 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.3% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 95.7 | 0.0 | 4.3 |
| | 2 | 0.0 | 99.7 | 0.3 |
| | 3 | 0.9 | 0.0 | 99.1 |
| Correct classification: 98.2% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.5 | 1.5 | 0.0 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | |

Table C.2: 3 gestures. Test subjects 3 (left) and 4 (right).

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 96.6 | 0.6 | 2.8 |
| | 2 | 3.9 | 96.1 | 0.0 |
| | 3 | 2.8 | 0.8 | 96.3 |
| Correct classification: 96.3% | | | | |

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 93.9 | 5.2 | 1.0 |
| | 2 | 2.3 | 96.8 | 1.0 |
| | 3 | 0.6 | 5.5 | 93.9 |
| Correct classification: 94.8% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 95.8 | 0.8 | 3.4 |
| | 2 | 3.1 | 96.9 | 0.0 |
| | 3 | 1.7 | 0.8 | 97.5 |
| Correct classification: 96.7% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 90.6 | 7.4 | 1.9 |
| | 2 | 2.6 | 95.8 | 1.6 |
| | 3 | 0.0 | 7.1 | 92.9 |
| Correct classification: 93.1% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 96.1 | 0.6 | 3.4 |
| | 2 | 7.0 | 92.7 | 0.3 |
| | 3 | 2.5 | 0.8 | 96.6 |
| Correct classification: 95.1% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 93.5 | 5.5 | 1.0 |
| | 2 | 2.3 | 97.1 | 0.6 |
| | 3 | 0.3 | 5.5 | 94.2 |
| Correct classification: 94.9% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 96.9 | 0.3 | 2.8 |
| | 2 | 5.1 | 94.9 | 0.0 |
| | 3 | 2.0 | 0.8 | 97.2 |
| Correct classification: 96.3% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 93.9 | 5.8 | 0.3 |
| | 2 | 2.6 | 96.5 | 1.0 |
| | 3 | 0.6 | 4.2 | 95.2 |
| Correct classification: 95.2% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 96.1 | 0.8 | 3.1 |
| | 2 | 3.7 | 96.3 | 0.0 |
| | 3 | 2.5 | 0.6 | 96.9 |
| Correct classification: 96.4% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 93.9 | 4.8 | 1.3 |
| | 2 | 1.9 | 97.4 | 0.6 |
| | 3 | 0.6 | 4.8 | 94.5 |
| Correct classification: 95.3% | | | | |

Table C.3: 3 gestures. Test subjects 5 (left) and 6 (right).

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 99.7 | 0.0 | 0.3 |
| | 2 | 0.0 | 99.4 | 0.6 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.7% | | | | |

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 90.0 | 9.4 | 0.6 |
| | 2 | 5.2 | 94.8 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 94.9% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 99.7 | 0.0 | 0.3 |
| | 2 | 0.0 | 99.4 | 0.6 |
| | 3 | 0.0 | 0.3 | 99.7 |
| Correct classification: 99.6% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 89.1 | 10.9 | 0.0 |
| | 2 | 5.8 | 94.2 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 94.4% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 99.1 | 0.0 | 0.9 |
| | 2 | 0.0 | 99.4 | 0.6 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 91.8 | 7.9 | 0.3 |
| | 2 | 7.0 | 93.0 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 94.9% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 99.4 | 0.3 | 0.3 |
| | 2 | 0.0 | 98.5 | 1.5 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.3% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 91.5 | 8.2 | 0.3 |
| | 2 | 6.7 | 93.3 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 94.9% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 99.7 | 0.0 | 0.3 |
| | 2 | 0.0 | 98.8 | 1.2 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 91.5 | 8.5 | 0.0 |
| | 2 | 7.0 | 93.0 | 0.0 |
| | 3 | 0.0 | 0.0 | 100.0 |
| Correct classification: 94.8% | | | | |

Table C.4: 3 gestures. Test subject 7 (left) and average over all subjects (right).

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 99.4 | 0.0 | 0.6 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.9 | 2.4 | 96.7 |
| Correct classification: 98.7% | | | | |

| M-B | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 89.5 | 6.2 | 2.4 |
| | 2 | 3.6 | 92.1 | 1.3 |
| | 3 | 0.6 | 1.0 | 98.3 |
| Correct classification: 94.9% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.5 | 0.3 | 1.2 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.9 | 1.8 | 97.3 |
| Correct classification: 98.6% | | | | |

| C.E | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 89.3 | 6.2 | 2.3 |
| | 2 | 3.6 | 92.5 | 0.9 |
| | 3 | 1.0 | 0.9 | 97.6 |
| Correct classification: 94.9% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.2 | 0.3 | 1.5 |
| | 2 | 0.0 | 99.7 | 0.3 |
| | 3 | 0.9 | 0.6 | 98.5 |
| Correct classification: 98.8% | | | | |

| C.C | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 89.3 | 5.4 | 2.5 |
| | 2 | 3.4 | 91.7 | 1.7 |
| | 3 | 1.1 | 0.9 | 97.7 |
| Correct classification: 94.9% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.8 | 0.0 | 1.2 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 0.9 | 0.9 | 98.2 |
| Correct classification: 99.0% | | | | |

| ICA All | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 89.2 | 6.2 | 2.4 |
| | 2 | 3.0 | 92.9 | 1.4 |
| | 3 | 0.7 | 0.9 | 98.1 |
| Correct classification: 95.0% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 98.8 | 0.0 | 1.2 |
| | 2 | 0.0 | 100.0 | 0.0 |
| | 3 | 1.2 | 1.2 | 97.6 |
| Correct classification: 98.8% | | | | |

| ICA Per | | Assigned gesture | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| True | 1 | 90.4 | 5.5 | 2.1 |
| | 2 | 3.4 | 92.3 | 1.3 |
| | 3 | 1.2 | 0.9 | 97.8 |
| Correct classification: 95.2% | | | | |

Table C.5: 4 gestures. Test subjects 1 (left) and 2 (right).

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 94.2 | 0.9 | 0.0 | 4.9 |
| r | 2 | 0.6 | 82.2 | 0.0 | 17.2 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.9 | 1.5 | 0.0 | 97.5 |
| Correct classification: 93.5% | | | | | |

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 98.2 | 1.8 | 0.0 | 0.0 |
| r | 2 | 0.0 | 99.7 | 0.3 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 93.5 | 1.2 | 0.0 | 5.2 |
| r | 2 | 0.6 | 81.8 | 0.0 | 17.5 |
| u | 3 | 0.0 | 0.0 | 99.7 | 0.3 |
| e | 4 | 0.9 | 1.5 | 0.3 | 97.2 |
| Correct classification: 93.1% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 98.8 | 1.2 | 0.0 | 0.0 |
| r | 2 | 0.0 | 99.7 | 0.3 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.6 | 99.4 |
| Correct classification: 99.5% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 92.0 | 0.9 | 0.0 | 7.1 |
| r | 2 | 0.6 | 81.5 | 0.0 | 17.8 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.9 | 1.5 | 0.0 | 97.5 |
| Correct classification: 92.8% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 98.2 | 1.8 | 0.0 | 0.0 |
| r | 2 | 0.0 | 100.0 | 0.0 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.6% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 94.5 | 0.9 | 0.0 | 4.6 |
| r | 2 | 0.0 | 83.1 | 0.0 | 16.9 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 1.2 | 1.5 | 0.0 | 97.2 |
| Correct classification: 93.7% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 98.2 | 1.8 | 0.0 | 0.0 |
| r | 2 | 0.0 | 99.7 | 0.3 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 93.5 | 1.5 | 0.0 | 4.9 |
| r | 2 | 0.6 | 81.5 | 0.0 | 17.8 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.9 | 1.5 | 0.0 | 97.5 |
| Correct classification: 93.2% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 98.5 | 1.5 | 0.0 | 0.0 |
| r | 2 | 0.0 | 99.1 | 0.9 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.4% | | | | | |

Table C.6: 4 gestures. Test subjects 3 (left) and 4 (right).

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 89.9 | 0.6 | 2.3 | 7.3 |
| r | 2 | 2.0 | 96.9 | 0.0 | 1.1 |
| u | 3 | 1.4 | 0.8 | 97.2 | 0.6 |
| e | 4 | 7.0 | 0.8 | 0.0 | 92.1 |
| Correct classification: 94.0% | | | | | |

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 85.8 | 10.0 | 4.2 | 0.0 |
| r | 2 | 2.9 | 87.4 | 8.1 | 1.6 |
| u | 3 | 0.0 | 5.8 | 93.9 | 0.3 |
| e | 4 | 0.0 | 1.9 | 3.9 | 94.2 |
| Correct classification: 90.3% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 87.6 | 0.6 | 2.0 | 9.9 |
| r | 2 | 3.7 | 95.2 | 0.3 | 0.8 |
| u | 3 | 2.3 | 0.3 | 96.3 | 1.1 |
| e | 4 | 5.9 | 0.8 | 0.3 | 93.0 |
| Correct classification: 93.0% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 83.9 | 11.9 | 4.2 | 0.0 |
| r | 2 | 4.2 | 89.4 | 4.8 | 1.6 |
| u | 3 | 0.3 | 6.1 | 92.6 | 1.0 |
| e | 4 | 0.0 | 1.9 | 3.2 | 94.8 |
| Correct classification: 90.2% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 84.8 | 0.8 | 2.0 | 12.4 |
| r | 2 | 1.7 | 96.9 | 0.0 | 1.4 |
| u | 3 | 2.3 | 0.6 | 96.3 | 0.8 |
| e | 4 | 6.8 | 0.8 | 0.6 | 91.8 |
| Correct classification: 92.5% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 86.5 | 7.4 | 6.1 | 0.0 |
| r | 2 | 3.9 | 83.2 | 11.3 | 1.6 |
| u | 3 | 0.0 | 5.5 | 93.5 | 1.0 |
| e | 4 | 0.0 | 1.6 | 3.9 | 94.5 |
| Correct classification: 89.4% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 87.3 | 0.8 | 2.0 | 9.9 |
| r | 2 | 2.0 | 97.2 | 0.3 | 0.6 |
| u | 3 | 1.1 | 0.6 | 97.7 | 0.6 |
| e | 4 | 7.3 | 0.8 | 0.3 | 91.5 |
| Correct classification: 93.5% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 82.9 | 12.9 | 4.2 | 0.0 |
| r | 2 | 2.9 | 88.4 | 8.1 | 0.6 |
| u | 3 | 0.0 | 5.5 | 93.2 | 1.3 |
| e | 4 | 0.0 | 1.3 | 4.5 | 94.2 |
| Correct classification: 89.7% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 88.7 | 0.3 | 1.7 | 9.3 |
| r | 2 | 1.7 | 97.5 | 0.0 | 0.8 |
| u | 3 | 1.4 | 0.8 | 97.7 | 0.0 |
| e | 4 | 6.2 | 0.8 | 0.3 | 92.7 |
| Correct classification: 94.2% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 87.1 | 8.7 | 4.2 | 0.0 |
| r | 2 | 4.2 | 86.5 | 7.1 | 2.3 |
| u | 3 | 0.3 | 5.2 | 93.2 | 1.3 |
| e | 4 | 0.0 | 1.3 | 4.2 | 94.5 |
| Correct classification: 90.3% | | | | | |

Table C.7: 4 gestures. Test subjects 5 (left) and 6 (right).

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 99.4 | 0.0 | 0.6 | 0.0 |
| r | 2 | 0.0 | 98.5 | 0.3 | 1.2 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.5% | | | | | |

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 82.7 | 7.0 | 10.0 | 0.3 |
| r | 2 | 14.5 | 85.2 | 0.3 | 0.0 |
| u | 3 | 3.0 | 0.0 | 97.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 91.2% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 99.1 | 0.0 | 0.9 | 0.0 |
| r | 2 | 0.0 | 98.5 | 0.6 | 0.9 |
| u | 3 | 0.0 | 0.0 | 99.7 | 0.3 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.3% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 84.5 | 6.4 | 8.8 | 0.3 |
| r | 2 | 12.4 | 87.3 | 0.3 | 0.0 |
| u | 3 | 4.5 | 0.0 | 95.2 | 0.3 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 91.7% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 99.4 | 0.0 | 0.6 | 0.0 |
| r | 2 | 0.0 | 98.2 | 0.6 | 1.2 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.4% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 84.5 | 6.7 | 8.8 | 0.0 |
| r | 2 | 11.8 | 87.9 | 0.3 | 0.0 |
| u | 3 | 5.8 | 0.0 | 93.9 | 0.3 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 91.6% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 99.1 | 0.0 | 0.9 | 0.0 |
| r | 2 | 0.0 | 98.5 | 0.6 | 0.9 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.4% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 82.4 | 6.7 | 10.0 | 0.9 |
| r | 2 | 10.9 | 88.8 | 0.3 | 0.0 |
| u | 3 | 3.9 | 0.0 | 95.8 | 0.3 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 91.7% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 99.1 | 0.0 | 0.9 | 0.0 |
| r | 2 | 0.0 | 98.5 | 0.6 | 0.9 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 99.4% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 85.5 | 7.0 | 7.6 | 0.0 |
| r | 2 | 11.5 | 88.2 | 0.3 | 0.0 |
| u | 3 | 6.4 | 0.0 | 93.3 | 0.3 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100.0 |
| Correct classification: 91.7% | | | | | |

Table C.8: 4 gestures. Test subject 7 (left) and average over all subjects (right).

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 78.8 | 18.5 | 0.0 | 2.7 |
| r | 2 | 9.7 | 89.1 | 0.3 | 0.9 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 1.2 | 1.5 | 97.3 |
| Correct classification: 91.3% | | | | | |

| M-B | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 89.8 | 5.5 | 2.4 | 2.2 |
| r | 2 | 4.2 | 91.3 | 1.3 | 3.2 |
| u | 3 | 0.6 | 1.0 | 98.3 | 0.1 |
| e | 4 | 1.1 | 0.8 | 0.8 | 97.3 |
| Correct classification: 94.2% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 76.7 | 16.7 | 0.0 | 6.7 |
| r | 2 | 5.2 | 93.0 | 0.0 | 1.8 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.6 | 0.9 | 98.5 |
| Correct classification: 92.0% | | | | | |

| C.E | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 89.2 | 5.4 | 2.3 | 3.2 |
| r | 2 | 3.7 | 92.1 | 0.9 | 3.2 |
| u | 3 | 1.0 | 0.9 | 97.6 | 0.4 |
| e | 4 | 1.0 | 0.7 | 0.8 | 97.6 |
| Correct classification: 94.1% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 81.5 | 15.5 | 0.0 | 3.0 |
| r | 2 | 8.8 | 90.3 | 0.3 | 0.6 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.9 | 1.5 | 97.6 |
| Correct classification: 92.3% | | | | | |

| C.C | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 89.6 | 4.7 | 2.5 | 3.2 |
| r | 2 | 3.8 | 91.1 | 1.8 | 3.2 |
| u | 3 | 1.1 | 0.9 | 97.7 | 0.3 |
| e | 4 | 1.1 | 0.7 | 0.8 | 97.4 |
| Correct classification: 93.9% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 81.2 | 16.1 | 0.0 | 2.7 |
| r | 2 | 5.2 | 93.6 | 0.0 | 1.2 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.6 | 1.5 | 97.9 |
| Correct classification: 93.2% | | | | | |

| ICA All | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 89.4 | 5.6 | 2.4 | 2.6 |
| r | 2 | 3.0 | 92.8 | 1.4 | 2.9 |
| u | 3 | 0.7 | 0.9 | 98.1 | 0.3 |
| e | 4 | 1.2 | 0.6 | 0.9 | 97.3 |
| Correct classification: 94.4% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 77.9 | 18.5 | 0.0 | 3.6 |
| r | 2 | 5.5 | 94.5 | 0.0 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100.0 | 0.0 |
| e | 4 | 0.0 | 0.9 | 1.8 | 97.3 |
| Correct classification: 92.4% | | | | | |

| ICA Per | | Assigned gesture | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 |
| T | 1 | 90.0 | 5.4 | 2.1 | 2.6 |
| r | 2 | 3.4 | 92.3 | 1.3 | 3.1 |
| u | 3 | 1.2 | 0.9 | 97.8 | 0.2 |
| e | 4 | 1.0 | 0.7 | 0.9 | 97.4 |
| Correct classification: 94.4% | | | | | |

Table C.9: 5 gestures. Test subjects 1 (left) and 2 (right).

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 88.0 | 0.3 | 0.0 | 7.1 | 4.6 |
| r | 2 | 0.0 | 96.3 | 0.0 | 1.8 | 1.8 |
| u | 3 | 0.0 | 0.0 | 98.2 | 1.8 | 0.0 |
| e | 4 | 0.6 | 0.6 | 0.0 | 96.6 | 2.2 |
| | 5 | 0.6 | 1.5 | 0.0 | 8.0 | 89.8 |
| Correct classification: 93.8% | | | | | | |

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 85.4 | 1.5 | 0.0 | 0.0 | 13.1 |
| r | 2 | 0.0 | 99.4 | 0.6 | 0.0 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.3 | 99.7 | 0.0 |
| | 5 | 3.9 | 0.0 | 1.5 | 2.4 | 92.2 |
| Correct classification: 95.3% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 85.5 | 0.6 | 0.0 | 9.2 | 4.6 |
| r | 2 | 0.0 | 95.4 | 0.0 | 2.2 | 2.5 |
| u | 3 | 0.0 | 0.0 | 99.4 | 0.6 | 0.0 |
| e | 4 | 0.9 | 0.6 | 0.0 | 96.3 | 2.2 |
| | 5 | 0.3 | 5.8 | 0.0 | 8.6 | 85.2 |
| Correct classification: 92.4% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 89.0 | 1.5 | 0.0 | 0.0 | 9.6 |
| r | 2 | 0.0 | 100. | 0.0 | 0.0 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 4.5 | 0.0 | 1.2 | 3.6 | 90.7 |
| Correct classification: 95.9% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 90.2 | 0.0 | 0.0 | 6.5 | 3.4 |
| r | 2 | 0.0 | 96.3 | 0.0 | 1.8 | 1.8 |
| u | 3 | 0.0 | 0.0 | 99.7 | 0.3 | 0.0 |
| e | 4 | 0.9 | 0.6 | 0.0 | 96.3 | 2.2 |
| | 5 | 0.6 | 2.5 | 0.0 | 9.2 | 87.7 |
| Correct classification: 94.0% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 88.1 | 1.5 | 0.0 | 0.0 | 10.4 |
| r | 2 | 0.0 | 100. | 0.0 | 0.0 | 0.0 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 3.0 | 0.0 | 0.9 | 2.7 | 93.4 |
| Correct classification: 96.3% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 87.7 | 0.3 | 0.0 | 7.4 | 4.6 |
| r | 2 | 0.0 | 92.3 | 0.0 | 1.5 | 6.2 |
| u | 3 | 0.0 | 0.0 | 99.7 | 0.3 | 0.0 |
| e | 4 | 0.6 | 0.0 | 1.2 | 95.1 | 3.1 |
| | 5 | 0.3 | 1.5 | 0.3 | 7.1 | 90.8 |
| Correct classification: 93.1% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 87.8 | 1.5 | 0.0 | 0.0 | 10.7 |
| r | 2 | 0.0 | 97.6 | 2.1 | 0.0 | 0.3 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 4.2 | 0.3 | 1.5 | 4.5 | 89.6 |
| Correct classification: 95.0% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 88.3 | 0.3 | 0.0 | 6.5 | 4.9 |
| r | 2 | 0.0 | 95.4 | 0.0 | 1.8 | 2.8 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.9 | 0.9 | 0.0 | 96.3 | 1.8 |
| | 5 | 0.3 | 2.5 | 0.3 | 6.8 | 90.2 |
| Correct classification: 94.0% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 87.8 | 1.8 | 0.0 | 0.0 | 10.4 |
| r | 2 | 0.0 | 99.4 | 0.6 | 0.0 | 0.0 |
| u | 3 | 0.0 | 0.0 | 99.7 | 0.3 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.3 | 99.7 | 0.0 |
| | 5 | 3.6 | 0.0 | 1.2 | 2.4 | 92.8 |
| Correct classification: 95.9% | | | | | | |

Table C.10: 5 gestures. Test subjects 3 (left) and 4 (right).

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 79.4 | 0.6 | 3.1 | 10.1 | 6.8 |
| r | 2 | 3.9 | 94.6 | 0.0 | 1.1 | 0.3 |
| u | 3 | 3.4 | 0.3 | 96.1 | 0.3 | 0.0 |
| e | 4 | 4.8 | 0.8 | 1.1 | 93.2 | 0.0 |
| | 5 | 21.1 | 0.3 | 0.3 | 0.3 | 78.0 |
| Correct classification: 88.3% | | | | | | |

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 87.7 | 5.8 | 2.9 | 0.0 | 3.5 |
| r | 2 | 1.6 | 69.0 | 5.5 | 0.3 | 23.5 |
| u | 3 | 0.0 | 1.9 | 91.0 | 0.3 | 6.8 |
| e | 4 | 0.0 | 0.3 | 3.2 | 91.9 | 4.5 |
| | 5 | 0.0 | 0.3 | 20.6 | 2.3 | 76.8 |
| Correct classification: 83.3% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 78.6 | 0.0 | 3.1 | 11.0 | 7.3 |
| r | 2 | 1.4 | 97.2 | 0.0 | 0.6 | 0.8 |
| u | 3 | 2.3 | 0.3 | 97.5 | 0.0 | 0.0 |
| e | 4 | 4.2 | 0.8 | 1.4 | 93.5 | 0.0 |
| | 5 | 18.3 | 0.0 | 0.0 | 0.0 | 81.7 |
| Correct classification: 89.7% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 84.8 | 9.0 | 2.9 | 0.6 | 2.6 |
| r | 2 | 2.6 | 78.7 | 1.9 | 0.3 | 16.5 |
| u | 3 | 0.3 | 2.6 | 90.0 | 0.6 | 6.5 |
| e | 4 | 0.0 | 1.6 | 2.3 | 90.0 | 6.1 |
| | 5 | 0.0 | 2.6 | 17.4 | 2.9 | 77.1 |
| Correct classification: 84.1% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 81.1 | 0.0 | 2.8 | 9.9 | 6.2 |
| r | 2 | 1.4 | 97.5 | 0.0 | 0.6 | 0.6 |
| u | 3 | 4.2 | 0.6 | 94.6 | 0.6 | 0.0 |
| e | 4 | 5.6 | 0.8 | 0.8 | 92.7 | 0.0 |
| | 5 | 25.4 | 1.4 | 0.0 | 0.0 | 73.2 |
| Correct classification: 87.8% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 85.5 | 4.2 | 5.5 | 0.0 | 4.8 |
| r | 2 | 2.9 | 69.7 | 8.7 | 1.0 | 17.7 |
| u | 3 | 0.0 | 1.3 | 91.3 | 0.3 | 7.1 |
| e | 4 | 0.0 | 0.6 | 4.5 | 90.0 | 4.8 |
| | 5 | 0.0 | 0.3 | 24.2 | 2.3 | 73.2 |
| Correct classification: 81.9% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 78.6 | 0.0 | 2.8 | 12.1 | 6.5 |
| r | 2 | 1.7 | 96.6 | 0.0 | 1.1 | 0.6 |
| u | 3 | 3.7 | 0.6 | 95.5 | 0.3 | 0.0 |
| e | 4 | 4.2 | 0.6 | 0.3 | 94.6 | 0.3 |
| | 5 | 21.1 | 0.3 | 0.3 | 0.8 | 77.5 |
| Correct classification: 88.6% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 83.5 | 9.0 | 3.9 | 0.0 | 3.5 |
| r | 2 | 2.9 | 74.2 | 2.9 | 1.0 | 19.0 |
| u | 3 | 0.3 | 1.9 | 91.0 | 0.3 | 6.5 |
| e | 4 | 0.0 | 1.0 | 3.5 | 90.6 | 4.8 |
| | 5 | 0.0 | 1.3 | 21.0 | 3.2 | 74.5 |
| Correct classification: 82.8% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 81.7 | 0.0 | 3.4 | 9.9 | 5.1 |
| r | 2 | 2.5 | 95.5 | 0.0 | 1.4 | 0.6 |
| u | 3 | 2.8 | 0.6 | 96.1 | 0.6 | 0.0 |
| e | 4 | 5.6 | 0.6 | 1.1 | 92.7 | 0.0 |
| | 5 | 24.5 | 1.1 | 0.0 | 0.0 | 74.4 |
| Correct classification: 88.1% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 86.5 | 6.1 | 4.5 | 0.3 | 2.6 |
| r | 2 | 2.9 | 71.9 | 3.9 | 0.0 | 21.3 |
| u | 3 | 0.0 | 2.6 | 91.0 | 0.6 | 5.8 |
| e | 4 | 0.0 | 1.0 | 2.9 | 90.6 | 5.5 |
| | 5 | 0.0 | 1.6 | 21.0 | 1.9 | 75.5 |
| Correct classification: 83.1% | | | | | | |

Table C.11: 5 gestures. Test subjects 5 (left) and 6 (right).

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 90.9 | 0.3 | 0.3 | 0.0 | 8.5 |
| r | 2 | 0.0 | 97.6 | 0.6 | 0.0 | 1.8 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 2.1 | 0.0 | 2.1 | 0.0 | 95.8 |
| Correct classification: 96.8% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 91.8 | 0.0 | 0.6 | 0.0 | 7.6 |
| r | 2 | 0.0 | 98.2 | 0.9 | 0.3 | 0.6 |
| u | 3 | 0.0 | 0.0 | 99.7 | 0.0 | 0.3 |
| e | 4 | 0.0 | 0.0 | 0.6 | 99.4 | 0.0 |
| | 5 | 2.1 | 0.0 | 2.1 | 0.0 | 95.8 |
| Correct classification: 97.0% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 89.4 | 0.3 | 0.3 | 0.0 | 10.0 |
| r | 2 | 0.0 | 97.3 | 1.2 | 0.0 | 1.5 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.3 | 99.7 | 0.0 |
| | 5 | 1.2 | 0.0 | 1.8 | 0.0 | 97.0 |
| Correct classification: 96.7% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 92.7 | 0.3 | 0.0 | 0.0 | 7.0 |
| r | 2 | 0.0 | 95.5 | 1.2 | 0.0 | 3.3 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 2.1 | 0.0 | 1.8 | 0.0 | 96.1 |
| Correct classification: 96.8% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 92.4 | 0.0 | 0.3 | 0.0 | 7.3 |
| r | 2 | 0.0 | 98.2 | 0.9 | 0.0 | 0.9 |
| u | 3 | 0.0 | 0.0 | 100. | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.3 | 99.7 | 0.0 |
| | 5 | 2.1 | 0.0 | 1.8 | 0.0 | 96.1 |
| Correct classification: 97.3% | | | | | | |

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 83.9 | 5.2 | 7.6 | 0.0 | 3.3 |
| r | 2 | 9.1 | 88.5 | 0.0 | 0.0 | 2.4 |
| u | 3 | 6.1 | 0.0 | 93.3 | 0.6 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 99.7 | 0.3 |
| | 5 | 0.0 | 0.0 | 0.9 | 1.8 | 97.3 |
| Correct classification: 92.5% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 80.6 | 7.0 | 8.8 | 0.0 | 3.6 |
| r | 2 | 7.9 | 90.0 | 0.3 | 0.0 | 1.8 |
| u | 3 | 5.5 | 0.0 | 94.5 | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 0.0 | 0.3 | 0.6 | 3.9 | 95.2 |
| Correct classification: 92.1% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 80.3 | 6.4 | 10.0 | 0.0 | 3.3 |
| r | 2 | 7.9 | 90.6 | 0.0 | 0.0 | 1.5 |
| u | 3 | 6.7 | 0.0 | 93.3 | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 0.6 | 0.3 | 0.9 | 5.5 | 92.7 |
| Correct classification: 91.4% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 80.0 | 6.7 | 10.6 | 0.0 | 2.7 |
| r | 2 | 8.8 | 88.5 | 0.3 | 0.0 | 2.4 |
| u | 3 | 5.5 | 0.0 | 93.9 | 0.6 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 0.6 | 0.0 | 0.9 | 2.1 | 96.4 |
| Correct classification: 91.8% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 81.5 | 5.5 | 10.0 | 0.3 | 2.7 |
| r | 2 | 8.2 | 90.6 | 0.3 | 0.0 | 0.9 |
| u | 3 | 6.4 | 0.0 | 93.6 | 0.0 | 0.0 |
| e | 4 | 0.0 | 0.0 | 0.0 | 100. | 0.0 |
| | 5 | 0.0 | 0.0 | 0.9 | 1.5 | 97.6 |
| Correct classification: 92.7% | | | | | | |

Table C.12: 5 gestures. Test subject 7 (left) and average over all subjects (right).

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 73.3 | 17.3 | 0.0 | 1.8 | 7.6 |
| r | 2 | 4.2 | 87.0 | 0.0 | 0.0 | 8.8 |
| u | 3 | 0.0 | 0.0 | 100 | 0.0 | 0.0 |
| e | 4 | 0.6 | 0.9 | 0.0 | 88.8 | 9.7 |
| | 5 | 0.6 | 0.3 | 0.9 | 1.5 | 96.7 |
| Correct classification: 89.2% | | | | | | |

| M-B | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 84.1 | 4.4 | 2.0 | 2.7 | 6.8 |
| r | 2 | 2.7 | 90.3 | 1.0 | 0.5 | 5.5 |
| u | 3 | 1.3 | 0.3 | 96.9 | 0.4 | 1.0 |
| e | 4 | 0.9 | 0.4 | 0.7 | 95.7 | 2.4 |
| | 5 | 4.1 | 0.3 | 3.8 | 2.3 | 89.5 |
| Correct classification: 91.3% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 77.0 | 17.0 | 0.0 | 2.4 | 3.6 |
| r | 2 | 4.2 | 89.7 | 0.0 | 0.0 | 6.1 |
| u | 3 | 0.0 | 0.0 | 100 | 0.0 | 0.0 |
| e | 4 | 0.6 | 1.5 | 0.0 | 91.8 | 6.1 |
| | 5 | 0.9 | 0.9 | 0.6 | 0.9 | 96.7 |
| Correct classification: 91.0% | | | | | | |

| C.E | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 83.9 | 5.0 | 2.2 | 3.3 | 5.6 |
| r | 2 | 2.3 | 92.7 | 0.4 | 0.5 | 4.0 |
| u | 3 | 1.1 | 0.4 | 97.3 | 0.2 | 1.0 |
| e | 4 | 0.8 | 0.7 | 0.6 | 95.9 | 2.0 |
| | 5 | 3.7 | 1.4 | 3.1 | 2.8 | 88.9 |
| Correct classification: 91.7% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 79.7 | 14.2 | 0.0 | 2.4 | 3.6 |
| r | 2 | 5.5 | 88.2 | 0.0 | 0.0 | 6.4 |
| u | 3 | 0.0 | 0.0 | 100 | 0.0 | 0.0 |
| e | 4 | 0.9 | 0.9 | 0.0 | 92.4 | 5.8 |
| | 5 | 0.3 | 0.6 | 0.3 | 0.6 | 98.2 |
| Correct classification: 91.7% | | | | | | |

| C.C | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 84.9 | 3.8 | 2.7 | 2.7 | 6.0 |
| r | 2 | 2.5 | 91.4 | 1.4 | 0.5 | 4.2 |
| u | 3 | 1.6 | 0.3 | 97.0 | 0.2 | 1.0 |
| e | 4 | 1.1 | 0.4 | 0.8 | 95.9 | 1.8 |
| | 5 | 4.4 | 0.7 | 4.0 | 2.9 | 87.9 |
| Correct classification: 91.4% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 75.2 | 14.5 | 0.0 | 4.5 | 5.8 |
| r | 2 | 4.5 | 85.8 | 0.0 | 0.0 | 9.7 |
| u | 3 | 0.0 | 0.0 | 100 | 0.0 | 0.0 |
| e | 4 | 0.6 | 0.9 | 0.0 | 95.8 | 2.7 |
| | 5 | 0.6 | 0.9 | 0.9 | 2.7 | 94.8 |
| Correct classification: 90.3% | | | | | | |

| ICA All | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 83.6 | 4.6 | 2.5 | 3.4 | 5.8 |
| r | 2 | 2.6 | 90.1 | 0.9 | 0.5 | 5.9 |
| u | 3 | 1.3 | 0.4 | 97.2 | 0.2 | 0.9 |
| e | 4 | 0.8 | 0.3 | 0.7 | 96.6 | 1.6 |
| | 5 | 4.1 | 0.6 | 3.8 | 2.9 | 88.5 |
| Correct classification: 91.2% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 76.4 | 17.6 | 0.0 | 3.0 | 3.0 |
| r | 2 | 3.6 | 88.8 | 0.0 | 0.0 | 7.6 |
| u | 3 | 0.0 | 0.0 | 100 | 0.0 | 0.0 |
| e | 4 | 0.3 | 0.9 | 0.0 | 94.8 | 3.9 |
| | 5 | 0.9 | 2.1 | 0.6 | 0.3 | 96.1 |
| Correct classification: 91.2% | | | | | | |

| ICA Per | | Assigned gesture | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 |
| T | 1 | 84.9 | 4.5 | 2.6 | 2.9 | 5.2 |
| r | 2 | 2.5 | 91.4 | 0.8 | 0.5 | 4.9 |
| u | 3 | 1.3 | 0.4 | 97.2 | 0.2 | 0.8 |
| e | 4 | 1.0 | 0.5 | 0.7 | 96.3 | 1.6 |
| | 5 | 4.5 | 1.0 | 3.7 | 1.8 | 88.9 |
| Correct classification: 91.7% | | | | | | |

# VITA

Luis Alberto Rivera graduated in Electronics Engineering from Del Valle University, Guatemala, in 2006. He worked for the Mathematics and Physics departments at Del Valle from 2007 to 2009. He was granted a Fulbright scholarship, and he received the Master of Science degree in Electrical Engineering from the University of Missouri in December 2011. He continued his studies at the University of Missouri, where he completed his Doctor of Philosophy degree in Electrical and Computer Engineering in the Summer of 2016. He did his graduate work at the Vision-Guided and Intelligent Robotics Lab. His research interests are in the areas of Machine Learning, Pattern Recognition and Robotic Assistive Technology. He has worked on systems and interfaces for controlling assistive technology devices such as power wheelchairs, using surface EMG signals, head motion, etc. He has also worked on methods for detecting patterns in mixed signals, with applications in Terahertz and Assistive Technology, Otolaryngology, and others.