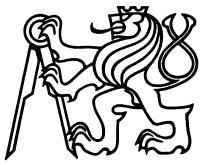
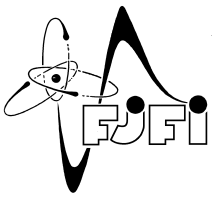


CZECH TECHNICAL UNIVERSITY IN PRAGUE
Faculty of Nuclear Sciences and Physical Engineering



Biomarker Analysis of Psychiatric Patients using EEG Signal Analysis and Machine Learning

Analýza biomarkerů psychiatrických pacientů pomocí analýzy EEG signálu a strojového učení

Diploma thesis

Author: **Miroslav Kovář**

Supervisor: **M.Sc. M.A. Sebastián Basterrech, Ph.D.**

Academic year: 2018/2019

- Zadání práce -

- Zadání práce (zadní strana) -

Acknowledgment:
Some acknoledgment here.

Author's declaration:
I declare that this research project is entirely my own work and I have listed all the used sources in the bibliography.

Prague, February 15, 2019 Miroslav Kovář

Název práce:

Analýza biomarkerů psychiatrických pacientů pomocí analýzy EEG signálu a strojového učení

Autor: Miroslav Kovář

Obor: Aplikace přírodních věd

Zaměření: Matematická informatika

Druh práce: Diplomová práce

Vedoucí práce: M.Sc. M.A. Sebastián Basterrech, Ph.D., Artificial Intelligence Center, FEE, CTU Prague

Abstrakt:

Klíčová slova:

Title:

Biomarker Analysis of Psychiatric Patients using EEG Signal Analysis and Machine Learning

Author: Miroslav Kovář

Abstract:

Key words:

Contents

Introduction

Depression is one of the most common brain disorders - it affacts 121-300 million people worldwide, and this number is expected to increase in the future [76] [67]. Although effective treatments are known, World Health Organization estimates that fewer than half of those affected receive those treatments. Major barriers include insufficient resources, lack of properly trained practitioners, inaccurate assessment and misdiagnosis. [67]

For these reasons, it is important that affordable, fast, accurate, and easy to use methods for its diagnosis are developed. Although electroencephalography (EEG) [1] may be one such method thanks to its comparatively low-cost and easy recording process, comparatively little research has been focused on this area. Non-linear dynamical analysis in particular has been proven very effective at diagnosing mental disorders, and this work is aimed at contributing to this important and relatively new topic.

In **Chapter 1**, we present some of the classical theory and methods of non-linear dynamical analysis and chaos theory, with focus on the terms used in the following text.

In **Chapter 2**, we introduce the basic concepts and terminology used in design and evaluation of convolutional neural networks.

In **Chapter 3**, we describe the methods proposed, experiments performed, and results obtained.

[1] In this work, we will use the same abreviation for electroencephalography (recording method) and electroencephalogrpham (the recorded data) where the distinction is apparent from the context.

Chapter 1

Non-linear time series analysis

The nature is constantly undergoing change. Around us, we can observe many processes evolving in time. Some of the aspects of these processes, we can measure, and attempt to discover apparent patterns in those measurements. The most simple of those patterns are periodicities, probably best exemplified, and first noticed by humans, are the motions of the sun and the moon. Weather, on the other hand, is an example of processes seemingly defying any simple description.

Those examples represent two classes of processes existent before the rise of non-linear dynamics: [6]

Deterministic process : periodic (or quasi-periodic), fully describable by its Fourier spectrum.

Stochasitc process : influenced by forces unpredictable under all circumstances.

Non-linear dynamical analysis studies a third class of processes, which are irregular, non-periodic, yet still deterministic. Every non-periodic, deterministic process is non-linear (bot not necessarily the other way around). Existence of these processes was known already in mid-19th century to J. C. Maxwell, but the field began to be developed only with the rising feasibility of numerical simulations, peaking in 1980s. [6]

1.1 EEG signal

Electroencephalography (EEG) is a noninvasive method of measuring fluctuations of electric potentials near the skull caused by synchronized firing of neurons in the upper cortical layers. Electroencephalogram is a record of these fluctuations measured over a period of time. [65]

Although EEG has significantly lower spatial resolution in comparison with other diagnostic techniques such as functional magnetic resonance sampling (fMRI) and magnetoencephalography (MEG) [88] and enables measuring only neural activity near the cortical surface, as a depression diagnostic tool, it has numerous benefits. Importantly, its significantly lower costs [96] [35], high portability, and ease of operation imply increased availability to the patients [86]. Moreover, it is perfectly noninvasive, which means less complications such as claustrophopia or anxiety [63].

Although the science of EEG signal analysis as a diagnostic tool brings compelling clinical promise as a result of the aforementioned benefits, it also presents multiple technical and conceptual challenges.

Definition 1 ([72]). *A series $\{X_t\}_{t\in\mathbb{Z}}$ is called **stationary**, if $\{X_t\}_{t\in\mathbb{Z}}$ for any set of times $t_1, t_2, \ldots, t_n$ and any $k \in \mathrm{N}$, $P[X_{t_1}, X_{t_2}, \ldots, X_{t_n}] = P[X_{t_1+k}, X_{t_2+k}, \ldots, X_{t_n+k}]$, i.e. the joint probability distribution of $\{X_t\}_{t\in\mathbb{Z}}$ is not a function of time. It is called **non-stationary**, if it is not stationary.*

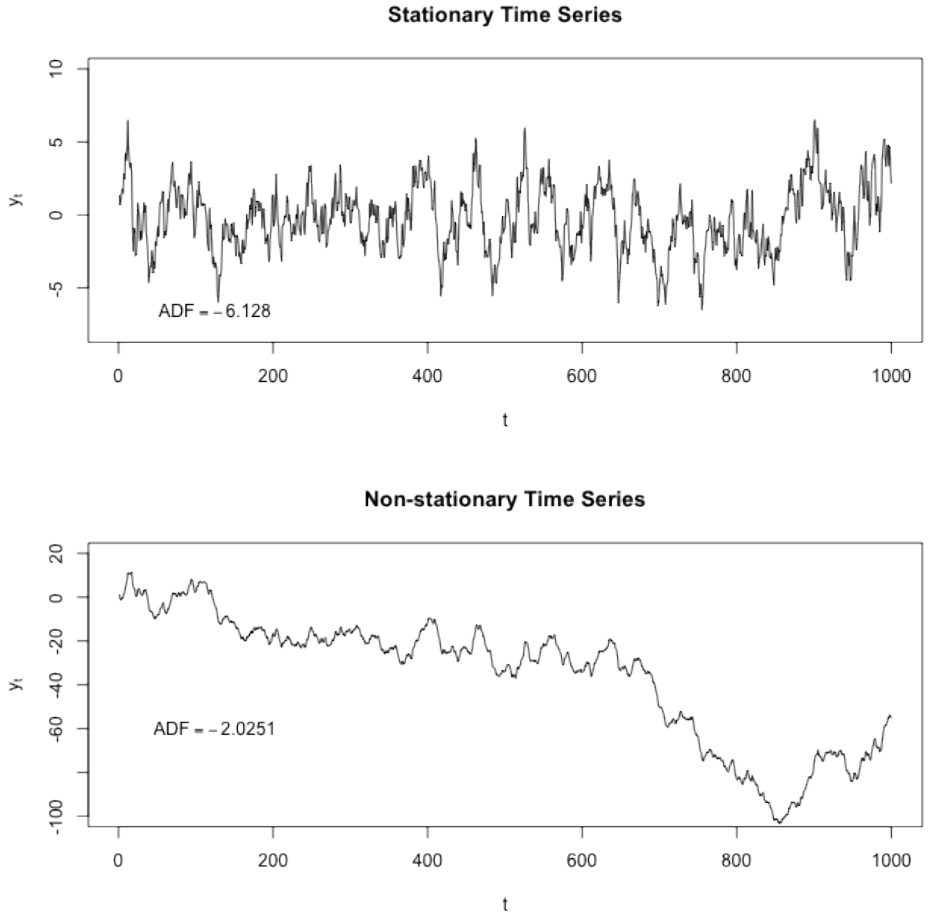


Figure 1.1: A comparison of stationary and non-stationary time series. (Courtesy: Protonk)

Definition 2 ([12]). *A series $\{X_t\}_{t\in\mathbb{Z}}$ is called (noisy chaotic)* ***non-linear****, if it satisfies the relation*

$$X_t = f(X_{t-1}) + \epsilon_t \tag{1.1}$$

for a general $f : \mathbb{R} \to \mathbb{R}$.

EEG signals are prone to be infected with *noise* due to imperfect isolation from surrounding environment. They are known to be *transient, non-Gaussian, non-stationary and nonlinear* [49] [89]. Since some patterns do not activate relative to a stimulus, a successful classifier must be able to detect a pattern *regardless of its starting time*, or find one. And finally, EEG records are relatively high dimensional - 16 electrodes sampling at 256 Hz result 4096 data points par second.

Moreover, due to the phenomenon of neural oscillations, patterns may appear in multiple frequency bands, from slow cortical potentials of δ-waves at 0.5-4 Hz, to high γ frequency band at 70-150 Hz.

Patterns of oscillatory activity in various frequency band have been linked to various mental states [17] [15] and diseases such as epilepsy [87], tremor [61], Parkinson's disease and depression [58]. Many of the diseases, including depression, share common oscillatory patterns known as thalamocortical dysrythmia, characterized by descrease in normal resting-state α (8-12 Hz) activity slowing down to θ (4-8 Hz) frequencies, accompanied by increase in β and γ (25-50 Hz) activity. [95]

1.2 Limitations in application to EEG

Some authors suggests that the since most plausible research target for explaining the brain dynamics are the assemblies of coupled and synchronously active neurons, and since majority of those assemblies are describable by non-linear differential equations, principles derived from nonlinear dynamics are applicable to characterization of these neuronal systems. [49]

The approach of estimating a finite embedding dimension, however, has been doubted by some of the most prominent figures in the field of non-linear dynamical analysis, such as the originators of Grassberger-Procaccia algorithm. There is very little evidence for the seemingly improbable hypothesis that such complex system with many extrinsic influences and interactions, such as the brain, would exhibit a level of comlexity comparable to e.g. a Lorenz system. Presumably, the the observed estimates of low dimension are due to artifacts or limited data size. [32] [73]. However, as we will see in Section 1.7, the techniqes derived from these theories still provide some useful information and are successfully applied in many practical situations. Therefore, it seems to be the case that indeed, brain dynamics are much more complex than we are forced to assume based on the theory, but non-linear dynamical analysis still manages to capture some of its important aspects.

1.3 Dynamical systems

Definition 3 ([6]). *Assume that state of a system can be fully described by a finite set of d variabes, such that each state corresponds to a point $\xi \in M$, where M is a d-dimensional differentiable manifold. Then we will call M a (true)* ***state space*** *or, equivalently, a (true)* ***phase space****, and d its (true)* ***dimension****.*

Although in this study, we will only consider Euclidean M, the true state space is needs not necessarily be Euclidean. For example, if some of the state variables are angles, the state space exhibits toroidal topology. However, any topological manifold is locally Euclidean [55] and, since, in EEG signal analysis both M and d are unknown, we have no alternative but to work in Euclidean M.

Definition 4 ([6]). *Let $\xi : \mathbb{R} \to \mathbb{R}^d$ be an $d \in \mathbb{N}$ dimensional state (phase) space vector dependent on time, and* $\mathbf{F}$ *a smooth vector field in $\mathbb{R}^d$. A* ***deterministic dynamical system***[1] *is described by a set of d first-order differential equations*

$$\frac{d}{dt}\xi(t) = \mathbf{F}(\xi(t)), \qquad t \in \mathbb{R}_0^+,$$

such that there exists a mapping $f^t : M \to M$ satisfying [2]

$$\xi(t) = f^t(\xi(0)).$$

We will call this mapping ***state evolution function****, and vector field* $\mathbf{F}$ ***dynamics of the system****. We call the system linear if* $\mathbf{F}$ *is a linear vector field.*

In late 1800s, H. Poincare developed a geometric approach to analyzing the stability (asymptotic evolution) of these systems via examination of the solution $(\xi_1(t), \xi_2(t), \ldots, \xi_d(t))$ as a *trajectory* in the phase space M (assuming the solution is known, e.g. measured). These ideas were later extended into deeper understanding of chaos in dynamical systems. [90]

In general, any system with temporally changing state is dynamic. A *deterministic* dynamical system is describable by a model giving precise transition of a system from one state to another in time. This means that total description of system's evolution in its phase space (its *trajectory*) is given by the initial state and a set of equations $\mathbf{F}$ (if $\mathbf{F}$ satisfies certain reasonable properties given by the uniqueness theorem). With *stochastic* dynamical systems, such mapping is not possible, since these transitions are not given precisely.

A non-linear dynamical system is a system where the differential equations describing its dynamics are non-linear. Unlike in a linear system, changes in the initial state of a non-dynamical system are allowed to have a non-linear relationship to the state space trajectory of the system. [49]

It is important to note the obvious fact that in the case of EEG signal analysis, it is not possible to measure the true state of the system $\xi(t)$. In fact, the observed variables are only a function of the true state of the system, $s(\xi(t))$ for some (generally non-invertible) measurement function $s : \mathbb{R}^d \to \mathbb{R}^n$, where $n \ll d$. Morover, the time between subsequent measurements is limited by a sampling frequency and the values of the variables themselves are taken and stored with a limited precision.

Add a few examples (Lorenz, Rossler, Mackey-Glass). Create my own plots instead of reusing.

1.3.1 Recurrence plot

When presented with a task of finding regularities in seemingly chaotic data, one possible approach is analysing at least approximate repetitions of simple patterns, which can be further used for reconstruction of more complicated rules. Recurrence plot is a method of visualizing obtained state-space trajectory segments in relation to each other to achieve this goal. Furthermore, it can be used to test necessary conditions for validity of dynamical parameters derivable from a non-linear time series such as the information dimension, entropy, Lyapunov exponents, dimension spectrum, etc. The information contained in recurrence plots is not easily obtainable by other known methods. [21]

[1] In this work, we are going to assume that brain is a deterministic dynamical system, and that any stochastic component is small and does not change non-linear properties of the system. Thus, by the term dynamical system, we will always mean a deterministic dynamical system.

[2] This condition is equivalent to satisfaction of the assumptions of the uniqueness theorem of differential equations.

Definition 5 ([21]). *Let N be the length of given time series, $\mathbf{s}_i$ for $i \in \{1, 2, \ldots, N\}$ be a i-th delay vector of any integer embedding dimension, $\|\cdot\|$ a norm, $\Theta(\cdot)$ a Heaviside step function, and $\epsilon \in \mathbb{R}_0^+$ a tolerance parameter. Then,* ***recurrence plot*** *is the matrix*

$$M_{ij} = \Theta(\epsilon - \|\boldsymbol{s}_i - \boldsymbol{s}_j\|)\,. \tag{1.2}$$

In other words, M_{ij} is a symmetric[3] binary NxN matrix, where $M_{ij} = 1$ when i-th and j-th points of the reconstructed trajectory enter each other's ϵ neighborhood. Since those points are, in fact, times, recurrence plots are a way of visualizing subtle time correlation information.

The essential drawback of recurrence plot is their size - it is quadratic in the length of the time series. A simple way of reducing its dimension is to partition the time series into disjoined segments, and let M_{ij} represent the distance between those two segments. This is known as **meta-recurrence plot**. [48]

In [21], J. Eckmann et al. analyzed patterns typically observed in recurrence plots and distinguished between large-scale *typology* and small-scale *texture*. Moreover, they were able identify multiple different patterns easily distinguishible by the human eye typical of dynamical systems with distinct properties. This work was furhter extended in [60].

A more objective approach to analyzing recurrence plots is an ensemble of techniques group under the term Recurrence Quantification Analysis (RQA). Using these techniques, a number of scalar measures can be used to quantify properties of recurrence plots. An important ingredient for computation of these measures is the distribution of lengths of diagonal lines in the plot. It can be shown that this distribution is directly related to correlation dimension, which we will cover in Section 1.5.2. [60]

1.3.2 Nonstationarity

Nonstationarity is a phenomenon which considerably complicates practical analysis of dynamical systems. All the techniques presented in this text assume stationary process, since this assumption is a prerequisite to deterministic chaos. [44] We will call system **nonstationary** if the dynamics of the system are influenced by causes lying outside of them (and **stationary** if the opposity is true). In ergodic theory (study of the invariant measures of dynamical systems), the concept of stationarity is defined more rigorously. However, these definitions are not suited numerical applications. [6] However, a relevant subset of nonstationary systems can be defined more explicitly:

Definition 6 ([6]). *A dynamical system is called* ***nonautonomous*** *if its dynamics* $\mathbf{F}$ *are explicitly dependent on time:*

$$\frac{d}{dt}\xi(t) = \mathbf{F}(\xi(t), t), \qquad t \in \mathbb{R}_0^+.$$

No reliable tests for nonstationarity in this strong sense exist. There is another common definition of a stationary process (sometimes referred to as weak stationarity). A process is called **weakly stationary**, if all statistical second-order quantities (like mean, variance, and power spectrum) are independent of the absolute time, and at most function of relative times. [44]

This weaker definition employs only linear quantites, and is therefore not strictly suitable for nonlinear time series analysis. On the other hand, statistical tests of this property exist. In this text, we use the following test discussed by H. Isliker and J. Kurths in [44].

This technique attempts to approximate a projection of so called *physical invariant measure* ρ defined as [23]

$$\rho \coloneqq \lim_{T\to\infty} \frac{1}{T} \int_0^t \delta_{\mathbf{x}(t)}\, \mathrm{d}t$$

[3] Although this is true for our definition, it may not be true for an alternative definition using a more general topology instead of a norm. For example, each point may have been assigned its own ϵ-neighborhood.

into one coordinate of the state space (given by the time series). Loosely speaking, this measure quantifies "how often" are different subsets of the state space visited over infinite time. In other words, it gives a probability that a randomly chosen point on a trajectory will happen to belong to a given subset "after enough time passed".

This measure is related to computation of correlation dimension. Mention it in corresponding section.

Since this measure is ergodic [4], the ergodic theorem basically states that the space and time averages are equal almost everywhere, i.e.

$$\int_{\text{statespace}} f(\mathbf{x})\rho(\mathrm{d}\mathbf{x}) = \lim_{T\to\infty} \frac{1}{T} \int_0^T f(\mathbf{x}(t))\,\mathrm{d}t$$

for any $f \in C$ defined on the state space.

Let x_1 represent the measured quantity, and N be the length of the time series. The range of the time series is divided into K intervals $[x_1^{(k)}, x_1^{(k+1)}]$, $k = 1, 2, \ldots, K.$, such that the interval boundaries are K-quantiles of the distribution of the values of the time series (i.e. application of the quantile function of the distribution to the values $1/K, 2/K, \ldots, (K-1)/K$), and the number of values falling into each of those intervals is counted:

Is this confusing? Should I just say that they intervals are "equiprobable"?

$$\begin{aligned} n_k &:= \#\{x_1^{(k)} \le x_1 \le x_1^{(k+1)}\} \\ &\approx \sum_{x_1} \int_{x_1^{(k)}}^{x_1^{(k+1)}} \delta(x - x_1)\,\mathrm{d}x \\ &= \sum_{x_1} \chi_{[x_1^{(k)}, x_1^{(k+1)}]}(x_1), \end{aligned}$$

where $\xi_{[a,b]}$ is the characteristic function of the set $[a, b]$. The density over the entire series is then approximated by a histogram with K bins as

$$p_k^{\text{all}} = \frac{n_k^{\text{all}}}{\sum_k n_k^{\text{all}}}.$$

If the system is stationary, then the distribution for the first half of the time the same. Hence, this distribution (with the same intervals) is computed for the first half of the time series (n_k^{half}). Then, the two probability distributions are compared using the ξ^2-test:

$$\chi^2 := \sum_k \frac{(n_k^{\text{half}} - Zp_k^{\text{all}})^2}{Zp_k^{\text{all}}},$$

where $Z = \lceil N/2 \rceil = \sum_k n_k^{\text{half}}$. [44]

1.3.3 Attractor

Depending on the properties of $\mathbf{F}$, there are several possibilities of how the system might evolve when as $t \to \infty$. In the following, we will focus on so called dissipative dynamical systems.

Definition 7 ([48]). *A dynamical system is called dissipative, when it is the case that*

$$E[\mathrm{div}\mathbf{F}] < 0, \tag{1.3}$$

where the expectation is taken over the state space M. In other words, average state space volume of a set of initial conditions of non-zero measure is contracted as the system evolves.

[4] This means, loosely, that it is "decomposable" into several different pieces, each again invariant.

For these systems, after sufficient passage of time, all future states will continue evolving on a bounded, time-invariant subset of M. This subset is a geometrical object called an **attractor**. Example of four basic attractors can be seen on Figure 1.2.

Since most physiogenerated signals are chaotic, their analysis is concerned primarily with *chaotic* (strange) *attractors*. These attractors are relatively complex, characteristic of dynamical systems with extending volumes in some directions. This property results fast divergence of two initial states, one of which has nonzero component in the direction of growth, i.e. sensitive dependence on the initial conditions. However, since atractors are bounded, the divergence eventually stops and the two trajectories fold together. This continuous expansion and folding may create an attractor with a *fractal structure* (an example of such an attractor is shown on Figure 1.3). [6] For our purposes it is sufficient to say that this means that these attractors can be characterized as having (quantifiable) self-similarity.[5] However, the following definition related to fractals will be useful in Section 1.4.1:

Definition 8 ([25])**.** *Let F be any non-empty bounded subset of $\mathbb{R}^n$, and let $N_\epsilon(F)$ be the smallest number of sets of diameter at most ϵ which can cover F. Then, the* ***box-counting dimension*** *(also known as Minkowski–Bouligand dimension) is defined as*

$$d_0(F) = \lim_{\epsilon \to 0} -\frac{\log N_\epsilon(F)}{\log \epsilon}, \tag{1.4}$$

if it exists.

Intuitively, the number of mesh cubes of side ϵ intersecting F gives an indication about how irregular the set is when inspected at scale ϵ, and the box-counting dimension reflects "how rapidly" the irregularities develop as $\epsilon \to 0$. [25]

1.4 State space reconstruction

Broadly, one possible approach to non-linear time series analysis consists of the following steps:

1. reconstruction of the attractor of given system from recorded data,
2. characterization of the reconstructed attractor,
3. checking validity of the results with surrogate data testing. [89]

Saying dynamics is not true. We are not reconstructing the vector field **F**.

Connect this to the content of this section. Expand on the steps.

1.4.1 Embedding

In the previous section, we have introduced a concept of state space of a dynamical system. In the case of EEG analysis, however, our observations do not directly form a state space object, but a set of time series (a sequence of scalar measurements), one for each electrode. Moreover, it is necessary to deal with the fact that our data, however rich, rarely represent complete information about the studied system. In the case of EEG signals, the complete state of the system at any moment is determined by many variables, and the sensors are only able to collect traces of their cumulative effects (and noise). So we are confronted with a problem: how to convert this data into state space trajectories? This procedure is called *state space reconstruction*.

[5] Cantor set being a canonical example of self-similarity.

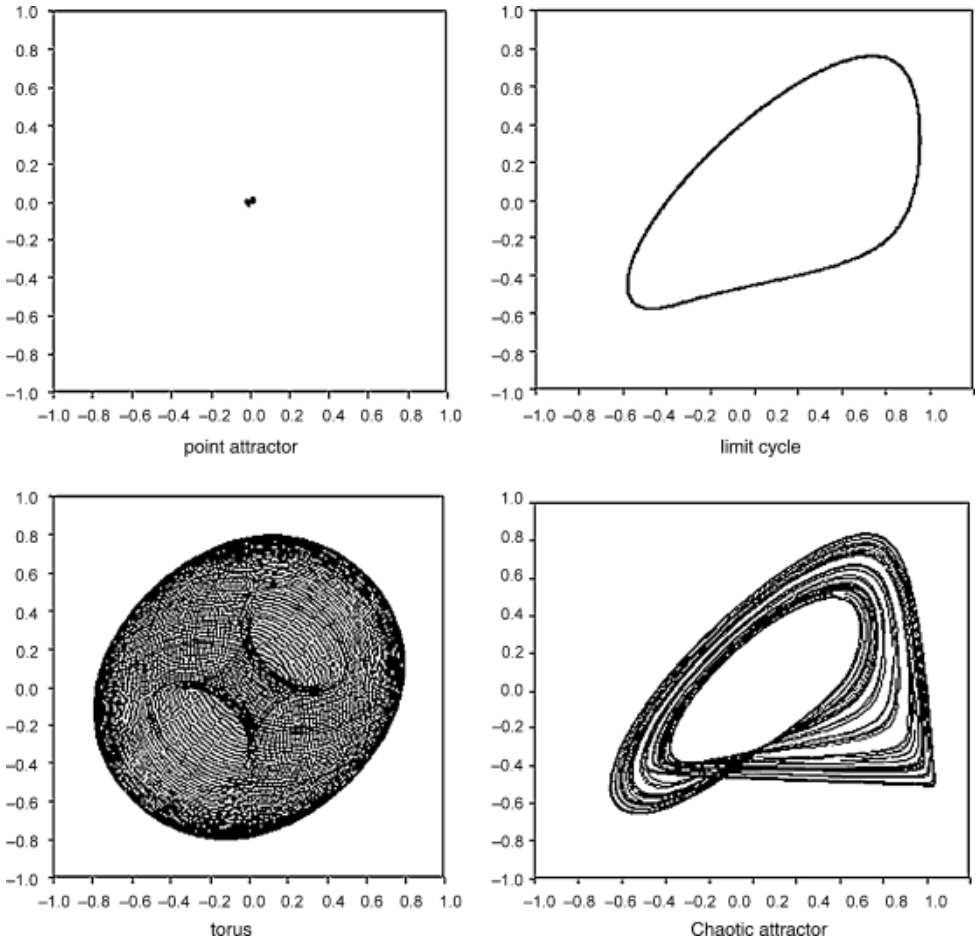


Figure 1.2: Visualization of four common attractor types (units are arbitrary). Left to right, top to bottom: **Point attractor** is the only type of attractor of linear deterministic dissipative systems. It consist of a single final state to which all points from the corresponding region of attraction evolve to. **Limit cycle** corresponds to a periodic dynamical system. It is formed by set of states visited periodically, consituting a trajectory through the state space. **Torus attractor** corresponds to a quasi-periodic dynamical system, resulting (in this example) from a superposition of two periodic oscillations. **Chaotic** (strange) **attractor**, characteristic of dynamical systems with extending (instead of shrinking) volumes in *some* directions. Corresponding dynamical system may appear stochastic, yet still is completely deterministic. [6] ([89])

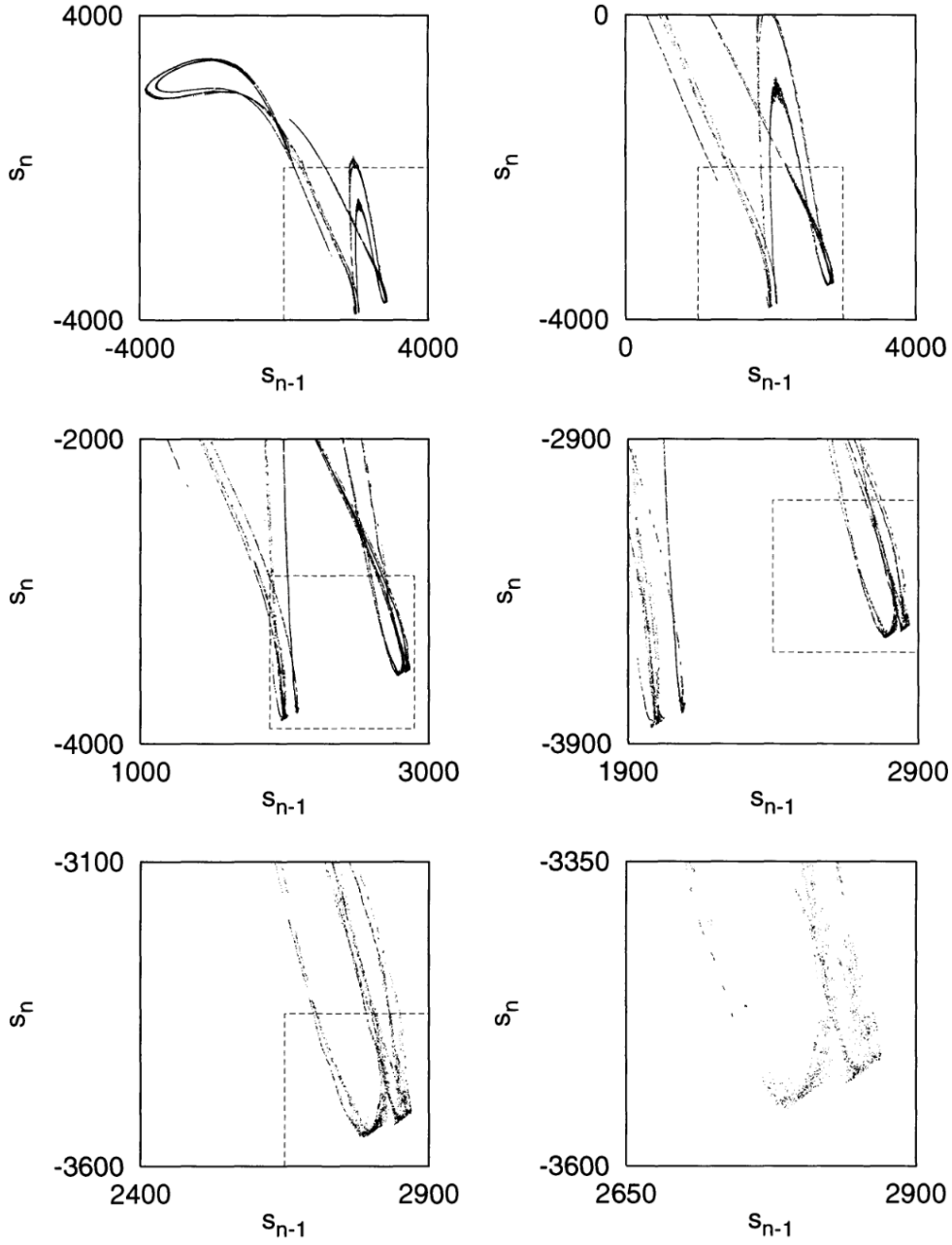


Figure 1.3: Noise-reduced visualization of successive enlargements of highly self-similar attractor. ([48])

To this goal, let $\mathbf{s}_n$ be the reconstructed vector we are trying to find, and let us have a time series of scalar measurements of a quantity depending on the current state of the system:

$$x_n = s(\xi(n\Delta t)) + \eta_n(n\Delta t)\,, \tag{1.5}$$

where ξ is a state space vector, $s(\cdot)$ is a measurement function and η_n is a measurement noise. Furthermore, let us consider a function $\Phi : M \to \mathbb{R}^m$, such that $\mathbf{x}_n = \Phi(\xi(n\Delta t))$. Such function is called an **embedding**. In the following, we will discuss what properties does Φ have to satisfy so that it provides useful information about the true state space trajectories.

Before we do that, let us mention the following. As we have stated in Section 1.3, our observations are formed by application of non-invertible measurement function $s : \mathbb{R}^d \to \mathbb{R}^{d'}$, $d' \ll d$, to the true states of the system. Aside from being a projection, s may be also be a distortion. Therefore, it might seem impossible to reconstruct the true state space trajectory and this indeed may be the case in some situations. On the other hand, there are quantities invariant under distortion which may be preserved. [6] Moreover, if our goal was to study only the attractor properties, perfect reconstruction may not even be desirable in the case that the attractor dimension is smaller than the dimension of the original space [48].

Firstly, note that we assume the studied dynamical system to be deterministic. If our reconstructed embedded space is to represent the true state space, evolution of any state on every trajectory we observe in the embedded space should depend only on its current state. Therefore, we may reasonably require Φ to be one-to-one, i.e. contain no intersections.

Secondly, since many of the attractor properties we care about (such as correlation dimensions, Lyapunov exponents, etc.) are only invariant under smooth non-singular transformations, in order to preserve these properties in the embedded space, we may require Φ to preserve the differential structure of the state space M. This corresponds to the tangent space $D\Phi$ also being a one-to-one mapping.

Add images illustrating these two conditions.

1.4.2 Method of time delays

There are two common approaches to the problem of state space reconstruction for EEG time series data:

Time delay embedding : state space is reconstructed separately for each time series.

Spatial embedding : each time series corresponds to a coordinate of the state space vector.

Int the following text, we will focus on the first one, because we are not using the second one in this thesis, and it has been widely criticised.

Add some citations.

It had been already known since 1936, that every n-dimensional differentiable manifold can be embedded in $\mathbb{R}^{2n+1}$, and that the set of such embeddings is open and dense in the space of generic smooth maps, which is known as Whitney's theorem. [98] [6]) In other words, $2n + 1$ independent measurements of a n-dimensional system can be uniquely mapped to a $2n + 1$ dimensional space, hence each such $2n + 1$ dimensional vector identifies identifies state of the system perfecly, thus reconstructing the true state space.

Time delay embedding is a technique of state space reconstruction, which achieves the same goal, but with a single measured quantity. It was first introduced into the field of non-linear dynamical system analysis by N. H. Packard in 1980 (although it was already being used in different fields in 1950s [6]). Studying the Rossler system, Packard noticed that by sampling a single coordinate, he was able to obtain

[6] The second part of the theorem is a consequence of the fact that two hyperplanes with dimensions d_1 and d_2 in m-dimensional space are likely to intersect if $d_1 + d_2 \geq m$.

a faithful phase-space representation of the original system by simply using a value of a coordinate with its values at two previous times. [68] In other he demonstrated numerically that past and future measurements of one variable contain information about the unobserved variables and can be used to define the present state.

In particular, for each time t, we define an embedding window τ_w, and use measurements obtained at times t' for $t - \tau_w \leq t' \leq t$. To this goal, we use m measurements, τ elements apart. Here, τ is called *lag* or *time delay*, and is measured in number of samples[7]. Using the notation of 1.5, the time delay reconstruction is then formed by the following vectors:

$$\mathbf{x}_n = (x_{n-(m-1)\tau}, x_{n-(m-2)\tau}, \ldots, x_{n-\tau}, x_n), \tag{1.6}$$

for $n > (m-1)\tau = \tau_w$. [48]

A year after Packard's discovery, in [91], F. Takens has proved theoretically that the attractor reconstructed using this method may have the same dynamical properties (entropy, dimension, Lyapunov spectrum) as attractor of the original system under some conditions. Takens delay embedding theorem is an important result of non-linear time series analysis and can be stated as follows:

Theorem 1 ([91])**.** *Let M be a compact[8] smooth manifold specifying the state space of a deterministic dynamical system of dimension $d \in \mathbb{N}$, $s : M \to \mathbb{R}^n, s \in C^2$ a smooth measurement function, $f^t : M \to M, f \in C^2$ a set smooth diffeomorphic state evolution functions for $t \in \mathbb{R}$. Then the set of maps $\phi_{(s,f^t)} : M \to \mathbb{R}^{2d+1}$, defined by*

$$\phi_{(s,f^t)}(x) = (s(\xi), s(f^{-\tau}(\xi)), \ldots, s(f^{-2d\tau}(\xi))), \tag{1.7}$$

for which Φ is an embedding is an open and dense set in the space of maps satisfying the assumptions above.

This idea has a simile in the existence theorems in the theory of differential equations, which say that a unique solution exists for each $x(t), \dot{x}(t), \ddot{x}(t), \ldots$. For example, in many body dynamics under Newtonian gravitation, knowledge of a body's position and momentum is sufficient to uniquely determine its future dynamics. [84]

Taken's theorem, although of theoretical importance, is not necessarily useful in practice, since even dense sets can have measure zero. Moreover, it is restricted to smooth manifolds. An add came ten years later, when T. Sauer both generalized Takens' result as follows (in a simplified form):

Theorem 2 (Sauer, [83])**.** *Let A be a compact fractal with box-counting dimension d_A, and let A be a subset of a m-dimensional manifold. Then*

$$\{\Phi : A \to \mathbb{R}^m | \Phi \in C^1, m > 2d_A\} \text{ is an embedding with probability } 1.$$

In conclusion, Theorem 1 and Theorem 2 together ensure that when m is chosen such that $m > d_A$i (which may be a considerable reduction in dimension compared to $m \geq 2d+1$), then Φ a true embedding of the underlying attractor for almost any τ (note only sufficiency of the result - $\mathbf{x}_n$ may be an embedding even for smaller m).

A fascinating consequence of Theorem 2 when applied to a sequence of measurements recorded from a physical system is that a successfully reconstructed attractor does not describe the time series, but the system itself. In the words of Theiler: "If one believes that the brain (say) is a deterministic system, then it may be possible to study the brain by looking at the electrical output of a single neuron. This example is an ambitious one, but the point is that the delay-time embedding makes it possible for one to analyze the self-organizing behavior of a complex dynamical system without knowing the full state at any given time". [92]

[7] Some authors use the time units $\tau\Delta t$, where $\Delta t = t_s = 1/f_s$ is the sampling period.

[8] This theorem can be proved for M non-compact provided less restrictions are imposed on s.

1.4.3 The effects of noise

Although these theoretical results are important to know about, they all make practically unrealistic assumptions, such as infinite amount of data and infinite measurement precision, and absence of noise. Moreover, practical applications present further challanges, such as presence of noise.

Several factors complicate successful reconstruction from real-world, experimental data: [19]

Observational noise. Given a reconstructed vector $\mathbf{x} \in \mathbb{R}^m$, there is a (approximatly Gaussian shaped in natural scenarios) distribution $p(\mathbf{x})$ in the reconstruction space due to the noise term in equation (1.5). [6]

Dynamic noise (nonstationarity). External influences perturb the system, which consequently appears nondeterministic.

Estimation error. Estimation of the dynamics of the system is performed using only limited amount of data.

Quantization error. The measured analogue quantity is converted and stored as a number with only finite number of bits.

Moreover, different reconstructions can amplify the already present noise to varying degree. In [19], Casdagli et al. provide a quantitative way of analyzing this amplification, and, by extension, of insight into selection of embedding parameters so that the noise amplification is minimized.

1.4.4 Time delay selection

A careful reader might have noted that the results of theorems in Section 1.4.2 do not depend on the value of the delay τ.[9]. Embeddings with the same value of the embedding dimension m, but different values of τ are theoretically equivalent. In practice, however, some theoretically sound time delay reconstructions may fail to be embeddings. Although some researchers propose that the only important parameter is the length of the embedding window $\tau_w = \tau(m-1)$ [52], as we will see, the choice of time delay has effects independent of the choice of embedding dimension, and vice versa.

For example:

1. The embedding may fail to be a one-to-one map due to finite precision, or presence of noise in the data. [6]
2. Highly chaotic systems with large Lyapunov exponents (see Section 1.5.1) and large dimension, projection to a low dimensional time series causes explosion in the noise amplification. As a result, this imposes limits on short term predictablity and state space reconstruction may become impossible. Such systems should be treated as operationally stochastic. [19]
3. It was shown that increasing τ leads to rise in entropy. [47]
4. Deterministic behavior can be observed only when τ_w is smaller than the time scale of the foldings naturally produced as result of time embedding.
5. If the values of τ are *too small* in comparison to the typical time scales of the series (measured e.g. by mean period), then the successive elements of reconstructed state space vectors become almost equal. This effect is often called *redundance*. Since $x_t \approx x_{t+\tau}$, the reconstructed attractor will concentrate along the main diagonal (see Figure 1.4, left hand side). Moreover, in this case, the effect of noise is amplified. [19]

[9] This is because of the fact that the measurements are infinitely precise. [19]

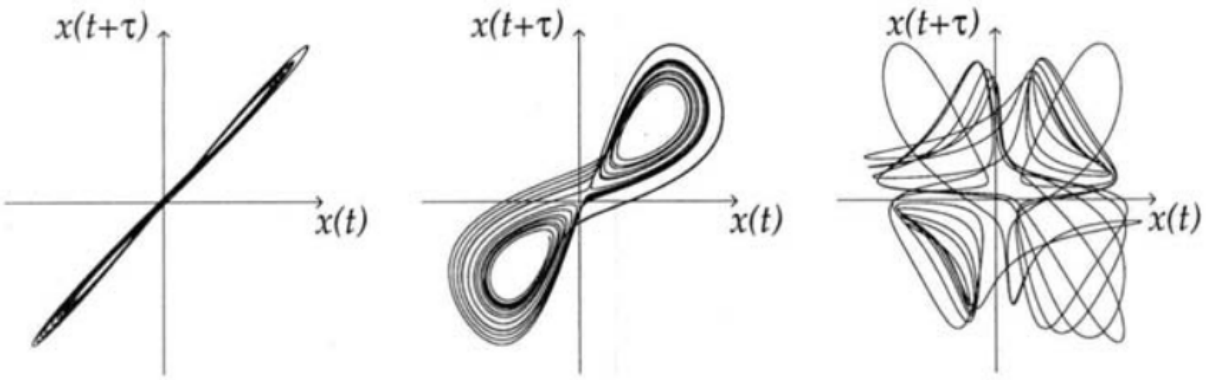


Figure 1.4: Time delay reconstructions of the Lorenz attractor for different values of τ. Figure on the left hand side shows choice of small τ and represents the case of redundance - the states concentrate along the main diagonal. Figure in the middle shows a successful reconstruction (although not an embedding, for which $m \geq 3$ is required). Figure on the right hand side shows a choice of large τ and represents the case of irrelevance - the reconstruction lacks apparent structure. ([6])

6. If the values of τ are *too large*, the successive elements in the reconstructed vector are almost independent. This effect, called *irrelevance* or *overfolding* is even magnified if the underlying attractor is chaotic, since deterministic correlations between states are lost even at very small time scales, i.e. even measurements performed at time t and $t + \tau$ for very small τ may be already unrelated. The reconstructed attractor will form a seemingly random clound in $\mathbb{R}^m$ - thus the reconstructed attractor may appear complex, even if the true attractor is simple (see Figure 1.4, right hand side).

In summary, picking the proper value of τ is a balancing act between redundance and irrelevance. It is important to minimize excessive foldings, and extreme closeness between adjacent points on the trajectory (ideally, the distances between points is same in the reconstructed as in the true space).

1.4.4.1 Autocorrelation

From the above, we understand that statistical non-correlation between values of coordinates of the reconstructed vectors $\mathbf{x}_n$ are desirable property of a time delay embedding. Thus, a natural method of estimating the optimal time delay is studying the *autocorrelation function* A, and picking the first τ where $A(\tau)$ decays below a threshold value - commonly used are $A(0)/e$ [89], $1 - A(0)/e$ [48], or even the first local minimum [5, 1] or the first 0 crossing [48].

Definition 9 ([48]). ***Autocorrelation*** *$A : \mathbb{R} \to \mathbb{R}$ for time delay τ is given by*

$$A(\tau) = \frac{E[(x_i - \overline{x})(x_{i-\tau} - \overline{x})]}{\sigma^2},$$

where $\overline{x}$ is the mean of the time series, and σ^2 is its standard deviation.

Computing the autocorrelation function is not only useful for examining the stationarity of the time series, but it also gives a geometrical insight into the shape of the attractor: if we approximate the cloud of reconstructed vectors $\mathbf{x}_n \in \mathbb{R}^m$ by an ellipsoid, lengths of its semi-axis are given by the square root of the eigenvalues of its auto-covariance matrix. In two dimensions, zero of the covariance matrix corresponds to those eigenvalues being equal, i.e. x_t and $x_{t-\tau}$ being completely uncorrelated. [48] An

obvious objection is that correlation between x_t and $x_{t-\tau}$ says nothing about correlation between x_t and $x_{t-2\tau}$, etc. Thus, this method, since it computes correlations only between two successive coordinates, is generally useful only for low dimensional systems.

Autocorrelation also provides a lower bound for τ in the following sense. If the data is noisy, vectors formed by time delay embedding procedure are practically meaningless, if the variation of the signal in the time covered in the time window $\tau_w = (m-1)\tau$ is less the the variation of noise. This means that τ should be selected such that $A(\tau) > A(0) - \sigma^2_{\text{noise}}/\sigma^2_{\text{signal}}$. [48]

1.4.4.2 Delayed mutual information

Another commonly used method is to use the first minimum of the *time delayed mutual information*. [28]

Definition 10 ([48])**.** *Let probability density of the values of a time series be split into ϵ-wide histogram bins. Let p_i be the probability that a signal assumes value in i-th bin of the histogram, and let $p_{ij}(\tau)$ be the the probability that x_t is in a bin i and $x_{t+\tau}$ is in a bin j.* ***Delayed mutual information*** *$\mathcal{I}_\epsilon$ for time delay τ is defined as*

$$\mathcal{I}_\epsilon(\tau) = \sum_{i,j} p_{ij}(\tau) \ln p_{ij}(\tau) - 2 \sum_i p_i \ln p_i.$$

In other words, time delayed mutual information the average mutual information between measurements obtained by the original time series and its τ-shifted (time delayed) counterpart. The optimal τ is usually selected as $\arg\min_\tau \mathcal{I}_\epsilon(\tau)$.

Although this approach yields coordinates independent in a more general sense than simple linear independence provided by the autocorrelation function, the same criticism applies: minimum dependence between x_t and $x_{t-\tau}$ says nothing about dependencies between other coordinates. Again, using this method is justifiable only for two-dimensional reconstructions. However, delayed mutual information has been generalized for multiple dimensions by its proponent A. M. Fraser using multidimensional distributions into a concept he called *redundancy*, which basically measures the degree to which the reconstructed vectors accumulate around the bisectrix of the embedding space. [27]

Another criticism of delayed mutual information that some systems exhibit slowly decaying mutual information which has no minima. [59]

1.4.4.3 Average displacement from diagonal

Average displacement from diaognal is a simple technique which simply measures the average distance of the embedding vectors from their original location:

$$\text{ADFD}(m,\tau) = \frac{1}{N_{(m,\tau)}} \sum_{i=1}^{N_{(m,\tau)}} \|\mathbf{x}_i^{(m,\tau)} - \mathbf{x}_i^{(m,0)}\|,$$

where $\mathbf{x}_i^{(m,\tau)}$ is the i-th vector of time delay embedding with embedding dimension m and time delay τ.

Rosenstein et al. presented multiple methods for quantifying expansion from the main diagonal, and found ADFD to be the most computationally efficient, robust to noise, and accurate. [80] They also experimentily identified optimal τ as the one for which the slope of ADFD drops below 40% of its initial value.

1.4.4.4 Singular values analysis

All the approaches described so far address the issue of irrelevance, but not that of redundance. In fact, based mostly on empirical, rather than the most time delay estimation techniques optimize for the following criteria [10]: [52]

1. The reconstructed attractor must be expanded from the diagonal.
2. The components of the reconstructed vector $\mathbf{x}_n$ must be uncorrelated.

Those criteria are noticeably similar, and bias towards larger estimates of τ. This leads many authors to suggest more advanced techniques, such as generalized delayed mutual information mentioned above, or some of those introduced in the following text.

Principal component analysis, in particular, can be used to measure the volume occupied by the reconstructed attractor. Both overfolded and redundant attractors may be marked by low volume. [6]

Given a fixed embedding dimension m, the corresponding m singular values as a function of τ contain information about the degree of extension of the embedded vectors in the m directions in the reconstructed space. Rapid increase followed by rapid decrease of some singular values accompanied by the opposite behavior of others indicate a collapse of the attractor. Also, high number of large singular values is an indicator of volume of the reconstructed attractor.

If we assume, without loss of generality, that the time series is standardized and denote

Is this so?

$$\mathbb{X} := \begin{pmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \cdots \\ \mathbf{x}_{N_{(m,\tau)}}^T \end{pmatrix},$$

then

$$(\mathbb{C})_{ij} := (\mathbb{X}^T\mathbb{X})_{ij} = A\left((i-j)\tau\right).$$

This matrix is symmetric and thus diagonalizable, and also at least non-negative definite. Its eigenvalues are called the singular values, and correspond the the magnitude of variance of projections of the embedded vectors into individual directions of the principal components.

If the time delay is too small, then all the elements of matrix $\mathbb{C}$ will have similar value $(\mathbb{C})_{ij} \approx A(0)$, and thus there will be one dominant singular value, while others will remain close to zero. This singular value then corresponds to the main diagonal of the attractor.

If the time delay is too large, then the diagonal elements will approach average of the squared time series $(\mathbb{C})_{ii} \approx \langle x^2 \rangle$, while the remaining elements will converge to zero due to decay of the autocorrelation function, $\mathbb{C} \approx c\mathbb{I}$for some constant c. This corresponds to the reconstruction forming a featureless noise. [6]

One drawback of this method is that its evaluation is largely subjective. Moreover, it was suggested that although this method is effective noise reduction technique, its effectiveness at delay estimation is less clear - the number of large singular values is sensitive to noise. [62]

1.4.4.5 Integral local deformation

The uniqueness theorem of differential equations requires that any no trajectories in the state space intersect. Moreover, in real physical systems, it may be reasonable to assume that it is highly unlikely to

[10] However, additional criteria may arise depending on the particular application.

find closeby trajectories of opposite or orthogonal directions. This property is maintained by a successful embedding, and (if the assumption holds) can occur only in an improper reconstruction.

T. Buzug and G. Pfister presented a quantitative measure of these close trajectory intersections by comparing the the evolutions of reference trajectories with centroids of points on the neighboring trajectories. [16] For the optimal embedding, divergence between these trajectories should be minimized.

First, multiple random reference points are chosen. Let $\mathbf{x}_i(0)$ be such a reference point at time 0. Then, either a fixed number of nearest neighbors or all neighbors within a given radius and their centroid $\mathbf{x}_i^{com}(0)$ are found. Then, the absolute growth of the distance between the centroid of those originally neighboring points and the reference point after qt_{ev} time steps is found as:

$$\Delta(q, m, \tau) = \|\mathbf{x}^{com}(qt_{ev}) - \mathbf{x}_i(qt_{ev})\| - \|\mathbf{x}_i^{com}(0) - \mathbf{x}_i(0)\|.$$

The values $\Delta(q, m, \tau)$ are discretely integrated from $q = 1$ to $q = q_{max}$:

$$\mathcal{D}(m, \tau, i) = \frac{t_{ev}}{2} \sum_{q=1}^{q_{max}} \left(\Delta(q-1, m, \tau) - \Delta(q, m, \tau)\right).$$

This expression, called **integral local deformation**, is then averaged over N_{ref} reference points and normalized:

$$\langle \mathcal{D}(m, \tau, i) \rangle_i = \frac{t_{ev} \sum_{i=1}^{N_{ref}} \sum_{q=1}^{q_{max}} \left(\Delta(q-1, m, \tau) - \Delta(q, m, \tau)\right)}{2N_{ref}\Delta t \left(\max_{i \in 1,2,\ldots,N} x_i - \min_{i \in 1,2,\ldots,N} x_i\right)}$$

1.4.5 Embedding dimension selection

1.4.5.1 False nearest neighbors

Since the dynamics $\mathbf{F}$ are assumed to be a *smooth* vector field and the attractor A is a *compact* set in the phase space, its members acquire near neighbors, which should be subject to similar evolution. Therefore, these neighbors should remain close to each other after a short interval of time (even though chaos may introduce exponential divergence between them). This is a useful fact, which can be used, for example, to predict future evolution of a trajectory, or a computation of Lyapunov exponents. The **false nearest neighbors** algorithm uses them for estimation of embedding dimension. [50]

The main idea is to use the transition from dimension m to dimension $m + 1$ in the embedding procedure to differentiate between "true" and "false" neighbors. If the embedding dimension m is too small, some members of A that are close to each other may not be neighbors in the true state space, simply because the true state space is projected down to a smaller space (see Figure []). These members are *false neighbors*, all other neighbors are *true*. When the attractor is fully unfolded into large enough dimension and is properly embedded, all neighbors are true.

Add the figure from Kennel.

Let use denote by $y^{(r)}(n)$ the r-th nearest neighbor of $y(n)$. Then, let $R_m(n, r)$ denote the Euclidean distance between $y(n)$ and its neighbor:

$$R_m(n, r) = \sqrt{\sum_{k=0}^{m-1} [x_{n+k\tau} - x_{n+k\tau}^{(r)}]^2}$$

Then, any near neigbor for which the distance increase after transition from dimension m to dimension $m + 1$ is large in comparison to the initial distance is marked as false:

$$\left[\frac{R_m^2(n, r) - R_{m+1}^2(n, r)}{R_m^2(n, r)}\right]^{1/2} = \frac{x_{n+k\tau} - x_{n+k\tau}^{(r)}}{R_m(n, r)} > R, \tag{1.8}$$

where $R \in \mathbb{R}$ is some threashold. The m for which the relative proportion of false neigbors to all neigbors reaches zero is the embedding dimension suggested by this criterion.

This criterion, by itself, is not sufficient for determining proper embedding dimension. When applied to limited amount of white noise data, it erroneously suggested embedding the noise into a low dimensional attractor. This happens because even though a state may be a nearest neigbor, it is not necessarily temporally close, and thus the assumptions above do not hold. The experiments performed by Kennel et al. show for such states it is usually $R_m(n, r) \approx R_A$, where R_A is radius of the attractor. Furthermore, for increasing amount of data, the embedding dimension suggested by this criterion also increased - behavior not observed for relatively small dimensional attractors. [50]

Therefore, Kennel et al. propose another criterion in addition to the one above. Since false neighbors which are near, but temporally distant, are usually stretched to the extremeties of the attractor with transition from m to $m + 1$, they suggest marking all near neighbors satisfying

$$\frac{R_{m+1}(n, r)}{R_A} > A \tag{1.9}$$

as false, where R_A may be computed as, for example

$$R_A = \frac{1}{N} \sum_{n=1}^{N} \left[x_n - \overline{x} \right]^2 .$$

Although this technique is commonly used, it is not without its drawbacks. An obvious point is that altough it is true that distance between neigbors in unfolded attractor should not grow with increase in dimension, the inverse is not necessarily true, i.e. stable distance between near neighbors with increase in dimension does not guarantee that these neighbors are true.

The authors suggest some values of the tolerance parameters they found useful in their experiments, but, in general, the results of this technique may depend on the choice of R and A. Their selection is subjective and somewhat arbitrary. The best course of action is to evaluate the technique for multiple values of R and A and select those with the most "reasonable" results.

In practice, it has been found that the results of this method are sensitive not only to the tolerance parameters R and A, but also to the lag as well. [52]

Also, this method tends to underestimate m for very small τ. Small τ forces the attractor to lie near the diagonal in $\mathbb{R}^m$ and further increasing m imposes very little effect on the geometry of the attractor. In effect, most points will appear as true neighbors leading to a wrong conclusion. [52]

Lastly, in presence of measurement noise, the proportion of false neigbors may increase after transition to a higher dimension, since even identical vectors will diverge. [48]

1.4.5.2 Average false neighbors

This technique by Cao [18] addresses one of the drawbacks of false nearest neighbors mentioned in the previous section - the variance of results based on subjective choice of embedding parameters. It does so by defining two parameter free functions dependent only on the embedding parameters.

The first function measures the variation of average ratio of distance of two neighbors in one dimension to the distance of the same neighbors in a higher dimension. More precisely, let

$$E(m) = \frac{1}{N_{(m,\tau)}} \sum_{i=1}^{N_{(m,\tau)}} \frac{\| \mathbf{x}_i^{(m+1)} - \mathbf{x}_{n(i,m)}^{(m+1)} \|_\infty}{\| \mathbf{x}_i^{(m)} - \mathbf{x}_{n(i,m)}^{(m)} \|_\infty},$$

where $n(i,m)$ denotes the nearest neighbor of vector $\mathbf{x}_i$ in dimension m, and $\|\cdot\|_\infty$ denotes the Chebyshev norm [11]. Then, the first statistic is defined as

$$E_1(m) = \frac{E(m+1)}{E(m)}.$$

In principle, $E_1(m)$ saturates and stops increasing after some threshold m for systems with finite embedding dimension.

For systems with infinite embedding dimensions it may be difficult in practice to resolve whether E_1 indeed stopped increasing or is still slowly increasing. Alternatively, it may still saturate because of limited amount of data. For this reason, Cao introduces another statistic, whose purpose is to distinguish stochastic from deterministic sources of data.

Let

$$E^*(m) = \frac{1}{N - m\tau} \sum_{i=1}^{N-m\tau} |x_{i+m\tau} - x_{n(i,m)+m\tau}|.$$

Then, similarly to above, the second statistic is defined as

$$E_2(m) = \frac{E^*(m+1)}{E^*(m)}.$$

Since, for random time series, the future values are independent of the present ones, the ratio $E_2(m)$ is expected to be close to 1 for all m.

1.5 Non-linear measures

In this section, we will study quantities invariant under embedding. These can be further use to characterize the dynamics of deterministic dynamical systems.

1.5.1 Lyapunov exponents

The characteristic property of chaotic systems is their sensitivity to initial conditions - similar causes need not have similar effects. Consequently, even small uncertainty in the current state of the system (due to, at best, with limited storage space) results in virtual impossibility of predicting future state of the sytem more than a short amount of time into the future, since uncertainty in the initial state is expanded at exponential rate with passage of time by the chaotic dynamics for the predicted future states (see Figure).

Lyapunov exponents can be used to quantify this sensitivity. Consider a small sphere of initial conditions $B_r(\mathbf{x})$ for a state $\mathbf{x}$ in the phase space, r infinitesimal, and $\mathbf{x}_n \in B_r(\mathbf{x})$. To study the evolution of states in this ball, we can use a linear approximation of $\mathbf{F}$. Let us assume, for simplicity, that $\mathbf{x}_{n+1} = \mathbf{F}(\mathbf{x}_n)$. Then for infinitesimal divergences $\delta\mathbf{x}_n$, $\delta\mathbf{x}_{n+1}$, we have

$$\delta\mathbf{x}_{n+1} = T^{(n)}\delta\mathbf{x}_n,$$

for a tangent map $T^{(n)}$, where

$$(T^{(n)})_{ik} = \frac{\partial F_i(\mathbf{x}_n)}{\partial x_{n+k}}.$$

Product of these tangent maps for subsequent states along a trajectory can be written as a product of two rotations and a diagonal matrix:

Isn't there a theorem for that?

$$\prod_{n=1}^{N} T^{(n)} = R_d T_{diag} R_b.$$

Then, the Lyapunov exponents can be defined as [34]

$$\lambda_i = \lim_{n\to\infty} \frac{1}{N} \log(T_{diag})_{ii}.$$

In other words, as the system evolves, $B_r(\mathbf{x})$ expands (or contracts) exponentially in m directions defining semiaxes of a sphere, where length of each semiaxis corresponds to the rate of expansion (or contraction) in the corresponding direction. The average lengths of these semiaxis for $\mathbf{x}$ over the entire state space are exactly Lyapunov exponents. Hence, m dimensional system has exactly m Lyapunov exponents, collectively called its *Lyapunov spectrum*.

Computation of the Lyapunov spectrum for analyticial given $\mathbf{F}$ is straightforward using the definition above. But for dynamics given implicitly in a time series is difficult (although some algorithms, e.g. the one introduced by Eckmann in 1986 [22]). It is commonly agreed that estimating Lyapunov exponents is even more difficult than esimating correlation dimension [6], although they have been successfully employed in EEG analysis. [81, 40, 89] It has been claimed by P. Grassberger et al. that any application of these measures to physical systems should be interpreted with caution, mainly because all physical measurements are corrupted by noise, and reliable separation of signal is not always possible. [34] They suggest that when emplying these techniques, the goal should not be to estabilish to strongest form of determinism, but to use them to ask whether determinism can be ruled out at all.

Since the direction of the largest Lyapunov exponent dominates growth, we can say that the average rate of separation between two points in the phase space with similar initial conditions can be characterized by the largest Lyapunov exponent. As a consequence, it is unnecessary to compute the entire Lyapunov spectrum - which would require identifying appropriate Lyapunov directions - if our goal is to find a global property of the system characterizing the degree of average instability and unpredictability. It is sufficient to measure the average rate of separation. [79]

Hence, let us define $\|\mathbf{s}_{n_1} - \mathbf{s}_{n_2}\| = d(0) \ll 1$ as an initial distance between two nearby points in the state space, and $d(i) = \|\mathbf{s}_{n_1+i} - \mathbf{s}_{n_2+i}\|$. Then, the largest Lyapunov exponent λ_1 can be approximated as

$$d(i) = d(0)e^{\lambda_1(i\Delta t)}, \quad d(i) \ll 1, \quad i \to \infty, \quad d(0) \to 0, \tag{1.10}$$

where Δt is sampling time of the time series.

The Lyapunov exponents carry the units of an inverse time - $1/\lambda_1$ gives a typical time scale for the divergence or convergence of nearby trajectories. [48] Equivalently, $1/\lambda_1$ is (on average) an upper bound on predictability in the system. [6] Also equivalently, they also can be seen as quantification of the degree of chaos in the system; a sigle positive exponents is a sufficient indication of presence of chaos. [79]

Say what different values of λ_1 say about the system.

1.5.1.1 Rosenstein's algorithm

In the following, we will describe *Rosenstein's algorithm* for computation of the largest Lyapunov exponent. [79] This algorithm was found to be relatively robust to noise, values of the embedding parameters and limited amount of data.

[11] This norm suggested by the author, but presumably, another norm can be used.

First, state space is reconstructed using time delay embedding (see Section 1.4.1). The suggested method of time delay selection is the autocorrelation method (see Section 1.4.4.1).

For given embedding dimension m and each point on the trajectory $\mathbf{x}_j$, the algorithm locates the nearest neighbor $\mathbf{x}_{n(j,m)}$, such that their distance in the embedded space is minimized:

$$d_j(0) = \|\mathbf{x}_j - \mathbf{x}_{n(j,m)}\|.$$

As an approximation, we want to assume $\mathbf{x}_j$ and $\mathbf{x}_{n(j,m)}$ to be nearby initial conditions, but at the same time, we know they lie on the same trajectory. Hence, we will impose a condition on their temporal separation:

$$\frac{1}{4}\text{ time series length} > |j - n(j,m)| > \text{mean period of the time series.}$$

Then, assuming the j-th pair of nearest neighbors diverge exponentially at a rate given by the largest Lyapunov exponent, we have

$$d_j(i) \approx d_j(0)e^{\lambda_1(i\Delta t)}.$$

By taking logarithm of both sides, we obtain

$$\ln d_j(i) \approx \ln d_j(0) + \lambda_1(i\Delta t).$$

This represents a set of lines, one for each point on the reconstructed trajectory, each with a slope roughly proportional to λ_1. So, the algorithm approximates the largest Lyapunov exponent by least squares fit to the average line

$$d(i) = \frac{1}{\Delta t}\langle \ln d_j(i)\rangle_{j=1,2,\ldots,N_{(m,\tau)}}.$$

Note that the sampling period Δt plays no role - one can decide to set $\Delta t = 1$ and work with units of time series indeces instead of seconds interchangeably. Relatedly, we can even rescale or shift the data, since Lyapunov exponents are invariant under any smooth invertible map.

Another prominent and widely used algorithm for estimation of the largest Lyapunov exponent is Wolf's algorithm [100], but due to its instability and the impossibility of distinguishing exponential divergence, it cannot be recommended. [48]

As we have mentioned already, the projection involved in the measurement may make distances shrink apparently for short times, although they grow in the true state space. [48] Moreover, in the true state space distances do not grow everywhere on the attractor with the same rate, and locally they may even shrink. LLE is average of those local divergence rates. Influence of noise can be minimised by using an appropriate averaging statistics.

1.5.1.2 Dataset size requirements

The minimum dataset requirements was estimated by Eckmann and Ruelle in [24] by imposing requirements on the distances and number of neighbors for each point. If $\Gamma(r) \gg 1$ is the average number of neighbors withing radius r, we may approximate it as

$$\Gamma(r) \approx \text{const.} \times r^m,$$

and we also know that $\Gamma(d) \approx N$, where d is the diameter of the attractor. Therefore, we obtain

$$\Gamma(r) \approx N\left(\frac{r}{d}\right)^m \gg 1 \implies N > \left(\frac{d}{r}\right)^m.$$

For example, if we require the ratio of the average distance to the nearest neighbor to the extent of the attractor to be $r/d \leq 0.1$, we have $N > 10^m$ as the minimum time series length requirement.

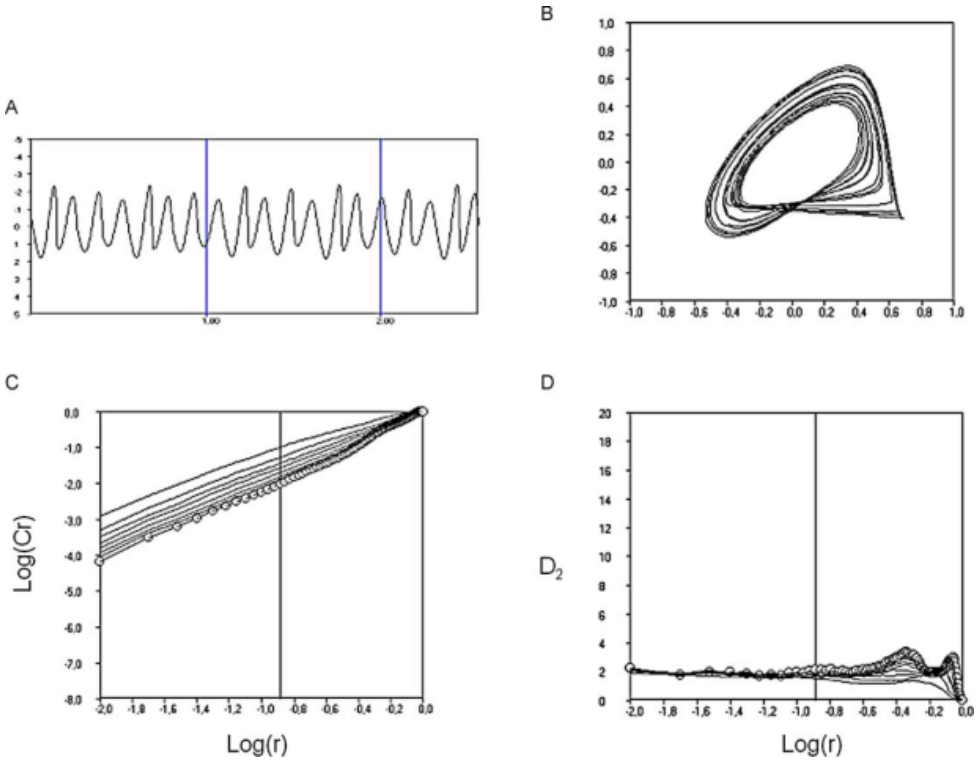


Figure 1.5: Computation of the correlation dimension [89]. TODO: Add description.

1.5.2 Correlation dimension

The world of mathematics offers numerous definitions of dimension (box-counting dimension (1.4), Hausdorff dimension, information dimension, etc.) and similar quantities, but many of them can be regarded as variations of the following, simple and intuitive analogy: [92]

$$\text{bulk} \approx \text{size}^{\text{dimension}} \implies \text{dimension} = \lim_{\text{size}\to 0} \frac{\log \text{bulk}}{\log \text{size}}. \tag{1.11}$$

In other words, dimension can be loosely defined as scaling of "bulk" (corresponds to mathematical concept of measure) as a function of its linear "size". Of course, dimensions of different definitions may not be equal to each other, but for our purposes, we are interested in the most computationally accessible.

Unlike Lyapunov exponents, which measure dynamical properties of the system, (correlation) dimension is a purely geometrical property of the attractor, independent of the ordering of the reconstructed vectors. Is this true?

In this thesis, we are interested in dimension estimation for the following reasons:

1. Even a system with high number of degrees of freedom, such as a brain, may actually evolve in a much lower-dimensional subspace. The number of active degrees of freedom may provide a measure of complexity of the observed system. This information is available in the attractor of the system and it can be shown that this property is preserved by state space reconstruction. [6]

2. It can help distinguish stochastic and deterministic processes, since stochastic processes, after sufficient passage of time, use all available state space dimensions.

Of course, although these expectations can be justified theoretically, the numerical reality may be different.

Most definitions of dimension are based on first covering the studied object in the state space with the smallest possible balls (using a given metric). Correlation dimension is a special case of generalized box-counting dimension (which is a generalization of box-counting dimension already introduced in Definition 8), defined as

$$d_\kappa(A) = \lim_{r\to 0} \frac{1}{\kappa} \frac{\log \int_M (\mu(B_r(\mathbf{x})))^\kappa \, \mathrm{d}\mu(\mathbf{x})}{\log r},$$

where the integration is over the whole state space M and μ is measure concentrated on A. If we define μ as

$$\mu(\mathbf{x}) \coloneqq \int_M \Phi(r - \|\mathbf{x} - \mathbf{y}\|) \, \mathrm{d}\mu(\mathbf{y}) \tag{1.12}$$

The we can write the, "bulk" of A, so called generalized correlation integral as

$$C(\kappa, r) = \left(\int_M (\mu(B_r(\mathbf{x})))^\kappa \right)^{\frac{1}{\kappa}} = \left[\int_M \left(\int_M \Phi(r - \|\mathbf{x} - \mathbf{y}\|) \, \mathrm{d}\mu(\mathbf{y}) \right)^\kappa \mathrm{d}\mu(\mathbf{x}) \right]^{\frac{1}{\kappa}}$$

It can indeed be shown that $C(\kappa, r) \propto r_\kappa^d$.

In the continuous case, correlation dimension than takes to form

$$d_2(A) = \lim_{r\to 0} \frac{\log C(r, 2)}{\log r}.$$

As explained in Section 1.3.1, correlation dimension is closely related to the distribution of lengths of diagonal lines on recurrence plots. Intuitively, we can see that both methods are measuring temporal correlations in the original time series.

1.5.2.1 Grassberger-Procaccia algorithm

There are essentially three ways of computing correlation dimension: box-counting algorithms, pairwise distance algorithms, and nearest neighbors algorithms. Grassberger-Procaccia algorithm, which we use to compute correlation dimension, is a variant of a pairwise distance algorithm.

This class of algorithms, used in discrete cases with limited amount of data, estimates the measure of a box centered on point $\mathbf{x}_i$ in the reconstructed space as

$$\mu_i = \frac{1}{N_{(m,\tau)}}$$

and zero everywhere else.

Thus, in the discrete case, the correlation sum $C(r)$ can be computed as

$$C(r) \coloneqq C(r, 2) = \frac{2}{N_{(m,\tau)}(N_{(m,\tau)} - 1)} \sum_{i<j} \Phi(r - \|\mathbf{x}_i - \mathbf{x}_j\|). \tag{1.13}$$

which corresponds to the fraction of points in the phase space whose distance is smaller than r. Under certain reasonable conditions, correlation sum is an unbiased estimator of the correlation integral. [33]

Typical behavior of the correlation sum is shown in Figure 1.6. We can see that the curves are forced to meet at the same point for all m - for high enough r, all points are counted and $C(r) = 1$ (or $C(r) = \binom{N_{(m,\tau)}}{2}$ not normalized). As the lines shift to the right with increasing m and stay parallel in the

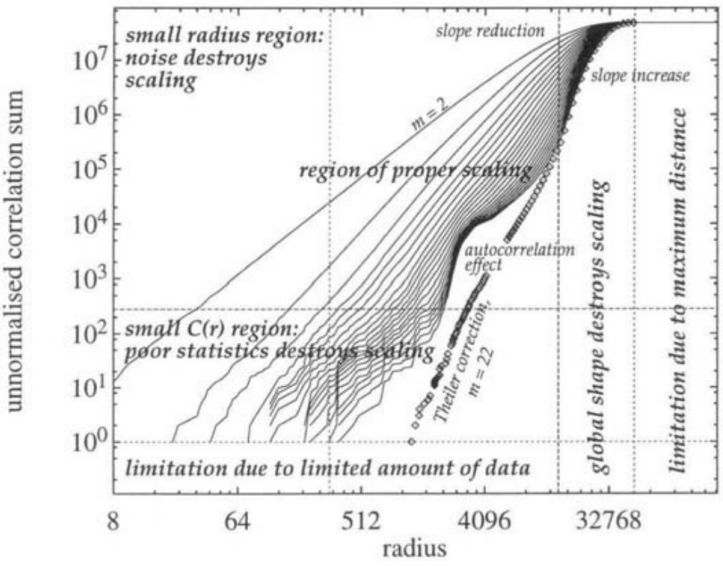


Figure 1.6: Log-log plot of typical behavior of $C(r)$.

proper scaling region, the slope near that point necessarily increases with m. For high enough m, the scaling region disappears. Moreover, the values of $C(r)$ are inaccurate for small r due to noise and for small $C(r)$ due to statistical fluctuations (corresponding to horizontal lines). Thus, there is only a limited interval of r and limited set of embedding dimensions m for which an accurate estimation of d_2 can be made. [6]

In our experiments, we used *local slopes approach* to estimating the correlation dimension, which is based on the idea of assigning a dimension estimate to each value of r by defining

$$d_2(r) = \frac{\partial \log C(r)}{\partial \log r}. \tag{1.14}$$

In our implementation, we perform a least squares fit of values $(\log r, \log C(r))$ for a window of 6 neigboring points for each sampled r. Expected behavior of the resulting function in a favorable case can be seen in Figure 1.7.

1.5.2.2 Dataset size requirements

There are multiple estimations of the minimum dataset size. Most of them are based on an attempt to avoid so called *edge effect*. It can be shown that the correlation dimension for a hypercube in m-dimensions of unit edge length the local correlation dimension is

$$d_2^{(m)}(r) = m - \frac{mr}{2-r} \approx m(1 - \frac{r}{2}).$$

For large enough r, $d_2^{(m)}(r)$ converges to zero. This result, which can be generlized to any finite object, is a consequence of the discontinuity of the measure (1.12) at the boundaries of the hypercube. Theiler, assuming evalution of the local correlation dimension for radius where each point has on average one neighbor (such that $C(r) = 1/N_{(m,\tau)}$), derived an estimate for the minimum data set size as

$$N_{(m,\tau)} = \frac{1}{(4\rho)^m},$$

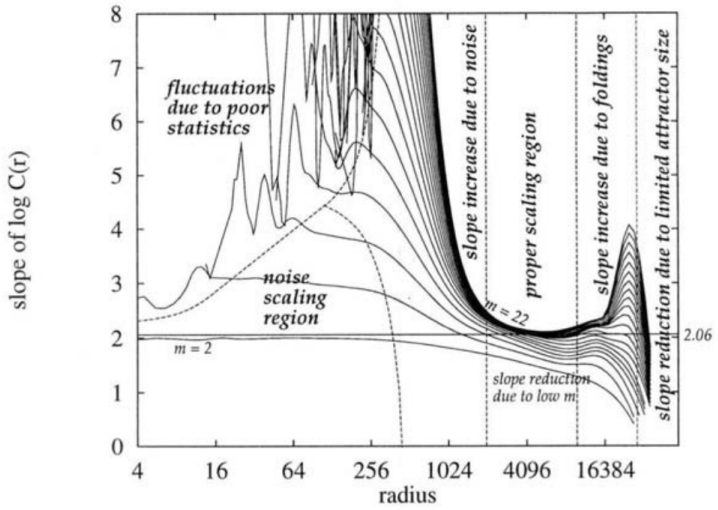


Figure 1.7: Log plot of a typical local dimension estimates in favorable case.

where ρ is the maximum error. This implies an exponential increase of minimum required dataset size with embedded dimension. For example, $N_{(m,\tau)} = 5^m$ for $\rho = 5\%$. [6]

1.5.3 Detrended fluctuation analysis

Physiological time series, such as EEG, may exhibit so called statistical self-affine properties. Self-affinity is a special case of self-similarity, which occurs when one or more small parts of a fractal object is exactly or approximately similar to itself. When self-similarity is expressed in terms of statistical properties (e.g. mean value and variance of a part of time series are scaled version of its overall mean and variance), then the object is called statistically self-similar.[12] Self-similarity, in turn, differs from self-affinity in that self-affine objects witness similarity anisotropically, i.e. after applying an anisotropic affine transformation. [13] [36] Stated more formally:

Definition 11 ([11])**.** *A time series X given by $x_1, x_2, \ldots, x_n$ is said to be* ***statistically self-affine*** *if*

$$\mathrm{std}(X, Lt) \approx L^H \mathrm{std}(X, t),$$

where $\mathrm{std}(X, k)$ *is the standard deviation of the process X calculated over windows of length k, H is the Hurst parameter, and L a window length factor.*

The Hurst parameter, which behaves similarly to the Hurst exponent (see Section 1.5.4), ranges between 0 and 1. Higher values of H describe smoother signals, with high values followed by low, whereas low values of H indicate radical oscillations between high and low values. Note that since a stationary process has constant variance across time scales, Definition 11 applies only to non-stationary processes. However, even stationary processes may exhibit scale-free behavior. These are modelled as so called fractional Gaussian noise (fGn), whereas non-stationary processes are modelled as fractional Brownian motion (fBm). These processes are related by the fact that increments in fBm can be modelled

[12] An example of statistically self-similar object are are naval coastlines.

[13] The measured property of a self-similar process (e.g. the size of a flower on Romanesco cauliflower) do not follow normal distributions, but power law distributions. Hence, mode and mean of provide a poor representation of this representation. These processes do not have a scale at which to measure these statistics to characterize them, and are therefore called scale-free.

as a fGn process with same H. This relationship allows us to generalize Definition 11 for non-stationary processes. [36]

DFA is a method of estimating H without making prior assumptions about stationarity of the process by exploiting the relationship between fGn and fBm procesess. First, a so called signal profile, i.e. integral of the de-meaned signal is computed as

$$y_k = \sum_{i=1}^{k} (x_i - \langle x \rangle) .$$

The resulting time series y is then divided into segments of varying length m (each value of m representing a time scale). A local linear least-squares fit is applied to each of these segments. Let us designate the resulting piecewise linear fit as $y_m(k)$. The integrated time series is then detrended by subtracting the local linear fit. The root mean square error is then given by

$$F(m) := \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - y_m(i))^2}.$$

Finally, if a log-log plot $F(m)$ as a function m shows a linear scaling region (i.e. the original time series exhibits self-similar, scale-free properties described above), the slope of this line approximates H and represents the result of DFA analysis. [56]

Maybe add some more intuitive explanation.

The importance of DFA for EEG analysis comes from the observation that it can reveal so called long-range temporal correlations (LRTC) in neuronal activity. Long-range temporal correlations, or long-range dependence, is a phenomenon which occurs when the average rate of decay of statistical dependence between increasingly (temporally) distant points in the time series is slower than exponential. Large-scale patterns in EEG activity may be characteristic of baseline processing during eyes-closed wakeful condition in healthy human brain. These parameters computed from the theta amplitudes were shown to be negatively correlated with (Hamilton) depression score, thus suggesting that depressed patients display abnormally small autocorrelations on large scale. [57] In our study, we observed similar results, see Section 2.6.2.

1.5.4 Hurst exponent

As mentioned in the previous section, similarly to the Hurst parameter in DFA, Hurst exponent is a measure of presence of long-range temporal dependencies in the time series. It was developed from Edwin H. Hurst observation when researching the optimal (or minimal required) storage sizing of river dams. Supposing there is a constant reservoir outflow equal to the mean annual water discharge, required storage size corresponds to the range (i.e. the difference between the maximum and the minimum value) of a cumulative sums of deviations from the mean annual discharge. We shall call this value, as a function of the number of years, $R(n)$. [42] After manually analyzing about a hundred records of natural phenomena, Hurst was able to demonstrate this value, on average and after normalizing by the standard deviation of the original time series, follows the following trend:

$$R(n)/\sigma(n) \propto (n/2)^K. \tag{1.15}$$

In this equation, $R(n)/\sigma(n)$ is called the rescaled range, and K is called the Hurst exponent. [43] Obviously, it is always the case that $0 \leq K \leq 1$.

The algorithm we used for computing estimation of the Hurst exponent is as follows. The time series is split into multiple subseries of varying length n, and cumulative mean adjusted is computed for each. Range $R(n)$ is computed from this cumulative time series, and $\sigma(n)$ from the origingal time series. In other words, let us have time series x, with values $x_1, x_2, \ldots, x_N$. For each subseries $x^{(n)}$ of length n, we compute

$$z_k^{(n)} = \sum_{i=1}^{n} (x_i - \langle x^{(n)} \rangle) \qquad \text{for } k = 1, \ldots, n$$
$$R(n) = \max_{i=1,\ldots,n} z_i - \min_{i=1,\ldots,n} z_i$$

and

$$\sigma(n) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \langle x^{(j)} \rangle)}.$$

Finally, the Hurst exponent K is computed by fitting the plot of the logarithm of scaled range $\log R(n)/\sigma(n)$ versus $\log n$.

Interestingly, if the quantites observed resulted from mutually completely independent events (i.e. white noise, with its corresponding cumulative sum, random walk), the relationship in equation (1.15) is replaced with

$$R(n)/\sigma(n) \propto 1.25\sqrt{N},$$

as can be easily verified by flipping a set of coins. [14] [43] This allows us to recognize stochastic processes with mutually uncorrelated values. The value of Hurst exponent for white noise is $K = 1/2$, and many natural processes, such as rainfalls, river water level heights, temperatures and pressures, annual growth of tree rings, and even financial markets have $K > 1/2$, suggesting long-term temporal correlations in the processes. Values of $0 \leq K < 1/2$, on the other hand, suggest long-time negative correlations, i.e. high values being often followed by low values in the future. [42]

1.5.5 Higuchi fractal dimension

In this section, it will be beneficial to change our usual notation for the purpose of readability. Let us have time series $x(1), x(2), \ldots, x(N)$. We construct a new time series, x_k^m, as

$$x(m), x(m+k), x(m+2k), \ldots, x(m + \lfloor \frac{N-m}{k} \cdot k \rfloor), \qquad m = 1, 2, \ldots, k, \qquad m, k \in \mathbb{N},$$

where m is the initial time, and k is the interval time. In this way, we sample k "subseries" where the interval (or size, see equation (1.11)) is precisely k. Then, we define "length of the curve" x_k^m (or bulk, see equation (1.11)) as follows:

$$L_m^k = \left(\sum_{i=1}^{\lfloor \frac{N-m}{k} \rfloor} |x(m+ik) - x(m+(i-1)k)| \right) \frac{N-1}{\lfloor \frac{N-m}{k} \rfloor \cdot k} \cdot \frac{1}{k},$$

where $\frac{N-1}{\lfloor \frac{N-m}{k} \rfloor \cdot k}$ is a normalization factor. Length of the original curve as a function of k is then defined as the average $\langle L(k) \rangle$ over k values of $L_m(k)$. If $\langle L(k) \rangle \propto k^{-D_H}$ for some value of D_H, then the curve

[14] Hurst himself actually made experiments, tossing 10 coins 1000 times. It took him almost 6 hours. [26]

can be considered fractal with fractal dimension D_H, which can be estimated by least-squares fitting the logarithm of the length $\log \langle L(k) \rangle$ as a function of $\log k$. [39]

In summary, Higuchi fractal dimension is can be computed, in analogy with correlation dimension and equation (1.14), as

$$D_H = \frac{\partial \log \langle L(k) \rangle}{\partial \log k}.$$

Moreover, comparing with equation (1.11), can see that bulk = $\langle L(k) \rangle$ and size = k.

Benefits: fast computation.

Add some successful application to EEG.

1.5.6 Sample entropy

Understood in the context of dynamical systems, entropy is the rate at which a given system produces information. It is equal to the sum of all positive Lyapunov exponents of the system's attractor, and positive entropy indicates presence of chaotic dynamics. [6] Computing entropy from a physiological time series directly using the information-theoretical definition, is, however, problematic. The time series produced during measurements on biological systems are often short and noisy. Moreover, in EEG analysis, the impact of noise is especially severe. To combat this issue, many methods of computing entropy for such time series has been devised. Sample entropy represents an improvement on other entropy measure popular in clinical settings, called approximate entropy, which has been successfully applied on EEG to classify diseases such as schizophrenia, epilepsy, and addiction. [102]

For a given embedding dimension m and tolerance parameter r, sample entropy can be defined as the negative natural logarithm of the conditional probability that two subseqences of the time series of length m which are similar, i.e. their distance is less than r (excluding self-matches[15]), will remain similar after including the next point, i.e. when their respective lengths are increased to $m + 1$. Thus, sample entropy is thus a measure of predictability (the opposite of regularity), lower value of sample entropy indicates more self-similarity, or, in a certain sense, less complexity. In other words:

Definition 12 ([75])**.** *Let us have a time series $x_1, x_2, \ldots, x_N$, and let $X_m(i) = (x_i, x_{i+1}, \ldots, x_{i+m-1})$ be the i-th vector in the embedding of dimension m, and $\|\cdot\|_\infty$ be the Chebyshev metric[16]. Then,* ***sample entropy*** *is defined as*

$$\mathrm{SampEn}(m, r, N) = -\ln \frac{A^m(r)}{B^m(r)},$$

where

$A^m(r)$ *is the number of vector pairs in the embedding satisfying $\|X_{m+1}(i) - X_{m+1}(j)\|_\infty < r$, $i \neq j$, and*

$B^m(r)$ *is the number of vector pairs in the embedding satisfying $\|X_m(i) - X_m(j)\|_\infty < r$, $i \neq j$.*

Obviously, it is always $A^m(r) \leq B^m(r)$, hence sample entropy is always non-negative. If $A^m(r) = B^m(r) = 0$, no regularity has been detected, and $B^m(r) \neq 0$ with $A^m(r) = 0$ corresponds to (the above mentioned) conditional probability of 0, and infinite value of sample entropy. The recommended value of r is $0.2 * \mathrm{std}(x)$. [75]

Sample entropy has been successfully employed for diagnosing depression [3, 37]. It was shown to be significantly different between middle aged and elderly women during sleep. [14]

Extend this section a little bit.

[15] This is one of the differences of sample entropy from approximate entropy.

[16] Any metric can be used, but Chebyshev metric is recommended by the original authors. [75]

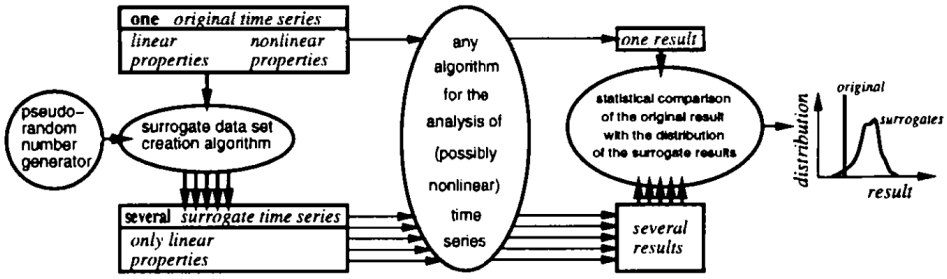


Figure 1.8: [6]

1.6 Surrogate data testing

It has been shown that, for example, fitered noise can mimic low-dimensional chaotic attractors when examined by Grassberger-Procaccia algorithm described above. [74] Hence, interpretation of results obtained by non-linear analysis require judgement. For example. obtaining finite-dimensional estimates for d_2 is not evidence of non-linearity, but may indicate lack of data, measurement error, on numerical inaccuracies. The result of these algorithms does not include any error estimate, and sometimes even non-linearity of the underlying process is uncertain. So, one may ask: would we obtain the same estimate for data with the same (non-measured) linear properties as the original, but differs in the (measured) non-linear property? In the following, we will describe a method for answering this question.

We may also want to know whether the process is determinsitic or not. There are tests for that, but we are not using them in this thesis.

To this end, we construct a Monte Carlo hypothesis test of non-linearity. We choose a null hypothesis of a model for the process creating obtained data which denies the property we assume to measure. For each time series, we create so called *surrogate data* which deliberately capture only properties consistent with chosen null hypothesis, and compute the estimates using the same method as for the original data. If the result for the original time series is significantly different from the surrogate estimates, we reject the the null hypothesis. In the opposite case, we fail reject the null hypothesis. A schematic depiction of the process can be seen in Figure 1.8.

Here, we use two sided test, and measure of significance is defined as

$$\mathcal{S} \equiv \frac{|Q_{\text{orig}} - \mu_{\text{surr}}|}{\sigma_{\text{surr}}}, \tag{1.16}$$

where Q_{orig} is the statistic computed for the original time series, and μ_{surr}, σ_{surr} are the mean and variance of the statistic computed for the surrogate time series. [94] If we assume that distribution of the generated is Gaussian, than $\mathcal{S} \geq 2$ is required for 95 % significance level. However, validity of this assumption is not always guaranteed. For non-Gaussian distributions, we may require larger $\mathcal{S}$, or, alternatively, use a rank based test, as follows. [48]

Using rank-based test, we want to test if Q_{orig} is smaller or larger than the expected value of estimates produced by the null hypothesis model. If we generate n_s surrogate estimates, then, we have n_s estimates following the null hypothesis, each having a probability $2/n_s$ of being the smallest or largest. A false rejection will happen if Q_{orig} happens to also follow the null hypothesis and is either the smallest or the largest, which happens with probablity $1 - \alpha := 2/(n_s + 1)$, where α is the confidence level. Hence, for confidence level $\alpha = 95\%$, the number of surrogates should be $n_s = 38$. [6] Note that, since many of the algorithms used for estimating non-linear measures are relatively computationally expensive, surrogate analysis with high confidence levels, especially on large datasets of multichannel signals such as EEG, is even more so.

1.6.0.1 Generating improved amplitude adjusted surrogates

For our purposes, since we assume that the data are produced by a non-linear process, a reasonable null hypothesis may be that the data are produced by a Gaussian linear stochastic process AR(p)

$$x_{t+1} = \mu + \sum_{j=0}^{p-1} a_j x_{t-j} + \sigma e_t, \tag{1.17}$$

with unknown parameters $a_j, e_t, \mu, \sigma \in \mathbb{R}$. [94]

If the computed non-linear statistic depends on the free parameters in AR(p) (1.17) (which is not true, e.g. for d_2), then one may try to estimate these parameters from the original time series. Alternatively (and this is the approach we use in our analysis), one may exploit the fact that AR(p) can be also perfectly described by its power spectrum. [94][17] Hence, to obtain a surrogate, one may simply perform a Fourier transform of the original time series, randomize phases, and apply inverse Fourier transform. This way, the amplitudes (composing the power spectrum) are preserved. This procedure has been named *Fourier transform phase randomization* (FTPR).

However, there is a drawback of FTPR. It has been shown that if the amplitudes of AR(p) are not Gaussian (as in (1.17)), e.g. non-linear,then the surrogates created using this method show non-linear behavior. [48] Rarely do the amplitudes of an experimental process follow a Gaussian distribution. Hence, we change our model to correspond a non-linear, time independent filter applied to the output of AR(p). Surrogate creation algorithm for this model was described by Theiler in [94]: rescale the values of the original time series so that they are Gaussian, apply FTPR described above, rescale the values back to follow the same distribution of the original time series. This surrogate creation method is called *amplitude-adjusted Fourier transform* (AAFT), and has been successfully applied to EEG signal. [93]

e.g. or i.e. here?

Even this method is not without its drawbacks: due to the final reordering, the original power spectrum is slightly distorted in the surrogate. In [93], it was proposed how to mitigate this effect. The amplitudes of Fourier transform of AAFT surrogates are replaced by the amplitudes of the original time series. The power spectrum is now correct, but the distribution is wrong. So, the original time series is reordered to according to ranks of values in this surrogate. This results in precisely the desired distribution of values, but again, slightly deviant power spectrum. These steps are then iterated and, experimentally, they results seem to converge. Hence, the final procedure, called *improved (iterated) amplitude-adjusted Fourier transform* (iAAFT) can be summarized as follows: [6]

Maybe talk about the problems, e.g. endpoint mismatch? We will need to refer to them later.

1. Compute and store the moduli of the original time series.
2. Create an AAFT surrogate as follows:

   Create a set of random numbers with Gaussian distribution.

   Rank order the original time series, and reorder the random numbers created in the previous step such that they achieve the same ordering as the original time series.

   Randomize the phases Fourier transform of the time series obtained in previous step and apply inverse Fourier transform.

   Find the rank ordering of the time series obtained in the previous step, and reorder the original time series so that it assumes the same rank ordering.

[17] This is due to Wiener-Khinchin theorem, which states, roughly, that spectral decomposition of autocorrelation of a stationary process is the power spectrum of the process.

3. Replace the moduli of these surrogates by those of the original time series and apply inverse Fourier transform.

4. Find the rank ordering of the time series obtained in the previous step, and reorder the original time series so that is assumes the same rank ordering.

5. Apply step 3. to time series obtained in the previous step, or stop if stopping criterion is reached.

1.7 Applications in disease diagnosis

This section is probably not sufficiently exhaustive.

Although non-linear dynamical analysis of EEG signal has been successfully applied to many psychological and psychiatric conditions, such as insomnia, schizophrenia, epilepsy, dementia, Alzheimer's disease, the number of studies applying methods of non-linear time series analysis for clinical depression diagnosis is relatively limited. [76]

It has been found that the EEG dynamics of depressed patients exhibit more predictability than those of non-depressed ones, with this indicator receding after treatment. [64] [69]

Another study analyzed sleep EEGs of depressed and control subjects, and found significantly decreased values of Lyapunov exponents in a sleep stage IV in depressed relative to control. [81]

In 2012, Ahmadlou et al. decomposed 5 EEG channels recorded from frontal lobes of healthy and depressed patients using wavelet filter banks, measured their complexity using Higuchi's fractal dimension, subsequently used ANOVA to discover the most meaningful differences between the groups, and trained a probabilistic neural network classifier, achieving 91.3% classification accuracy on limited amount of data. This research suggested potential of frontal lobe signal assymetry as a measure for depression. [4]

In the same year, Hosseinifard et al. extracted Higuchi's correlation dimension, Lyapunov exponents and Higuchi's fractal dimension from 4 EEG channels of 90 patients split evenly between depressed and non-depressed subjects, achieving 90% accuracy using a logistic regression classifier. [40]

In 2013, Bachmann et al. compared two non-linear analysis methods, spectral assymetry index (SASI) and Higuchi's fractal dimension (HFD), for depression diagnosis, on 34 subjects split evenly between depressed and control group. SASI achieved true detection rate in 88% in depressives and 82% in the controls, while HFD provided true detection rate of 94% in the depressives and 76% in the controls. [9]

Sleep disorder diagnosis may also relevant to this work for the very close connection of depression with disturbed sleep and insomnia [66]. The first study emplying techniques of non-linear analysis on human EEG was published in 1985 and dealt with sleep recordings. [8] This early success sparked intensive research focus on applying non-linear analysis to sleep data, thus generating relatively large amount of results.

Many studies focused of extracting Lyapunov exponents of EEGs measured during various sleep stages. The general pattern that emerged was that deep sleep stages exhibit lower complexity evidenced by lower dimensionality lower values of the largest Lyapunov exponent [89].

Recurrence plots, and RQA in particular, have been demonstrated to be effective at decoding neuroscientific physiological time series. For example, they have been suggested as a method of lowering singal-to-noise ratio in analysis of event related potentials in response to a surprising stimulus, where repeated exposure would influence the outcome (and thus classical averaging methods are not viable). [60] Moreover, they have been successfully employed in detecting epileptic seizures using intracranial recordings. [70] Simple K-nearest neighbors classifiers achieved surprisingly high accuracies at emotion recognition tasks [10], and convolutional neural networks used recurrence plots for activity recognition.

[30] Most importantly for our study, recurrence plots of signals in the left hemisphere were observed to qualitatively differ between healthy baseline and depressed patients. The authors suggested that this area is worth further exploration. [2]

Chapter 2

Non-linear analysis approach

2.1 Dataset

The EEG recordings were performed by and obtained from the Czech National Institute of Mental Health. The dataset comprises total of 133 subjects, 104 women and 29 men, ranging in age from 30 to 65 (47.7 ± 9.58). Handedness was not recorded. Montgomery-Asberg Depession Rating Scale (MADRS) [99] questionnaire assessed by a trained psychologist was used to measure depression severity. This psychometric measurement results in a depression score ranging from 0 (normal) to 40 (severe depression), usually with the following cutoff points: [38]

0 - 6 : symptom absent,

7 - 19 : mild depression,

20 - 34 : moderate depression,

34 - 40 : severe depression.

The experiment lasted 4 weeks. At the beginning of week 1, each subject's depression score was measured, their EEG signal was recorded, and, based on the measurement and patient's history, prescription of up to 4 treatments (drugs or rTMS) was made. After 4 weeks, depression score was remeasured and EEG signal recorded again.

During the EEG recording, 19 electrodes were placed on the scalp in accordance with the Internation 10-20 system (FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz, Pz), see Figure 2.1 for reference. EEG signals of 99 subjects were recorded at sampling frequency f_s of 250 Hz, while 1000 Hz was used for the remaining 34 patients. The patients were not told to close their eyes for the duration of the recording, resulting in unwanted artifacts in the signal. Some of the artifacts were removed manually by the researchers by omitting those parts from the recording, and concatenating the remaining parts. Durations of the resulting measurements range from 23.5 s to 170 s (75.6 ± 20 s) for f_s = 250 Hz , and from 48.8 s to 140.4 s (79.5 ± 18.4 s) for f_s = 1000 Hz. A typical recoding can be seen on Figure 2.2.

We should recognize limitations of this dataset:

- That the patients were not randomly selected - all the patients entered the study because they were experiencing problems negatively impacting their lives. Thus, as a study of depression biomarkers, the experiment lacks truly symptom absent group. However, the patients did differ significantly in severity of the disease.

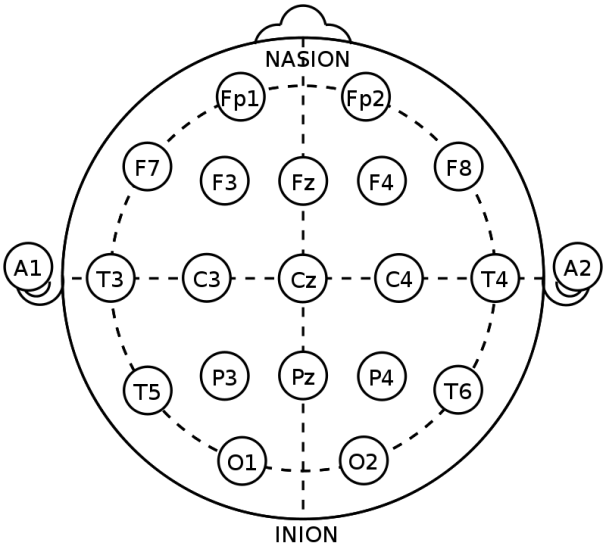


Figure 2.1

- For a study of brain regions associated with depression, this study lacks data on patients' handedness, which may be relevant for distribution of activity in the hemispheres.
- For a study of remission, this dataset lacks a control group given no drugs.
- For a study of treatment effects, patients were assigned different cominations of drugs, making an attempt of finding the singular cause of any observed effects impossible.

2.2 Preprocessing

Recordings of $f_s = 1000$ Hz were downsampled (decimated) by factor 4 to 250 Hz using the Fourier method (also known as trigonometric interpolation), i.e. by performing discrete Fourier transform on the original series, dividing it into $2 * 1000/250 = 8$ intervals, removing all but the first and the last intervals (thus removing the highest positive and negative frequencies, corresponding to low-pass filtering), and performing inverse discrete Fourier transform. This procedure assumes that the signal is periodic, and may have some influence on the obtained results. However, it was observed that this effect is almost negligible, even for considerably higher decimation factors. [20]

In further analysis, unless otherwise specified, recordings were shortened to a fixed length. To balance the data requirements (see Sections 1.5.1.2 and 1.5.2.2), decrease in dataset size due to removal of too short recordings, and stationarity (see Section 1.3.2), the threshold was selected to be 60 s (15 000 datapoints per time series), resulting in exclusion of 26 recordings from the total of 266.

In some studies, band-pass filtering was used to remove frequencies which are physiologically impossible to produce by neural oscillations (e.g. high-pass filtering with 0.5 Hz threshold or lowpass filtering with 70 Hz threshold). [40] Sometimes, it is suggested to notch filter at power line frequencies (40 Hz or 50 Hz). However, some authors suggest that linear filtering may adversely affect the results of non-linear analysis. [6] Others, on the other hand, observed that simple linear filtering does not influence the reconstruction of embedding space considerably. [77] If quality of the data is sufficient, filtering is not necessary. [45] By visual inspection, we found our data to be of sufficient quality and therefore decided to not risk influencing the results by filtering.

In fact, we tried both, but for filtered the results looked slightly more uniform.

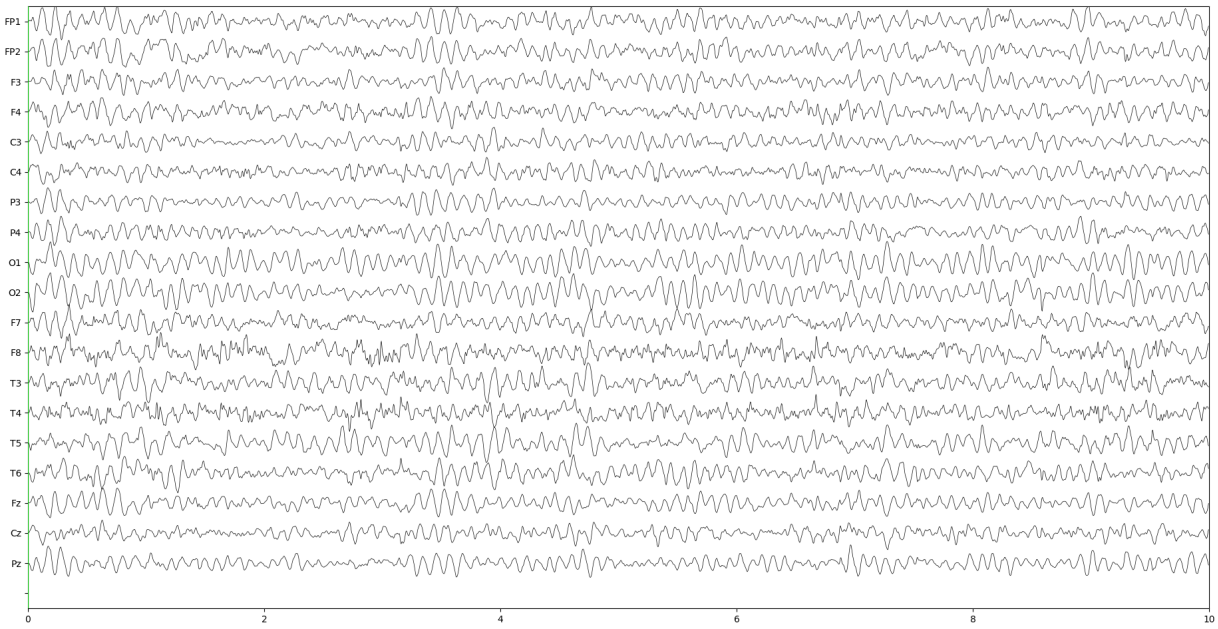


Figure 2.2

2.3 Stationarity

Stationarity was evaluated on multiple time scales using the stationarity test described in Section 1.3.2. However, we found that selecting the least stationary window using this test was did not improve the results as measured by the surrogate data analysis. Moreover, it may be beneficial for future analysis to have each measure computed for the same time window in each channel. Thus, we decided to skip this window selection step and pick a fixed time window for all channels and all recordings.

2.4 State space reconstruction

2.4.1 Time delay

In order to estimate the time delay, we used the following techniques:

1. Reconstruction plots
2. Autocorrelation $A(\tau)$ (see Section 1.4.4.1)
3. Delayed mutual information $\mathcal{I}(\tau)$ (see Section 1.4.4.2)
4. Average displacement from diagonal (ADFD) (see Section 1.4.4.3)
5. PCA reconstructions comparison (see Section 1.4.4.4)
6. Integral local deformation (ILD) (see Section 1.4.4.5)

In this section, we will analyze the results of these techniques for time series obtained from FP1 electrode of patient 75, second session, shown in Figure 2.3. The time series was clipped to 60 s (15000 data points). In the following sections , we will explain how these techniques were used to obtain estimates of individual non-linear measures.

Add concrete references.

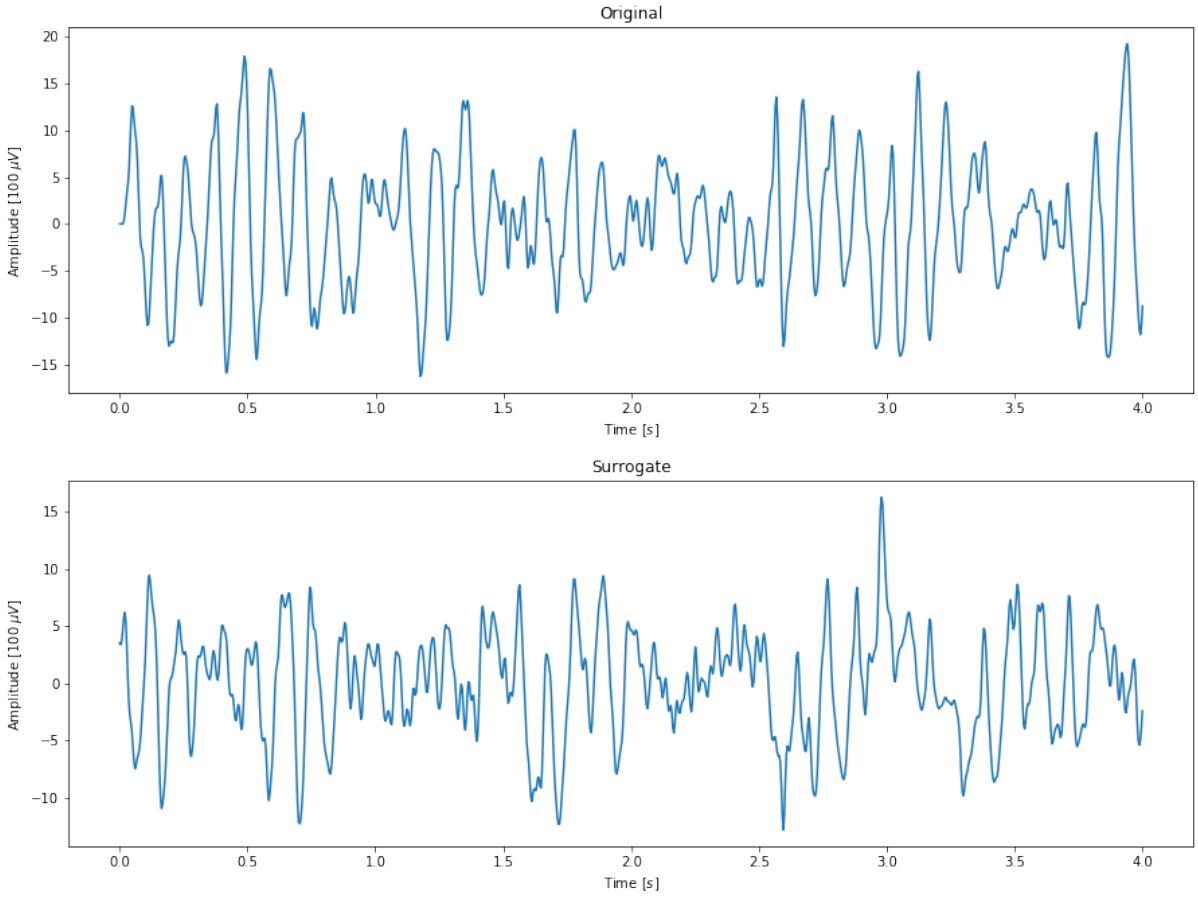


Figure 2.3: The first 4 *s* of a time series and its surrogate.

Figure 2.4 shows reconstructed trajectories for the first 4 s (1000 data points) of the recording, for varying time delay τ. As expected, the reconstructed attractors for small delays cluster along the main diagonal, expand, and then become increasingly chaotic with larger τ. However, it is impossible to judge objectively on the degree of folding in the attractor from these plots (even for shorter time series), which highlights the importance of qualitative measures for EEG signals.

Typical plots of autocorrelation and delayed mutual information can be seen on Figure 2.5. First local minima of DMI and first τ for which $A(\tau) \leq 1/e$ are marked by yellow dots. For this channel, these are $\tau_{\text{DMI}} = 7$ and $\tau_A = 6$. These values were computed for all channels of this recording, and their distribution for both DMI and autocorrelation can be seen in Figure 2.6. For this patient, autocorrelation shows less variance and lower suggested time delays. This behavior was observed across patients.

I should provide mean τ reported by $A(\tau)$ and DMI for all recordings.

Figure 2.7 shows singular values of the PCA reconstruction as functions of τ. The two prominent singular values correspond to the main axes of the attractor. We can see several collapses: smaller ones at $\tau = 5$ and $\tau = 7$, and larger one at $\tau = 14$, corresponding to the sharp peak of ILD on Figure 2.9. For $\tau = 3$, the three largest singular values show convergence - however, the small singular values suggests that the attractor has not fully unfolded in their corresponding directions. Overall, this behavior suggests $\tau_{\text{svd}} = 6$ as optimal. Note that results of this technique are difficult to evaluate by an automatic procedure.

Is this correct?

Not sure if this wording is precise and understandable.

The results obtained by ADFD can be seen in Figure 2.8. The average displacement tends to increase with m, is not monotonically increasing on its domain, and saturates for relavely small values of τ - thus, the estimated time delays are (consistently)lower than those obtained by other techniques. Moreover, this technique requires prior selection of m. However, the algorithms for selection of m, require estimation τ, making this technique impractical.

We need to smartly separate general observations with this analysis.

The result of ILD, the most powerful algorithm for estimation of the embedding parameters we used, can be seen on Figure 2.9. There is a clear minimum at $\tau_{\text{ILD}} = 4$, and the ILD curves become very similar for approximately all $m \geq 10$, except near the minimum, where they converge slower. Almost identical behavior was observed across all channels in this recording. This algorithm is computationally expensive (it takes around an hour to generate a single plot), and so is impractical for large datasets.

There are interesting patterns in these plots across patients and channels.

As explained in Section 1.4.4, these techniques should be used only as inspection tools, not as reliable guides for selection of τ. The ultimate goal of the reconstruction is to obtain as accurate values of the non-linear parameters as possible, and thus selection of the optimal embedding parameters may differ for each of them.Thus, for example, in order to select the proper embedding parameters for computation of the largest Lyapunov exponent, we inspected the scaling regions for multiple values of m, τ, Theiler window, and other parameters, and picked those with the longest scaling regions (since the length of the scaling regions is proportional to the certainty of the estimate [50]).

Find some studies doing this also. Is there a way to justify this theoretically?

Table 2.1 shows an overview of estimated values of τ. Autocorrelation, DMI, and singular values analysis report lower values than ADFD and ILD. However, Rosenstein notes that the best estimates of largest Lyapunov exponents were obtain for the autocorrelation threshold of $1 - 1/e$. For this threshold, the autocorrelation suggests $\tau_A = \tau_{ILD} = 4$ as optimal (and the distributions shift accordingly), thus in agreement wil ILD.

Add description of Theiler window to 1.5.1.1

In the Section 2.5.1, we will show the effects of increasing τ on the average divergence.

Should I implement some better methods? MI and acorr are not practical for EEG and other high dimensional systems.

2.4.2 Embedding dimension

For estimating the embedding dimension, we used combination of *false nearest neighbors* (FNN) algorithm described in Section 1.4.5.1 and average false neighbors (AFN) described in Section 1.4.5.2. The convergence of ILD curves and saturation of correlation dimension also provides insight into optimal choice of embedding dimension.

	Optimal time delay estimate
Reconstruction plot	-
Autocorrelation	6, 4
Delayed mutual information	7
Singular values analysis	6
Average displacement	2, 3
Integral local deformation	4

Table 2.1: Optimal time delay estimates of individual techniques for patient 75, second session.

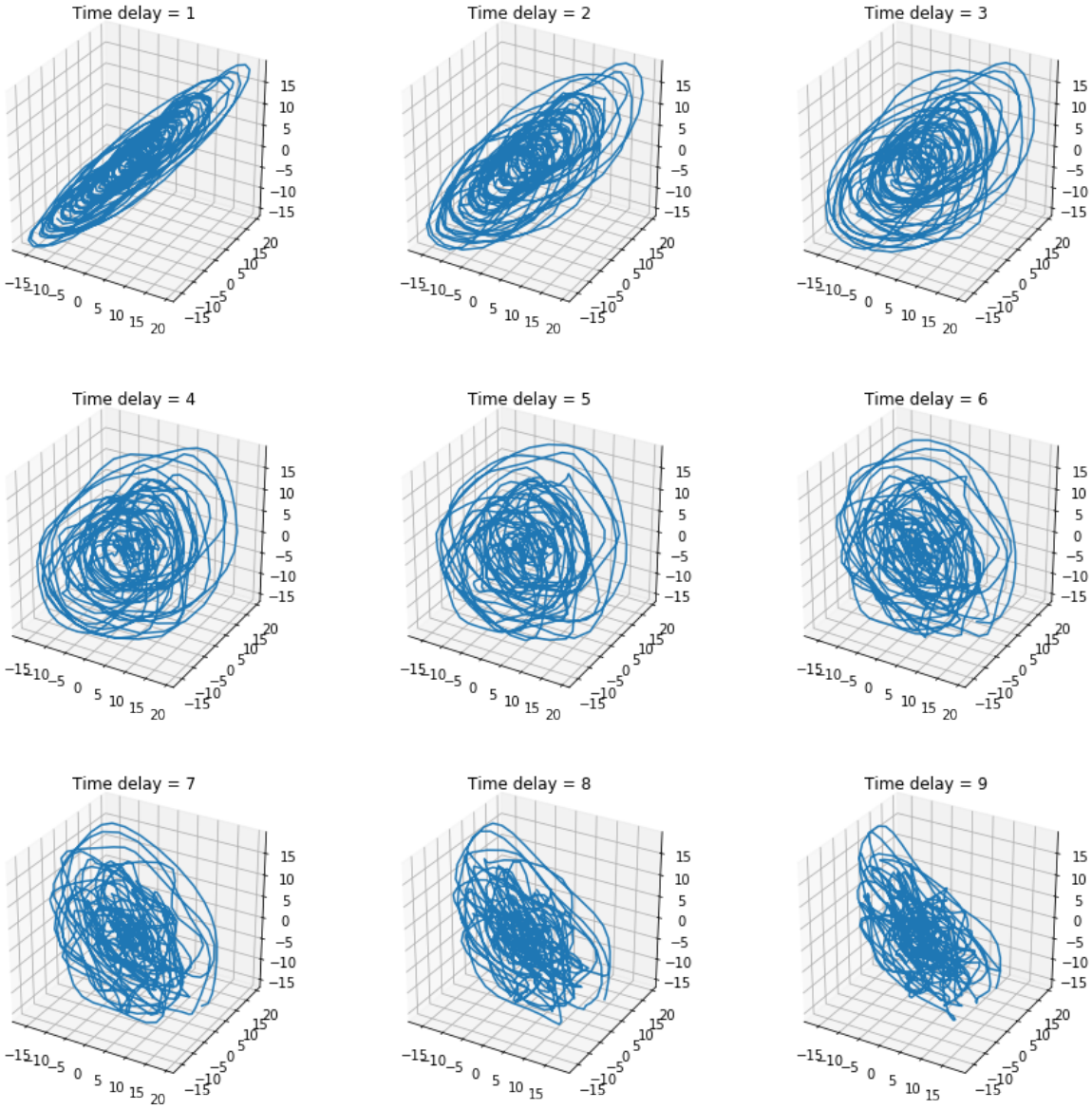


Figure 2.4: 3D time delay reconstructions for various values of τ.

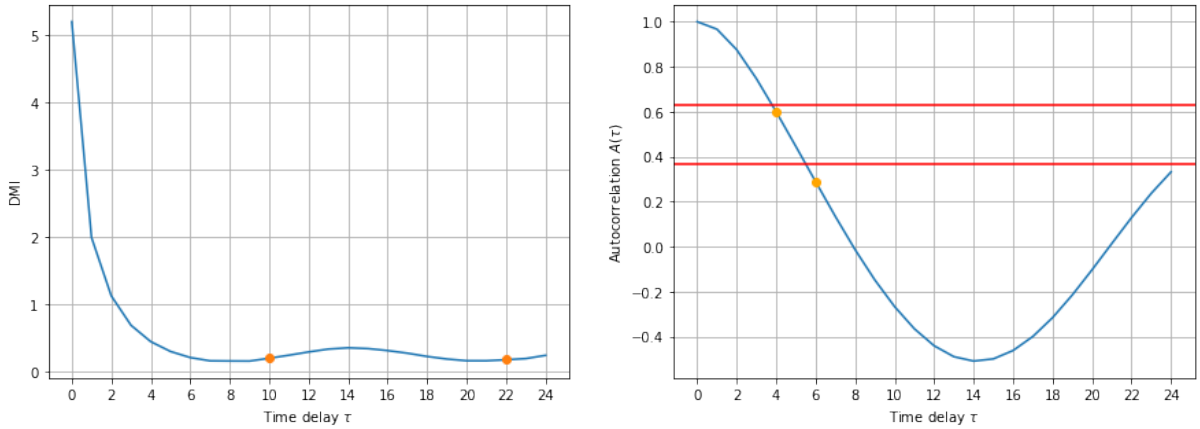


Figure 2.5: Delayed mutual information and autocorrelation as functions of τ. The red line shows threshold values $1 - 1/e$ and $1/e$ respectively. The plots of surrogate data are equivalent.

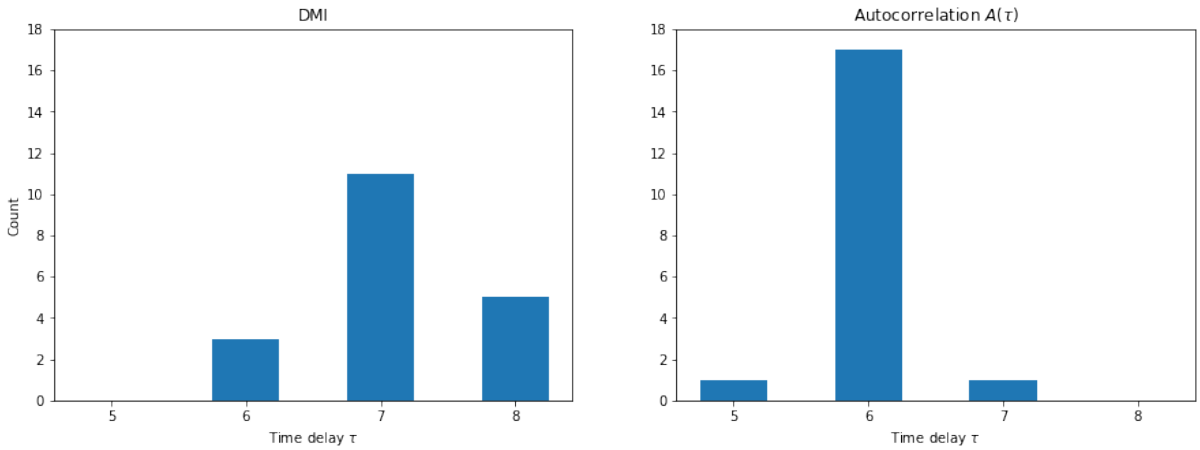


Figure 2.6: Distributions of time delays computed using delayed mutual information and autocorrelation for threshold $1/e$.

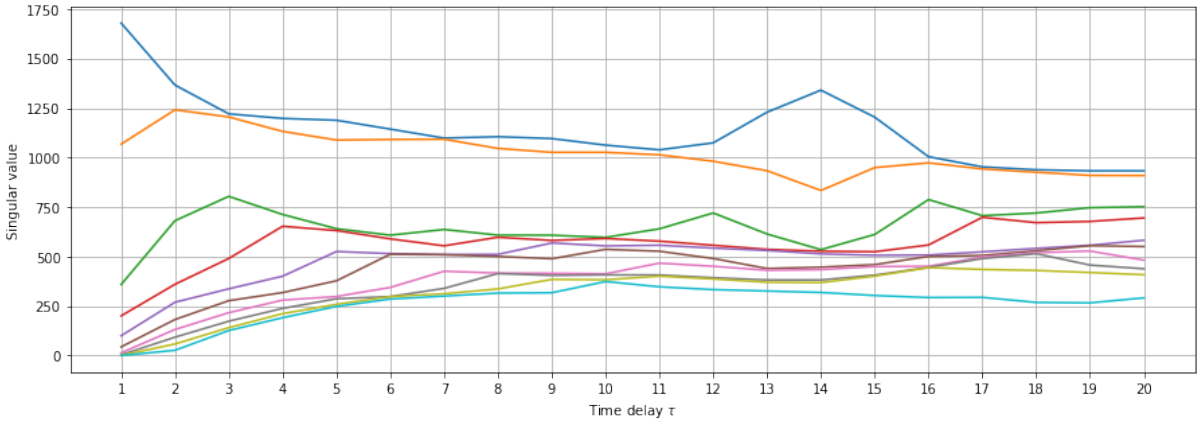


Figure 2.7: Plot of singular values as functions of τ for $m = 10$.

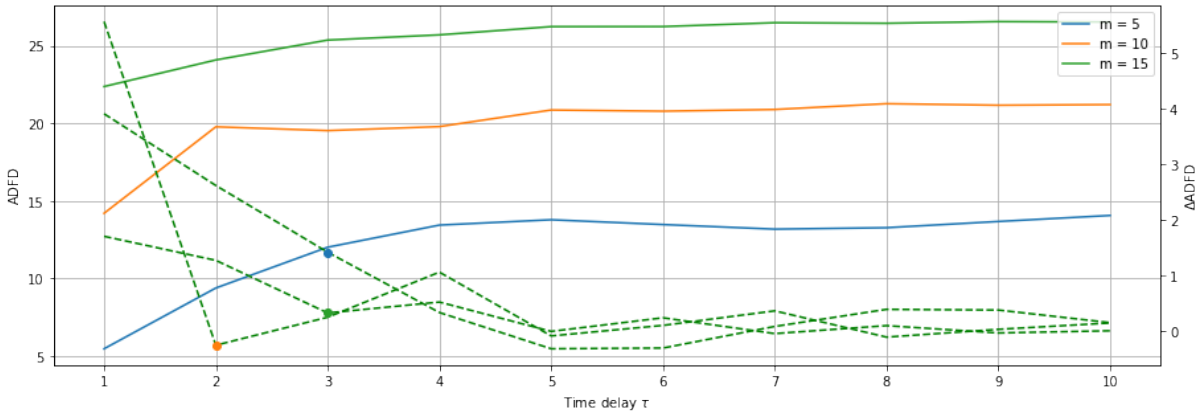


Figure 2.8: Plot of average displacement from diagonal for $m = 10$.

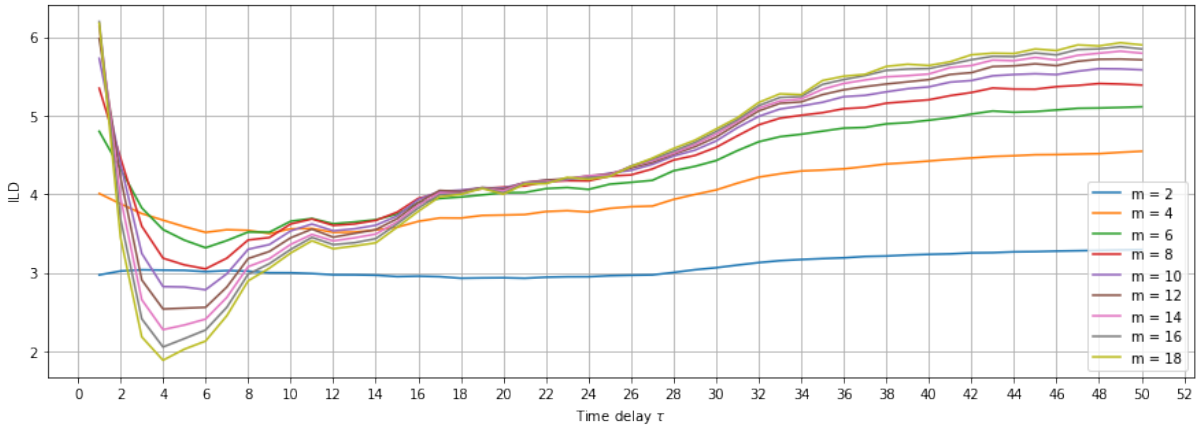


Figure 2.9: Plot of integral local deformation. The parameters used for this computation are $q_{\max} = 10$, $t_e = 3$, $N_{\text{ref}} = N_v$, $k = 20$ and $w_t = 10$.

The percentage of reported false neighbors depends strongly on the selected values of R and A from equations (1.8) and (1.9). This is illustrated on Figure 2.10, showing the percentage of false neighbors reported by the respective criteria for varying values of A and R, and for several values of time delay τ.

The percentages reported by the criterion I are almost independent of τ, whereas increasing τ tends to increase the percentage reported by criterion II. For high enough τ, criterion II will report all neighbors as false.

The apparent independence of the results of the criterion I on τ is indicates that, regardless of τ, the same percentage of near neighbors changes their distance proportionally with increase in m. As explained in Section 1.4.5.1, this behavior that can be expected of randomly generated uniformly distributed sequence of numbers. Indeed, behavior of the criterion II is consistent with this hypothesis - it eventually increases to 100% for all values of A, essentially indicating infinite dimension.

Actually explain it there - nearest ≠ close, etc. . . , [50]

By selecting proper parameters and using both criteria cojointly, however, FNN can still be used to obtain reasonable results, consistent with estimates obtained by ILD and AFN.

Report average m computed by ANN and FNN, $R = 2.5$, $A = 2.0$, $\Delta E_1 \leq 0.005$ for this patient using a histogram.

The E_1 statistic of AFN usually stops increasing for approximately the same value as reported by criterion I of FNN for $R = 2.5$, see Figure 2.11. The E_2 statistic, tends to oscillate in small neighborhood of value 1, which is an indecation of nondeterminism [18].

We plotted the value of correlation dimension against m for various values of τ (see Figure 2.16) - for details about the computation, see Section 2.5.2.

Add statistics of m computed this way.

2.5 Estimation of non-linear features

2.5.1 Largest Lyapunov exponents

For all computations of the largest Lyapunov exponent, we used the Rosenstein's algorithm [79] described in Section 1.5.1.1, with Theiler window w_t length of 50 (200 ms). We found that the results were similar for values w_t of 10, 50, 100 and 1000.

Why? This is unexpected.

Figure 2.12 shows divergence plots for different values of the embedding dimension m and time delay τ. Let us remind the reader that longer scaling regions correspond to higher certainty of the estimate. The short scaling regions and high slopes for small embedding dimension may appear because, when the attractor is not unfolded, near neighbors are not actually close in the phase space and thus their trajectories diverge quickly. With increasing embedding dimension the scaling region clearly lengthens, but the slope also slowly approaches zero, and scaling region gradually disappears. Therefore, selecting proper embedding dimension based on divergence plots is a balancing act between those two effects. Moreover, notice that the length of the scaling region is approximately $m\tau$.

How to explain this?

With increasing time delay τ, we observe gradually damped oscillation-like behavior with period τ and amplitudes also increasing with τ. Average divergence computed using Kantz' algorithm also exhibits this behavior. The explanation is as follows: let $x_1, x_2, \ldots, x_N$ represent equidistantly sampled time series, and $y_i \in \mathbb{R}^m$ an embedded point in the reconstructed orbit. Then

$$\begin{aligned}
y_i &= \begin{pmatrix} x_i & x_{i+\tau} & \ldots & x_{i+(m-2)\tau} & x_{i+(m-1)\tau} \end{pmatrix} \\
y_{i+\tau} &= \begin{pmatrix} x_{i+\tau} & x_{i+2\tau} & \ldots & x_{i+(m-1)\tau} & \mu_1 \end{pmatrix} \\
&\ldots \\
y_{i+(m-1)\tau} &= \begin{pmatrix} x_{i+(m-1)\tau} & \mu_1 & \ldots & \mu_{m-2} & \mu_{m-1} \end{pmatrix},
\end{aligned}$$

and so if $x_i \approx x_{i+\tau}$ for enough i, then $y_i \approx y_{i+\tau} \approx y_{i+2\tau} \approx$, and this oscillation with period τ gradually vanishes over $m - 1$ periods.

Why doesn't this happen always?

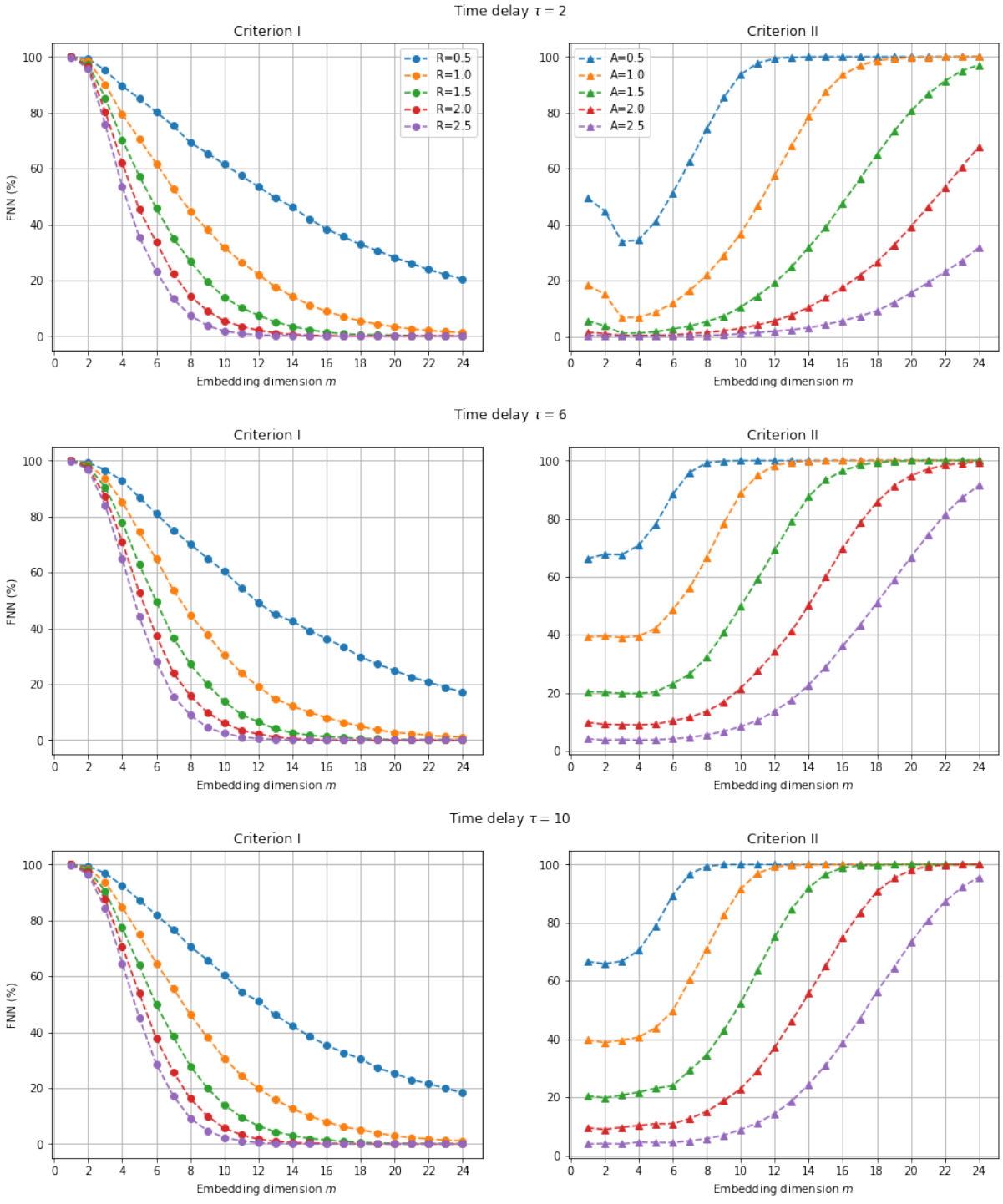


Figure 2.10: The effect of values of the tolerance parameters on the percentage of false neighbors reported by I. criterion (1.8) and II. criterion (1.9), Theiler window $w_t = 50$.

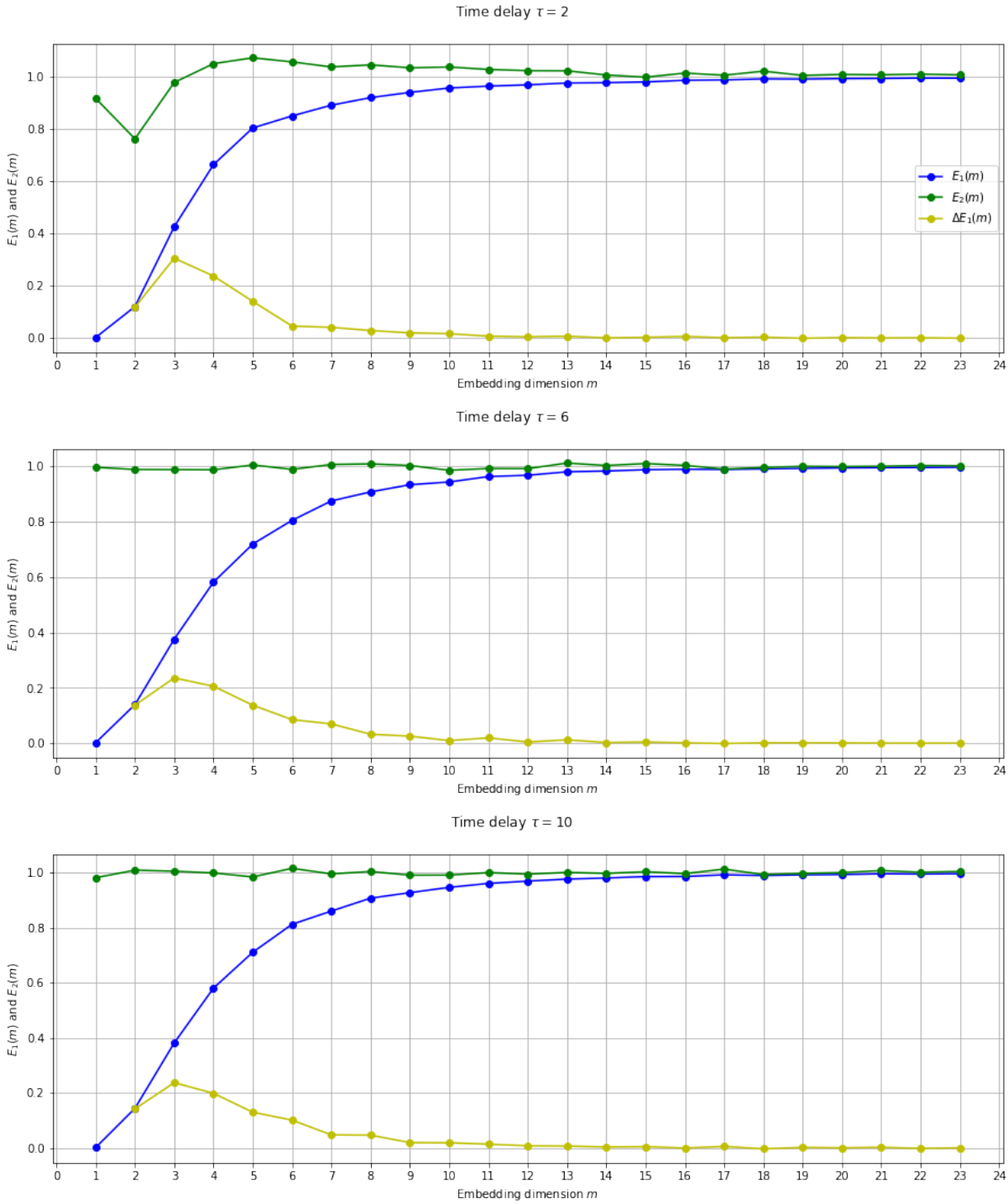


Figure 2.11: The results of AFN for varying values of time delay τ, Theiler window $w_t = 50$.

This effect can be aleviated by choosing smaller τ, but we are unaware of any way of eliminating it completely for Rosenstein's algorithm. [1] Another metric to optimize after length of the scaling region is, therefore, reduction of the degree of deformation of the scaling region by the periodic oscillations.

Are there others? Somehow introduce randomness?

We computed LLE for fixed m and τ, and then by selecting them automatically. Describe the reasoning for selecting the fixed ones, and the process of automatic selection via the method in the previous section.

Oscillation-like behavior was observed for white noise data in [79], and for periodic data with period equal to the dominant period of the system in [48].

Can this occur due to measurement projection? Also, even if the largest Lyapunov exponent is positive, in dissipative systems (i.e. those possessing an attractor, see Section 1.3.3) the sum of all Lyapunov exponents is negative, and thus, even on average, states will diverge in some directions. These effects can be compensated for by using proper averaging statistics [48].

We observe very similar behavior of the average divergence for the surrogate data. This, together with the observations made in previous sections, gives rise to the hypothesis of lack of chaos in the data. We tested the hypothesis of linear Gaussian process in Section 2.5.8.

Good thing is that Eckmann's algorithm gives similar results with very different approach. Maybe we shoud incorporate this somehow?

To compute the LLE estimates with automatic selection of proper embedding parameters, we proceeded as follows. First, we found the 60 s subsection of the time series with the lowest p-value of the χ^2 stationarity test using moving window of length 15000 and slide 100. Selection of time delay was done using autocorrelation function with threshold $1 - 1/e$. The selected τ was used to compute the embedding dimension with smallest FNN percentage from embedding dimensions in range from 1 to 20, i.e. $m_1 = \arg\min_{m' \in \{1,\dots,20\}} \mathrm{FNN}(m')$. The tolerance parameters wer $R = 2.5$, $A = 2.0$ and $w_t = 50$. Moreover, we found the first embedding dimension m_2 for which $E_1(m_2) - E_1(m_2 - 1) < 0.008$. The selected embedding dimension was $m = \lceil m_1 + m_2)/2 \rceil$. The length of the scaling region $t_{\max} = m\tau$ and the Theiler window $t_w = 50$.

Add the average m's computed this way.

2.5.2 Correlation dimension

For corr dim, we also used two ways to compute it - automatic and fixed. Explain here why have we chosen $m = 10$ and $\tau = 3$.

To compute the correlation sum $C(r)$, we used classical Grassberger-Procaccia algorithm described in Section 1.5.2 using Chebyshev metric, $w_t = 50$, for values of r either in geometrical progression of 100 values from 0.05 to 10 or by an automatic procedure described further. Then, these $(r, C(r))$ pairs were used to compute local least square fits of the equation $C(r) = r^{d_2}$ inside windows of length 7 for each pair.

This paragraph can be much improved (wording, etc.).

Figure 2.13 shows log-log plots of normalized correlations sum $C(r)$ against radius r for varying values of time delay τ. There are clear straight lines indicating expected relationship $C(r) \propto r^{d_2}$. We can see that the lines shift to the right, increasing their slopes with m. The correlation sum is almost independent of time delay.

Figure 2.14 shows the log plot of local slope of of $\log C(r)$ as a function of r. There are no apparent scaling regions at all. Morever, by comparing with the same plot for iAAFT surrogate of the same time series (see Figure 2.15), we cannot even reject the hypothesis of a linear stochastic process.

We decided to compute the correlation dimension as follows. First, as with computation of the Lyapunov exponent, we use a moving window of length 15000 datapoints and shift 100 to locate 60 s section of the time series with the lowest p-value for the χ^2 stationarity test described in Section . Then, we create embeddings for embedding dimensions in range from 2 to 30 with the optimal time lag selected according to the autocorrelation function with threshold $1 - 1/e$. For each embedding, we evaluate the slope of $\log C(\log r)$ on the interval $[r_{\mathrm{lower}}, r_{\mathrm{upper}}]$, where r_{lower} corresponds to the average nearest neighbor distance on the reconstructed attractor, r_{upper} is given by

Add description of the test.

$$\log r_{\mathrm{upper}} = \log r_{\mathrm{lower}} + \frac{1}{10}\left(\log r_{\max} - \log r_{\mathrm{lower}}\right),$$

[1] There are algorithms, such as modifications of Wolf's algorithm [78], whose results are almost independent on τ (as is theoretically expected).

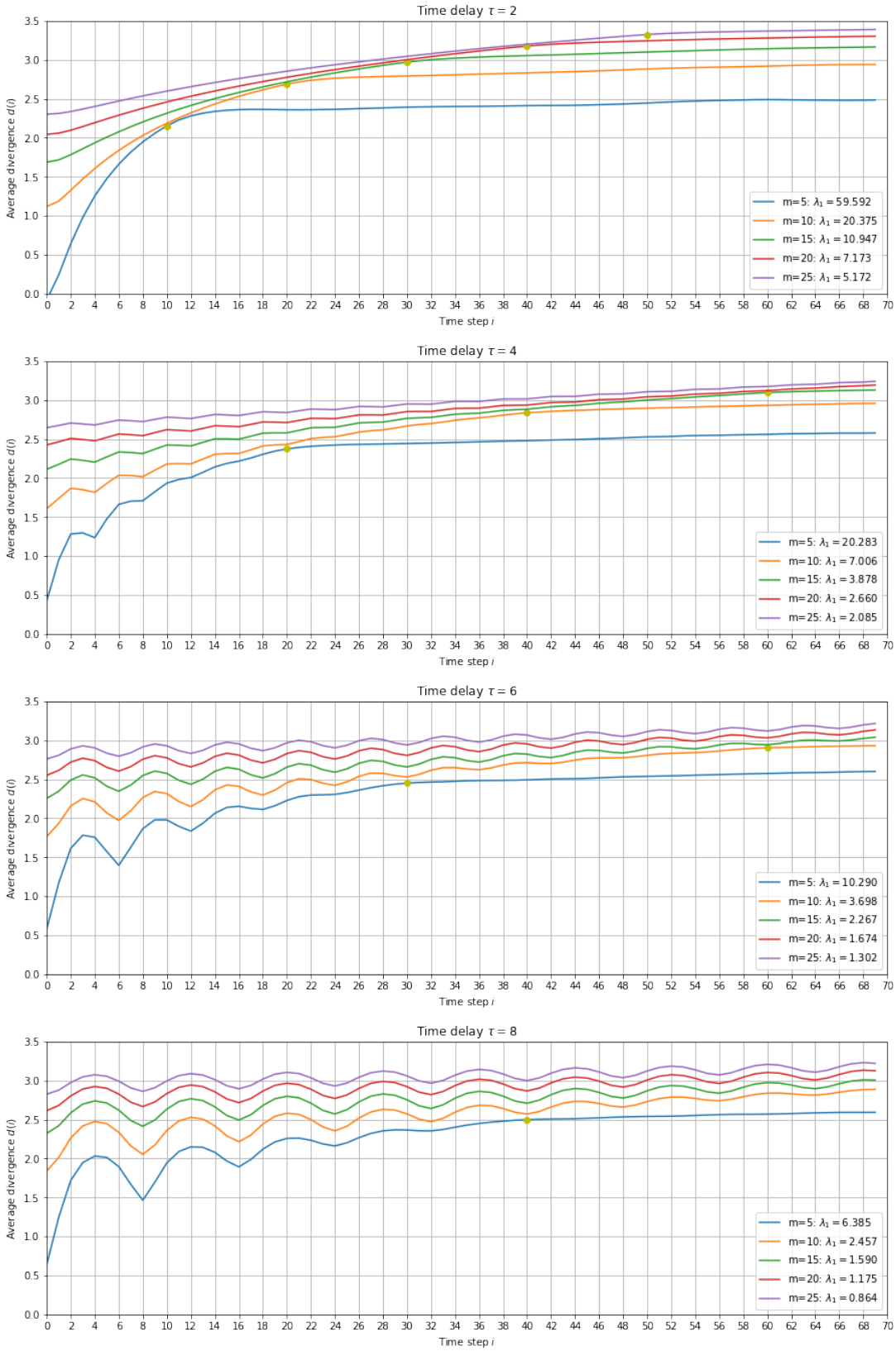


Figure 2.12: Average divergence plots for varying values of m and τ.

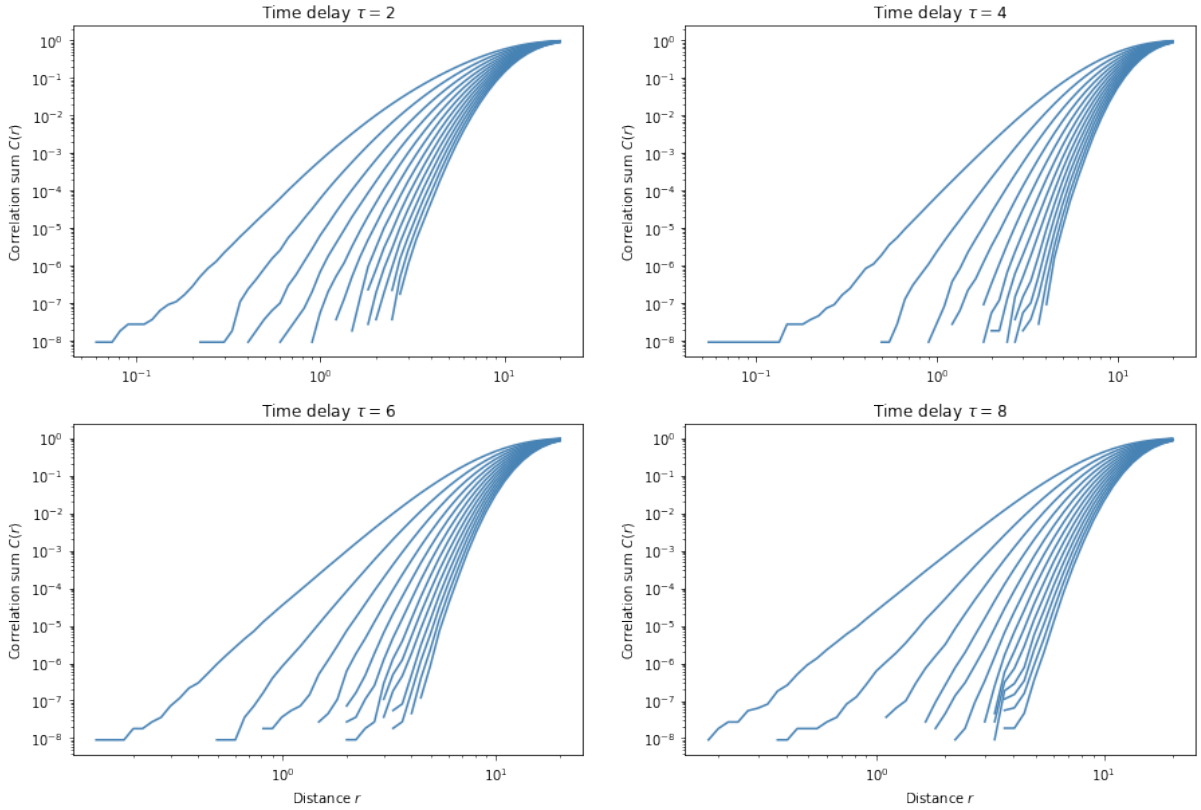


Figure 2.13: Normalized correlation sum as a function of radius r for dimensions in range from 5 to 30 (from left to right).

where $r_{\max}$ denotes the largest ocurring pairwise distance on the attractor. This approach of automatic selection radius bounds for evaluation of d_2 is borrowed from [6].

Figure 2.16 shows d_2 computed this way as a function of the embedding dimension m for varying values of the embedding dimension τ. There d_2 are no signs of saturation, correlation dimension reaches a global maximum and then starts to decrease.

Conclusion? No finite value, or no chaos?

2.5.3 Detrended fluctuation analysis

Show plots, how and why we chose the params so that the fit is optimal.

Figure 2.17 how we fitted the scaling region in the log-log plot of $F(n)$.

Moreover, following the suggestions in [36], we tried computing DFA of the envelope of the signal band-pass filtered to beta frequency (3-7 Hz). However, the resulting values were too large, and we found no differences between the studied groups.

2.5.4 Hurst exponent

Generally, the number of values n for which to calculate the scaled range $R(n)/\sigma(n)$ is a tradeoff between the length and number of the considered subsequences. In order to avoid both small and large values of n, we used 15 values of n spaced evenly on the middle 25 % of the logarithmic scale between 0 and $\ln N$

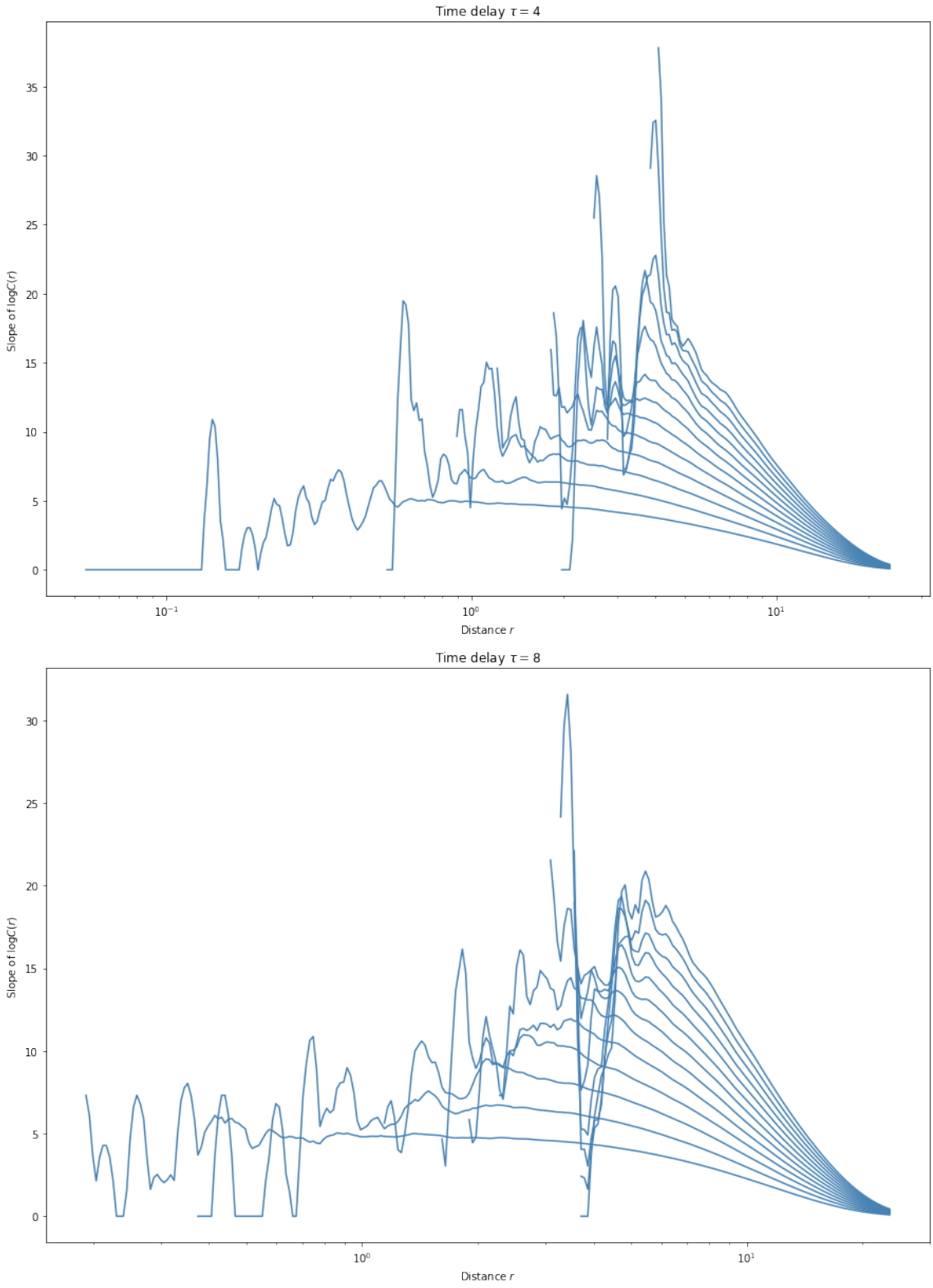


Figure 2.14: Local correlation dimension d_2 as a function of radius r for dimensions in range from 5 to 30 (from bottom to top) and time delays $\tau = 4$ and $\tau = 8$.

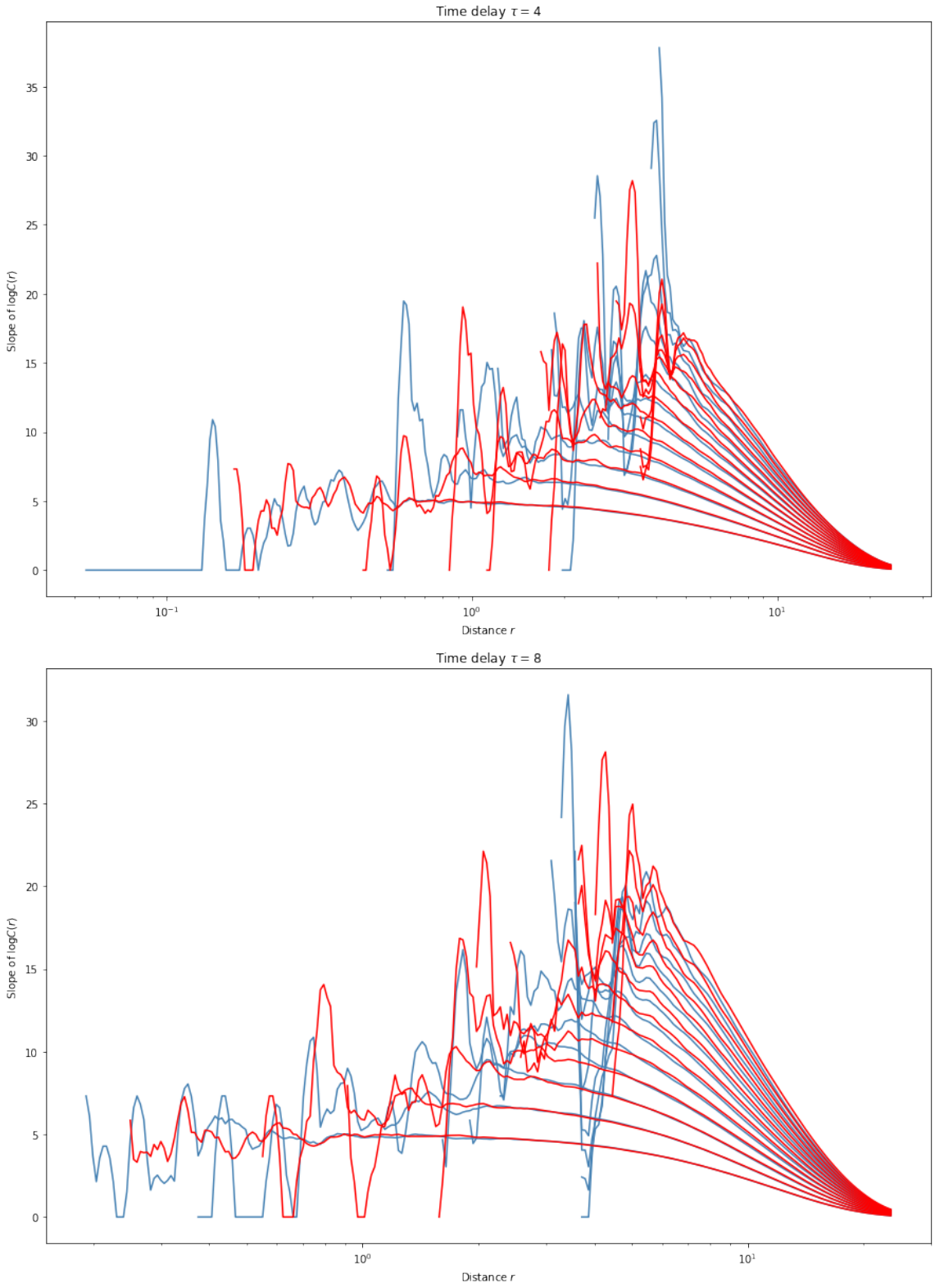


Figure 2.15: Local correlation dimension d_2 as a function of radius r for dimensions in range from 5 to 30 (from bottom to top) and time delays $\tau = 4$ and $\tau = 8$ for the original series (blue) and its surrogate series computed using iAAFT.

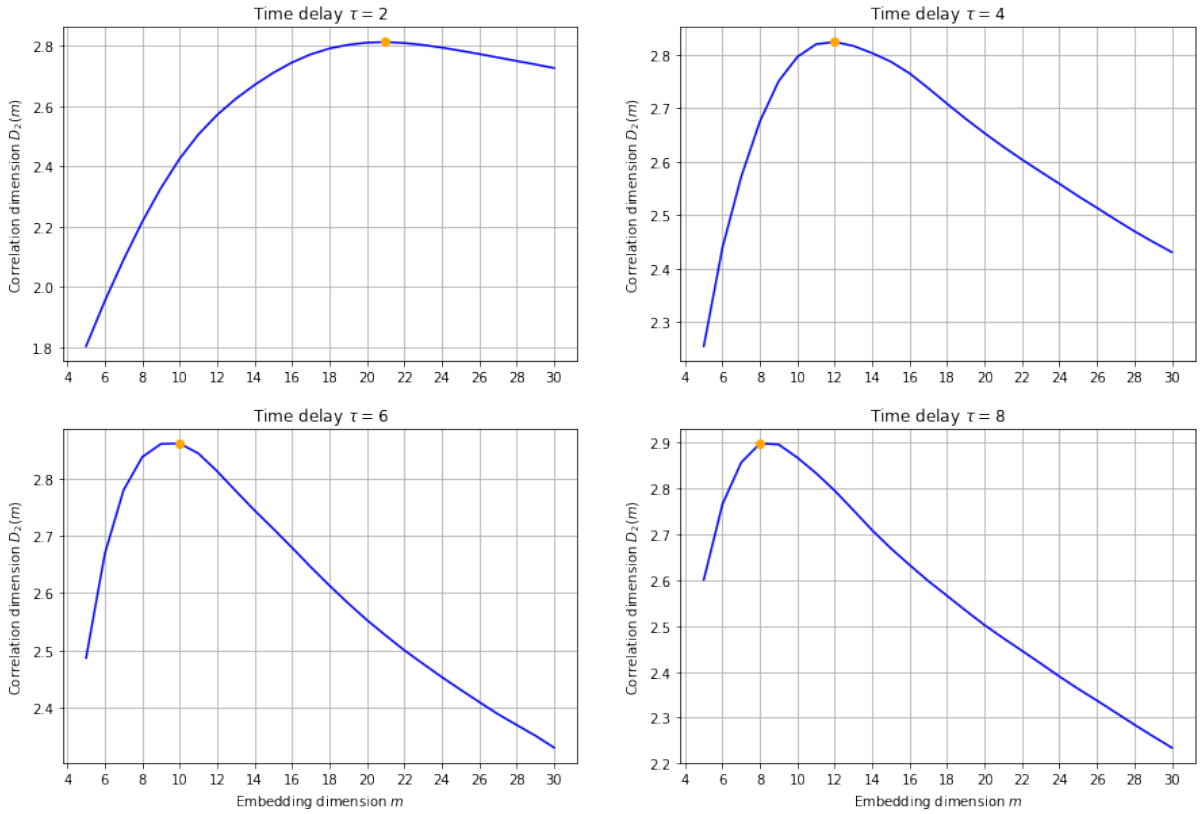


Figure 2.16: Correlation dimension as function of the embedding dimension m.

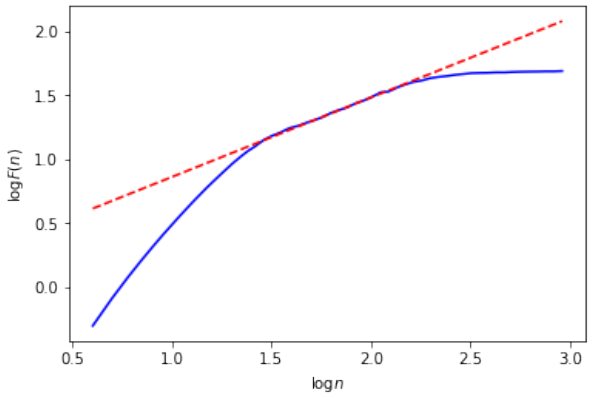


Figure 2.17: Computation of DFA.

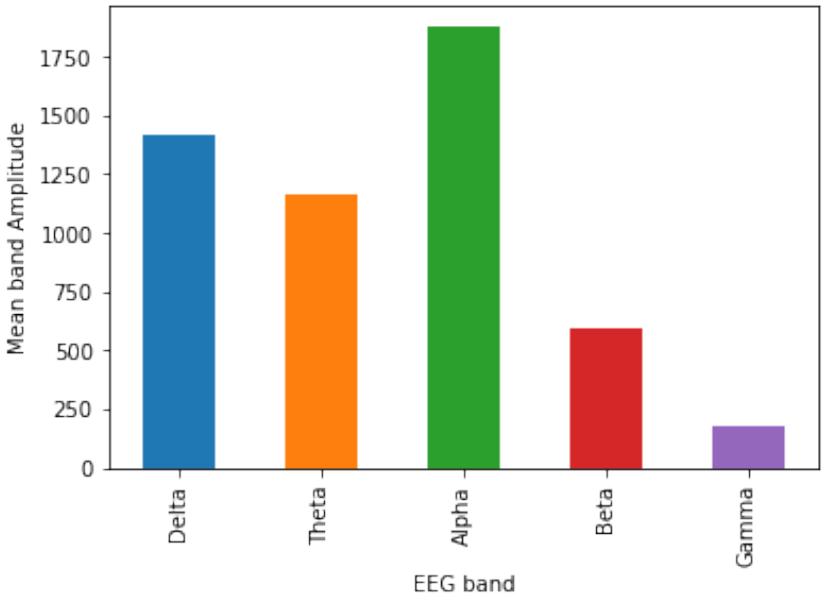


Figure 2.18: Typical range of mean band amplitudes in channel FP1.

2.5.5 Higuchi fractal dimension

We could have tried optimizing the params more.

2.5.6 Sample entropy

Time lag 1.

2.5.7 Frequency band amplitudes

<++> We also analyzed mean frequency amplitudes in alpha, beta, gamma, delta and theta frequency bands, and found no differences between the studied groups. See Figure 2.18.

2.5.8 Surrogate analysis

As mentioned in Section 1.6, surrogate analysis for high condifence levels requires generating surrogate dataset tens of times larger than the original dataset, and computing corresponding non-linear measures for these surrogate signals. To achieve confidence level $\alpha = 95\%$ for our dataset, this translates to generating $266 * 19 * 38 = 192052$ surrogate samples and computing on them each non-linear measure considered in our study. Since non-linear algorithms considered (described in the preceding subsections) are relatively computationally expensive, performing surrogate analysis for all measures and all patients is computationally infeasible. Thus, we analyzed each algorithm (with varying parameters) on a single recording (patient number 75, second session), using 19 surrogate samples. For generating the surrogate data, we used the iAAFT algorithm described in Section 1.6.0.1. Note that to perform this as a test of non-linearity, we have to assume that the choice of embedding parameters for the algorithms is correct.

An example of a result of such analysis for the largest Lyapunov exponent (embedding dimension 10, time delay 3) can be seen in Figure 2.19; the results for other measures were similar. First, we can

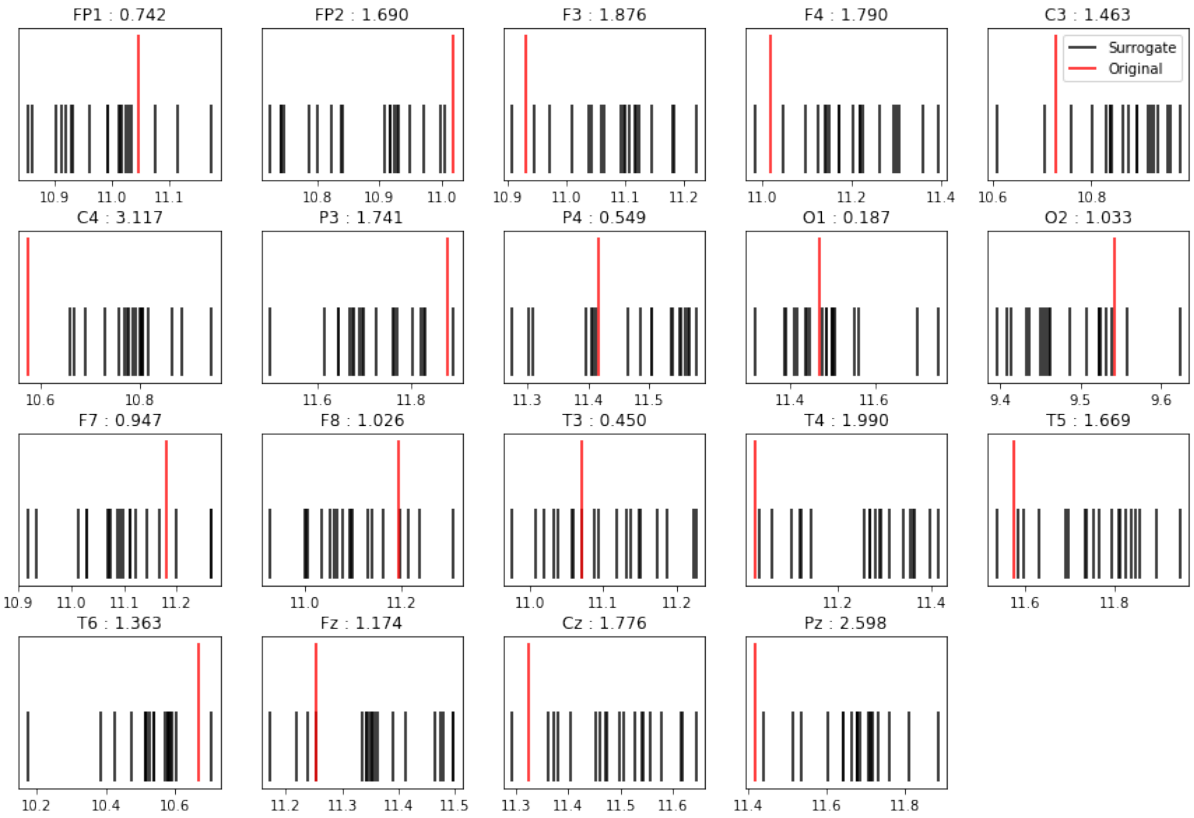


Figure 2.19: Example distribution of the largest Lyapunov exponent (embedding dimension 10, time delay 3) for 19 surrogate samples and the original for all channels. The number next to each channel name represents the confidence in sigma, computed as in equation 1.16.

observe that the distribution of the values computed for the surrogate data does not seem normal for all channels. As mentioned in Section 1.6, this increases the required value sigma to achieve the same confidence, or requires performing a rank based test. It can be easily observed, then, that based on the rank based test, the hypothesis of a linear stochastic process cannot be rejected on (admittedly relatively low) confidence level $\alpha = 1 - 2/(19+1) = 90\%$ for all channels except FP2, C4, T4, Pz. Obviously, does not necessarily imply that that the process underlying corresponding time series is stochastic, because, for example, there still may be other non-linear measures (or different choice of embedding parameters) which can discriminate between the original time series and the surrogate data. Neither does it suggest that the choice of embedding parameters is incorrect, because the process underlying corresponding time series may be stochastic. It is simply a failed attempt at disproving the null hypothesis of a stochastic linear process. Moreover, all our analyses concerned only a single patient.

2.6 Analysis of measure distributions between groups

2.6.1 Before and after treatment

As the first step of our analysis, we conducted an investigation of the differences in the non-linear measures computed from the signals obtained before and after treatment. The purpose of this inquiry is to determine brain regions and measures affected by treatment. This is warranted by the fact that

This should be cited!

the patterns in EEG signals tend to be relatively stable over time. On the other hand, we realize the limitations of this attempt in the case of this study, since each patient recieved personalized method of treatment, and the methods may have differing impact.

We separated the patients into terciles according to the ratio of the depression scores before and after treatment. Out of these three populations, we selected the first and last, containing 46 and 44 samples respectively, to obtain population we call *responding* (responders) to treatment and *non-responding* (non-responders) to treatment. The second tercile was not considered in this analysis to minimize the effect of inaccuracy of the self-reported depression score. Comparison of mean values of individual measures between the two populations can be seen in Figures 2.21, 2.22 and 2.23. Length of an error bar corresponds to one standard deviation.

Kruskal is better, redirect to distributions.

For each group, we performed two-sided Kolmogorov-Smirnov test for the null hypothesis that the distributions of values computed for measurements before and after treatment are the same. No significant differences in distributions were found for d_2 computed using automatic selection of embedding parameters so in this section, we used λ_1 and d_2 computed for $m = 10$, $\tau = 3$, $w_t = 50$. Moreover, we found no significant differences in DFA, so we decided to leave it out of this analysis. The results can be seen in Tables 2.2, 2.3 and 2.4.

However, we found that responders had significantly lower λ_1 computed using these methods in C3 and C4 electrodes (6.899 ± 1.278 vs. 7.342 ± 1.838 for C3, 6.731 ± 1.116 vs. 7.365 ± 1.475 for C4) on recording performed before treatment.

For all measures computed this way, we found significantly differences in temporal areas, especially T3. The distributions of Largest Lyapunov exponents were also significantly different in the frontal and "central" areas, whereas d_2 differed mainly in prefrontal areas.Sample entropy mimics the pattern seen in λ_1, differing mainly in frontal and "central" areas.

We also performed unsupervised analysis of before / after groupds using PCA in 2,3, and 4 dimensions, and compared centroids and mean distances between before and after treatment recording for each group. However, the resulting plots and heatmaps are featureless and thus we will leave them out. The mean distances are also uninformative.

Which are associated with depression, but we want to leave that out in this section.

Change the tables and text to using Kruskal test instead of KS. Justify by saying the distributions are not generally normal.

Channel	Before	After	p-value	Sig.
mean	10.151 ± 0.950	9.919 ± 1.074	0.121	
std	0.628 ± 0.239	0.724 ± 0.295	0.089	
FP1	9.770 ± 1.130	9.545 ± 1.287	0.432	
FP2	9.764 ± 1.186	9.565 ± 1.281	0.432	
F3	9.794 ± 1.082	9.493 ± 1.177	0.065	*
F4	9.862 ± 1.090	9.413 ± 1.330	0.010	***
C3	9.846 ± 1.068	9.579 ± 1.117	0.089	
C4	9.922 ± 1.046	9.598 ± 1.196	0.033	**
P3	10.447 ± 0.865	10.291 ± 1.055	0.212	
P4	10.437 ± 0.883	10.266 ± 1.046	0.832	
O1	10.539 ± 1.174	10.485 ± 1.271	0.965	
O2	10.518 ± 1.198	10.409 ± 1.312	0.273	
F7	10.096 ± 1.351	9.886 ± 1.402	0.432	
F8	10.118 ± 1.297	9.785 ± 1.545	0.273	
T3	9.872 ± 1.308	9.387 ± 1.544	0.000	***
T4	9.842 ± 1.317	9.449 ± 1.534	0.065	*
T5	10.506 ± 1.092	10.329 ± 1.262	0.273	
T6	10.584 ± 1.087	10.380 ± 1.189	0.347	
Fz	10.257 ± 1.004	10.117 ± 1.096	0.161	
Cz	10.204 ± 0.906	10.075 ± 0.998	0.273	
Pz	10.490 ± 0.897	10.408 ± 1.032	0.735	

Table 2.2: Mean values of λ_1 of all patients before and after treatment.

Channel	Before	After	p-value	Sig.
mean	7.522 ± 0.441	7.593 ± 0.433	0.481	
std	0.383 ± 0.125	0.414 ± 0.165	0.071	*
FP1	7.812 ± 0.611	7.880 ± 0.704	0.387	
FP2	7.826 ± 0.650	7.935 ± 0.790	0.035	**
F3	7.594 ± 0.592	7.681 ± 0.586	0.179	
F4	7.639 ± 0.602	7.726 ± 0.582	0.387	
C3	7.342 ± 0.592	7.395 ± 0.591	0.585	
C4	7.334 ± 0.550	7.412 ± 0.574	0.387	
P3	7.274 ± 0.515	7.319 ± 0.522	0.305	
P4	7.325 ± 0.573	7.349 ± 0.506	0.888	
O1	7.539 ± 0.566	7.543 ± 0.524	0.987	
O2	7.516 ± 0.518	7.569 ± 0.547	0.387	
F7	7.680 ± 0.530	7.812 ± 0.550	0.305	
F8	7.702 ± 0.534	7.822 ± 0.565	0.179	
T3	7.669 ± 0.585	7.877 ± 0.624	0.011	***
T4	7.684 ± 0.588	7.840 ± 0.556	0.024	**
T5	7.556 ± 0.523	7.593 ± 0.481	0.585	
T6	7.536 ± 0.518	7.593 ± 0.483	0.585	
Fz	7.339 ± 0.535	7.350 ± 0.525	0.987	
Cz	7.359 ± 0.566	7.354 ± 0.533	0.998	
Pz	7.199 ± 0.494	7.210 ± 0.543	0.888	

Table 2.3: Mean values of d_2 of all patients before and after treatment.

Channel	Before	After	p-value	Sig.
mean	0.761 ± 0.108	0.790 ± 0.130	0.240	
std	0.071 ± 0.040	0.086 ± 0.048	0.094	
FP1	0.804 ± 0.149	0.837 ± 0.176	0.403	
FP2	0.802 ± 0.156	0.830 ± 0.175	0.403	
F3	0.800 ± 0.132	0.839 ± 0.156	0.179	
F4	0.790 ± 0.137	0.842 ± 0.168	0.046	**
C3	0.793 ± 0.122	0.825 ± 0.147	0.314	
C4	0.781 ± 0.126	0.821 ± 0.151	0.046	**
P3	0.720 ± 0.087	0.740 ± 0.115	0.619	
P4	0.720 ± 0.093	0.736 ± 0.116	0.975	
O1	0.707 ± 0.113	0.718 ± 0.134	0.734	
O2	0.712 ± 0.113	0.732 ± 0.154	0.314	
F7	0.786 ± 0.163	0.811 ± 0.176	0.619	
F8	0.781 ± 0.156	0.821 ± 0.195	0.403	
T3	0.806 ± 0.160	0.867 ± 0.197	0.006	***
T4	0.812 ± 0.167	0.861 ± 0.197	0.131	
T5	0.723 ± 0.110	0.743 ± 0.133	0.403	
T6	0.714 ± 0.112	0.729 ± 0.123	0.506	
Fz	0.747 ± 0.107	0.762 ± 0.124	0.506	
Cz	0.756 ± 0.096	0.767 ± 0.110	0.840	
Pz	0.716 ± 0.093	0.728 ± 0.113	0.996	

Table 2.4: Mean values of sample entropy of all patients before and after treatment.

Channel	Before	After	p-value	Sig.
mean	9.994 ± 0.890	9.655 ± 1.064	0.022	**
std	0.639 ± 0.229	0.701 ± 0.267	0.452	
FP1	9.599 ± 1.086	9.335 ± 1.360	0.625	
FP2	9.590 ± 1.090	9.281 ± 1.293	0.308	
F3	9.588 ± 1.119	9.190 ± 1.190	0.123	
F4	9.682 ± 0.999	9.199 ± 1.339	0.072	*
C3	9.690 ± 1.065	9.349 ± 1.111	0.072	*
C4	9.827 ± 1.052	9.407 ± 1.221	0.041	**
P3	10.294 ± 0.797	10.032 ± 1.050	0.072	*
P4	10.265 ± 0.873	10.004 ± 1.104	0.308	
O1	10.343 ± 1.081	10.117 ± 1.176	0.452	
O2	10.261 ± 1.160	9.961 ± 1.212	0.123	
F7	9.998 ± 1.324	9.682 ± 1.419	0.199	
F8	9.991 ± 1.170	9.659 ± 1.492	0.308	
T3	9.789 ± 1.387	9.172 ± 1.492	0.005	***
T4	9.703 ± 1.261	9.164 ± 1.403	0.022	**
T5	10.370 ± 1.091	10.073 ± 1.214	0.011	***
T6	10.335 ± 0.954	10.021 ± 1.166	0.123	
Fz	10.096 ± 0.970	9.849 ± 1.126	0.072	*
Cz	10.150 ± 0.886	9.847 ± 1.006	0.123	
Pz	10.318 ± 0.805	10.113 ± 1.062	0.801	

Channel	Before	After	p-value	Sig.
mean	10.400 ± 0.969	10.015 ± 1.088	0.423	
std	0.623 ± 0.260	0.813 ± 0.370	0.018	***
FP1	10.034 ± 1.166	9.510 ± 1.325	0.108	
FP2	10.045 ± 1.196	9.752 ± 1.269	0.778	
F3	10.116 ± 1.034	9.619 ± 1.200	0.108	
F4	10.098 ± 1.146	9.343 ± 1.364	0.034	**
C3	10.160 ± 1.010	9.585 ± 1.147	0.018	***
C4	10.162 ± 1.060	9.681 ± 1.170	0.062	*
P3	10.711 ± 0.874	10.468 ± 1.065	0.423	
P4	10.720 ± 0.897	10.453 ± 1.005	0.595	
O1	10.765 ± 1.292	10.772 ± 1.352	0.924	
O2	10.823 ± 1.242	10.604 ± 1.434	0.595	
F7	10.234 ± 1.340	9.875 ± 1.459	0.282	
F8	10.307 ± 1.405	9.556 ± 1.715	0.108	
T3	10.073 ± 1.207	9.292 ± 1.602	0.004	***
T4	10.018 ± 1.431	9.394 ± 1.726	0.179	
T5	10.709 ± 1.140	10.490 ± 1.301	0.778	
T6	10.933 ± 1.072	10.649 ± 1.219	0.778	
Fz	10.568 ± 0.951	10.327 ± 1.024	0.423	
Cz	10.383 ± 0.867	10.291 ± 0.939	0.924	
Pz	10.744 ± 0.894	10.630 ± 1.023	0.423	

Table 2.5: Mean values of λ_1 of responding / non-responding patients before and after treatment.

Channel	Before	After	p-value	Sig.
mean	7.536 ± 0.394	7.585 ± 0.465	0.765	
std	0.401 ± 0.121	0.400 ± 0.134	0.917	
FP1	7.851 ± 0.588	7.841 ± 0.751	0.580	
FP2	7.921 ± 0.647	7.903 ± 0.553	0.580	
F3	7.614 ± 0.579	7.714 ± 0.634	0.765	
F4	7.640 ± 0.575	7.696 ± 0.591	0.408	
C3	7.399 ± 0.575	7.416 ± 0.659	0.989	
C4	7.303 ± 0.481	7.378 ± 0.615	0.765	
P3	7.247 ± 0.488	7.288 ± 0.552	0.580	
P4	7.338 ± 0.510	7.337 ± 0.543	0.765	
O1	7.554 ± 0.479	7.593 ± 0.571	0.917	
O2	7.539 ± 0.464	7.599 ± 0.560	0.765	
F7	7.662 ± 0.585	7.797 ± 0.601	0.269	
F8	7.717 ± 0.469	7.762 ± 0.574	0.408	
T3	7.694 ± 0.524	7.902 ± 0.636	0.269	
T4	7.682 ± 0.563	7.826 ± 0.522	0.100	
T5	7.606 ± 0.532	7.589 ± 0.477	0.765	
T6	7.578 ± 0.460	7.625 ± 0.485	0.765	
Fz	7.335 ± 0.522	7.340 ± 0.571	0.989	
Cz	7.321 ± 0.574	7.354 ± 0.532	0.917	
Pz	7.188 ± 0.451	7.162 ± 0.538	0.765	

Channel	Before	After	p-value	Sig.
mean	7.483 ± 0.432	7.648 ± 0.369	0.548	
std	0.366 ± 0.116	0.450 ± 0.162	0.048	**
FP1	7.709 ± 0.552	7.984 ± 0.658	0.149	
FP2	7.749 ± 0.582	7.909 ± 0.544	0.244	
F3	7.517 ± 0.576	7.717 ± 0.548	0.244	
F4	7.585 ± 0.621	7.886 ± 0.603	0.086	
C3	7.271 ± 0.579	7.427 ± 0.583	0.377	
C4	7.335 ± 0.561	7.435 ± 0.527	0.738	
P3	7.272 ± 0.474	7.392 ± 0.488	0.548	
P4	7.268 ± 0.457	7.417 ± 0.518	0.548	
O1	7.510 ± 0.693	7.530 ± 0.484	0.548	
O2	7.494 ± 0.533	7.623 ± 0.563	0.377	
F7	7.643 ± 0.419	7.874 ± 0.486	0.048	**
F8	7.651 ± 0.557	7.914 ± 0.554	0.012	***
T3	7.616 ± 0.582	8.032 ± 0.646	0.012	***
T4	7.687 ± 0.604	7.984 ± 0.648	0.086	
T5	7.517 ± 0.518	7.625 ± 0.441	0.548	
T6	7.488 ± 0.494	7.613 ± 0.493	0.377	
Fz	7.287 ± 0.509	7.359 ± 0.459	0.548	
Cz	7.380 ± 0.536	7.326 ± 0.490	0.902	
Pz	7.195 ± 0.498	7.262 ± 0.499	0.548	

Table 2.6: Mean values of d_2 of responding / non-respoding patients before and after treatment.

Channel	Before	After	p-value	Sig.
mean	0.768 ± 0.093	0.811 ± 0.107	0.086	
std	0.078 ± 0.039	0.093 ± 0.047	0.377	
FP1	0.816 ± 0.156	0.866 ± 0.178	0.548	
FP2	0.819 ± 0.158	0.866 ± 0.171	0.377	
F3	0.815 ± 0.128	0.872 ± 0.145	0.086	
F4	0.801 ± 0.126	0.868 ± 0.160	0.048	**
C3	0.796 ± 0.107	0.841 ± 0.123	0.149	
C4	0.780 ± 0.112	0.844 ± 0.129	0.012	***
P3	0.718 ± 0.075	0.744 ± 0.089	0.548	
P4	0.722 ± 0.089	0.751 ± 0.103	0.548	
O1	0.704 ± 0.074	0.738 ± 0.111	0.244	
O2	0.725 ± 0.094	0.746 ± 0.116	0.149	
F7	0.790 ± 0.166	0.837 ± 0.168	0.149	
F8	0.786 ± 0.138	0.845 ± 0.190	0.548	
T3	0.813 ± 0.163	0.894 ± 0.181	0.012	***
T4	0.828 ± 0.165	0.886 ± 0.162	0.048	**
T5	0.722 ± 0.078	0.762 ± 0.110	0.086	
T6	0.723 ± 0.094	0.752 ± 0.090	0.086	
Fz	0.758 ± 0.094	0.782 ± 0.109	0.244	
Cz	0.760 ± 0.082	0.783 ± 0.089	0.377	
Pz	0.724 ± 0.090	0.742 ± 0.103	0.902	

Channel	Before	After	p-value	Sig.
mean	0.757 ± 0.113	0.798 ± 0.137	0.676	
std	0.067 ± 0.044	0.095 ± 0.057	0.061	*
FP1	0.796 ± 0.137	0.850 ± 0.178	0.479	
FP2	0.799 ± 0.146	0.823 ± 0.162	0.975	
F3	0.783 ± 0.130	0.843 ± 0.163	0.479	
F4	0.782 ± 0.137	0.863 ± 0.169	0.111	
C3	0.780 ± 0.128	0.826 ± 0.157	0.193	
C4	0.774 ± 0.131	0.820 ± 0.153	0.193	
P3	0.713 ± 0.092	0.747 ± 0.134	0.676	
P4	0.713 ± 0.092	0.735 ± 0.130	0.975	
O1	0.715 ± 0.130	0.723 ± 0.159	0.975	
O2	0.716 ± 0.130	0.750 ± 0.201	0.975	
F7	0.794 ± 0.167	0.824 ± 0.186	0.314	
F8	0.784 ± 0.172	0.850 ± 0.207	0.314	
T3	0.802 ± 0.159	0.901 ± 0.211	0.031	**
T4	0.807 ± 0.176	0.887 ± 0.222	0.193	
T5	0.722 ± 0.123	0.747 ± 0.152	0.863	
T6	0.709 ± 0.127	0.721 ± 0.140	0.975	
Fz	0.731 ± 0.104	0.757 ± 0.124	0.314	
Cz	0.751 ± 0.093	0.757 ± 0.105	0.975	
Pz	0.713 ± 0.092	0.730 ± 0.121	0.863	

Table 2.7: Mean values of sample entropy of responding / non-responding patients before and after treatment.

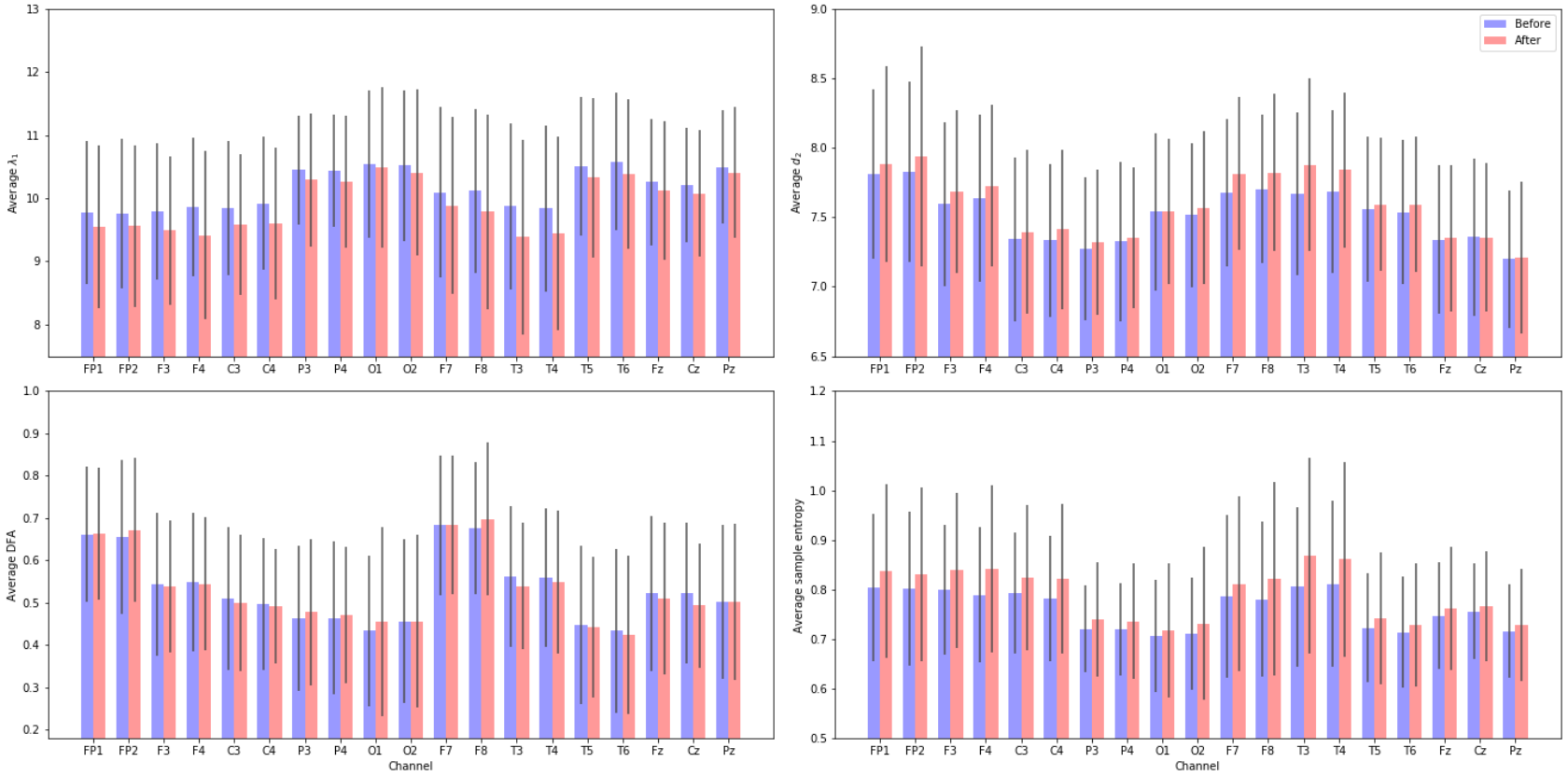


Figure 2.20: Values of individual measures computed before and after treatment.

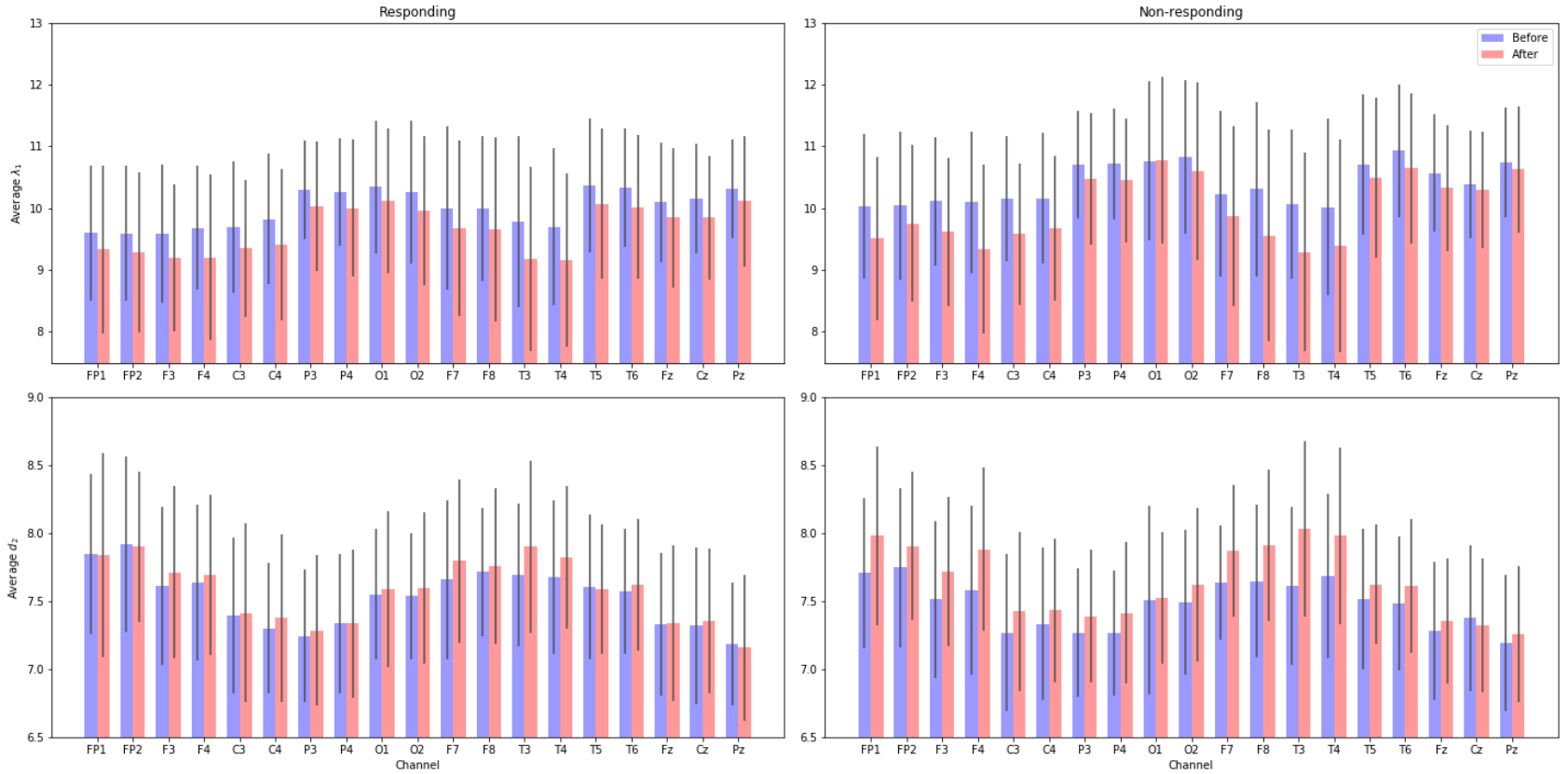


Figure 2.21: Comparison of mean values of largest Lyapunov exponent and correlation dimension between responders and non-responders computed using embedding dimension $m = 10$ and time delay $\tau = 3$.

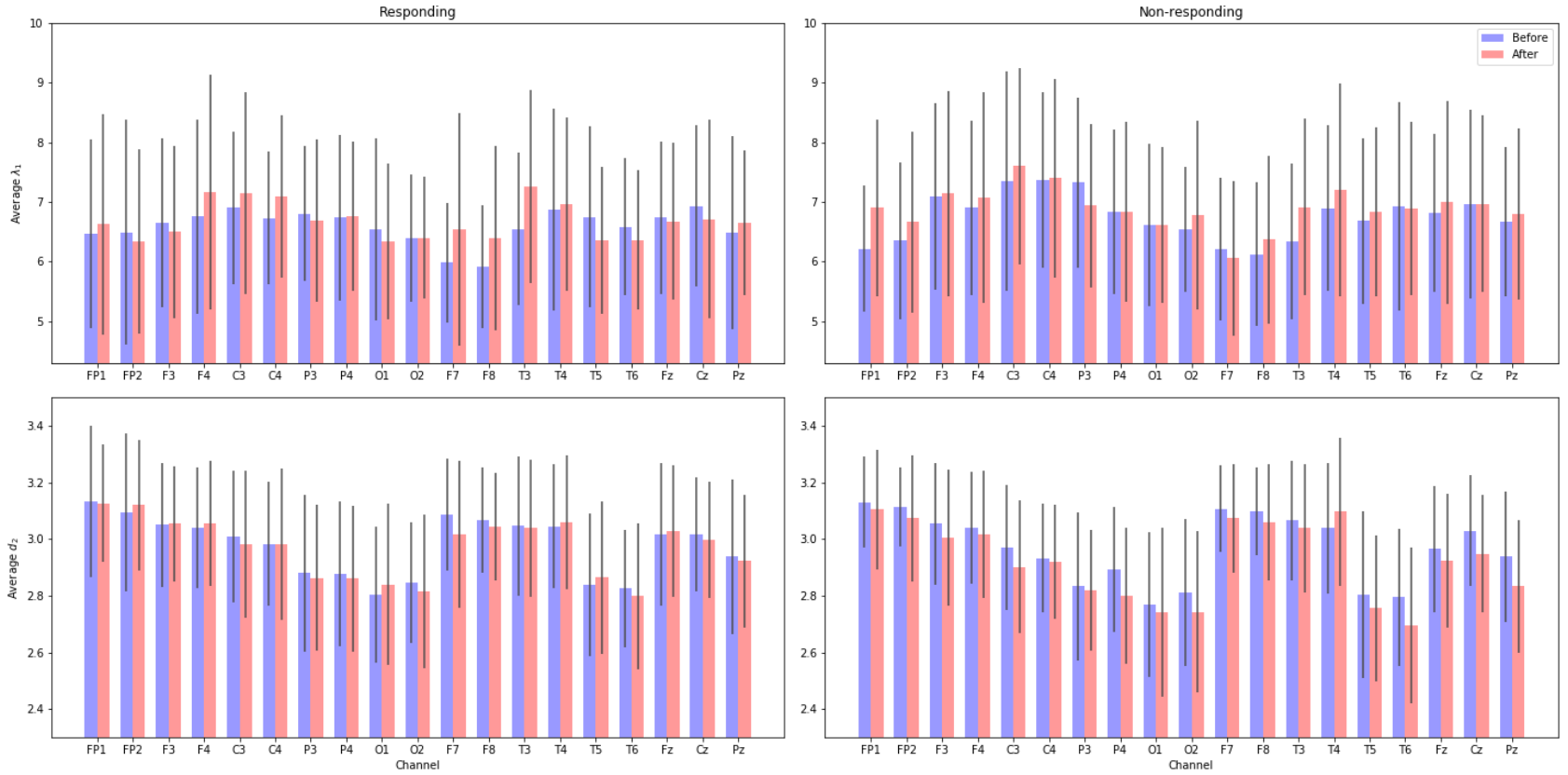


Figure 2.22: Comparison of mean values of largest Lyapunov exponent and correlation dimension between responders and non-responders computed using automatic procedure described in Section .

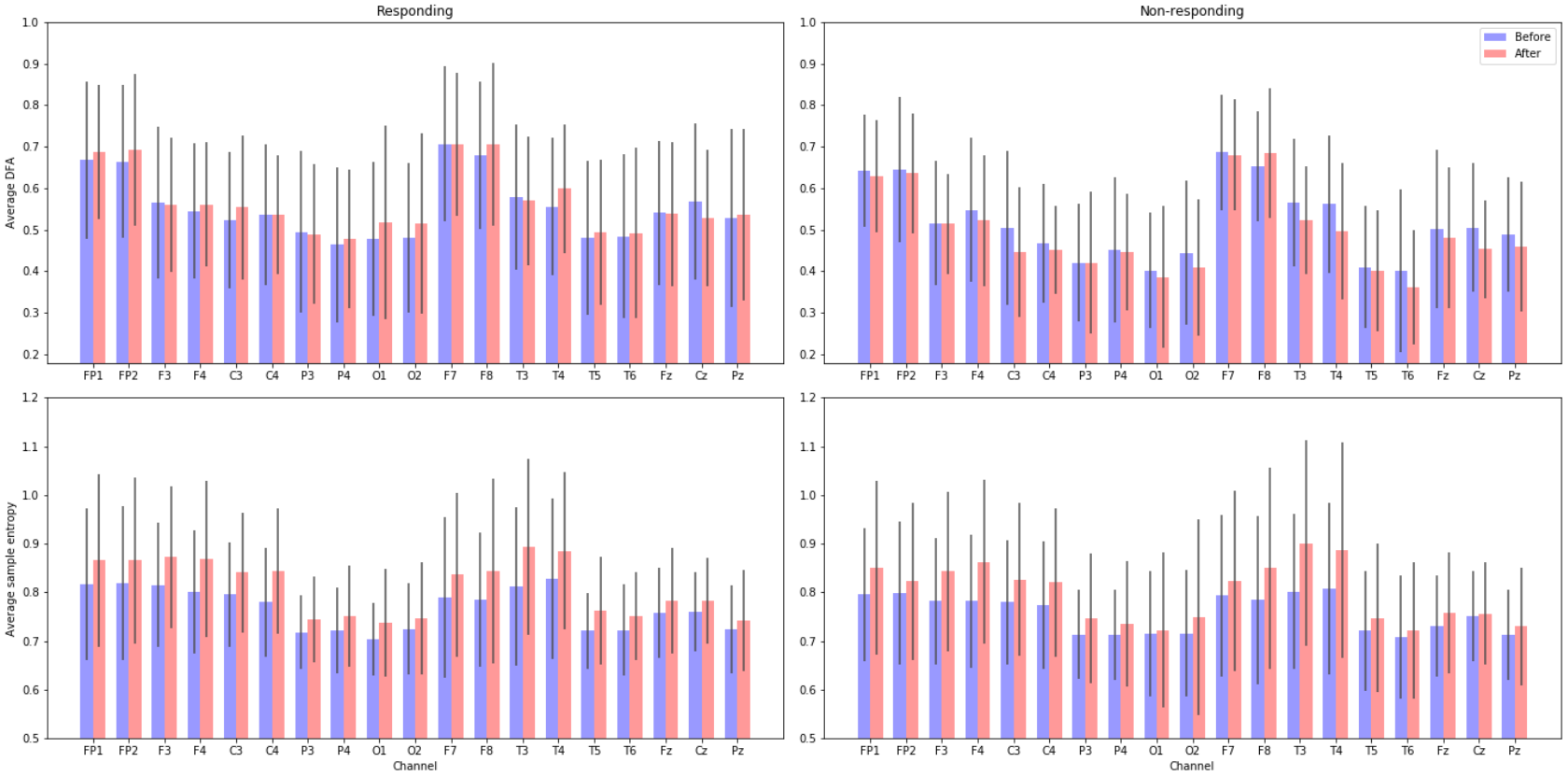


Figure 2.23: Comparison of mean values of computed detrended fluctuation analysis and sample entropy between responders and non-responders.

2.6.2 Low and high depression score

As mentioned in Section 2.1, studied dataset lacks symptom absent group. This makes the task of training a classifier for depression diagnosis inherently difficult. The patients, however, still vary in severity of their symptoms, which allows us to study correlation between symptom severity (which may, in turn, inform the task of finding a classifier). To this goal, we explored the differences in distributions of computed non-linear between groups of the "healthiest" and most depressed patients visually and using statistical tests, and in this section, we present some of the results.

With the goal of analyzing the differences between the lightest and most severe symptoms, we selected two classes of recordings for analysis in this section as follows. The first class, called *healthy*, of 50 recordings with reported depression score ≤ 16, and the second class, called *depressed*, of 50 recordings with depression score ≥ 28. We should recognize that including after treatment recordings does not control for possible effects of treatment not reflecting in the depression scores but reflecting in the signal, or the inverse. Indeed, all the healthy recordings were made after treatment, and most of the depressed recordings were made before treatment.

First, we looked at histograms of computed measures between the two groups. There were striking trends in the means of the two distributions in almost all channels for all measures except correlation dimension. Means of depressed recordings are typically shifted to the left of the mean of healthy recordings for all measures except for largest Lyapunov exponent, for which the means are shifted to the right. For correlation dimension, the distributions are similar. Figure 2.24, which shows the distributions of the largest Lyapunov exponents for both groups, exemplifies the differences. Another observation is that the distributions are, with exceptions, generally approximately normal.

Moreover, we investigated the differences in the distributions using Kolmogorov-Smirnov test.[2] Table 2.8 shows the results. The p-value cutoffs for significance ratings are 0.05, 0.01, 0.005. We may observe significant differences in most channels, with the strongest being in the occipital and temporal regions. Very significant differences seem to occur in the largest Lyapunov exponents corresponding to left and right temporal electrodes.

Furthermore, we inspected the correlations between the individual measures and correlation scores. Figure 2.25 shows visually clear negative correlation for DFA, and Figure 2.26 shows positive correlation for the largest Lyapunov exponent. Trends similar to the one observed for DFA were observed for all remaining features except for correlation dimension. Of course, these results are expected given the previous observations. However, the correlation becomes less significant when the classes are extended to include more recordings.

Is this true?

2.6.3 Low and high remission

Neurocorrelates of remission, or, in other words, positive response to a treatment, are interesting apart from the neurocorrelates of depression itself. Instead of indicating whether a treatment should be prescribed in the first place, the effects of various drugs on the brain may help in designing more individualized treatments, or in developments of new drugs, even for other conditions. However, as noted in Section 2.1, in our dataset, different kinds of treatments (including rTMS) are mixed for most patients, thus making the singular causes of any observed changes challenging. Nevertheless, we may still attempt to find discrepancies between the remitting and stagnant patients. If we assume that any prescribed treatment was beneficial, we may be able identify traits of patients who are difficult to treat. Indeed, medical literature recognizes entire categories of such patients. [46]

Hence, we assigned each patient a number called *change*, the ratio of score recorded during the second treatment to the score recorded during the first treatment. Mean change is 2.47, mode 1.66, standard

[2] Kruskal-Wallis test showed differences only for the largest Lyapunov exponent and Higuchi fractal dimension.

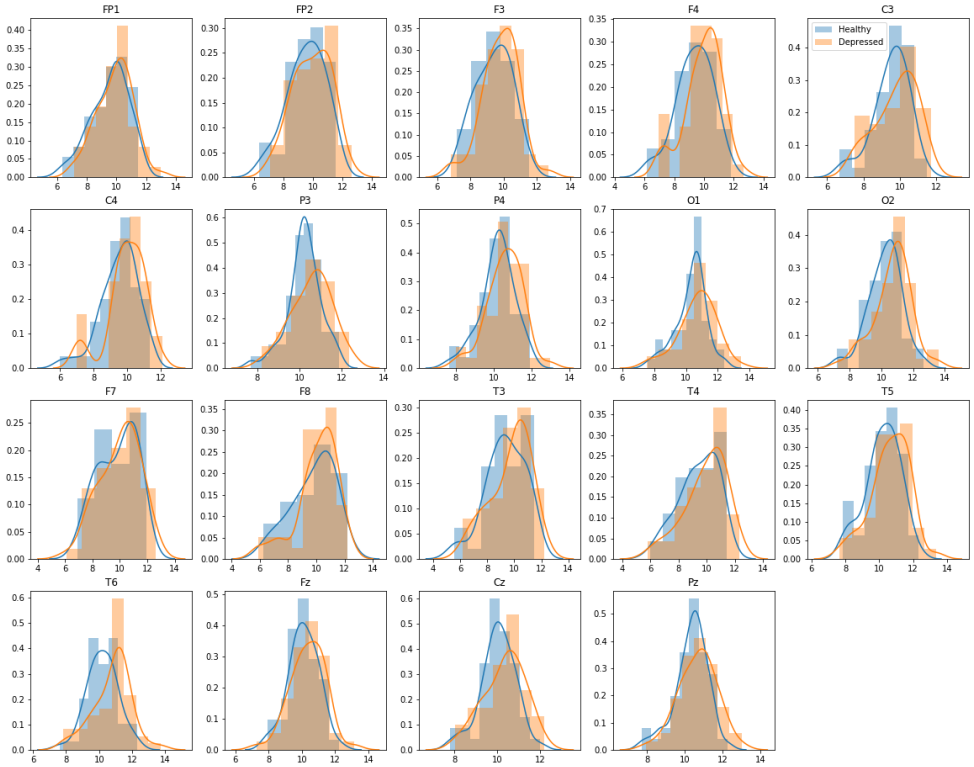


Figure 2.24: Distributions of the largest Lyapunov exponents between healthy and depressed patients. Most notable differences can be observed in the left and right temporal areas, T3 and T6. The distributions seem generally normal (however, this is not true for all measures).

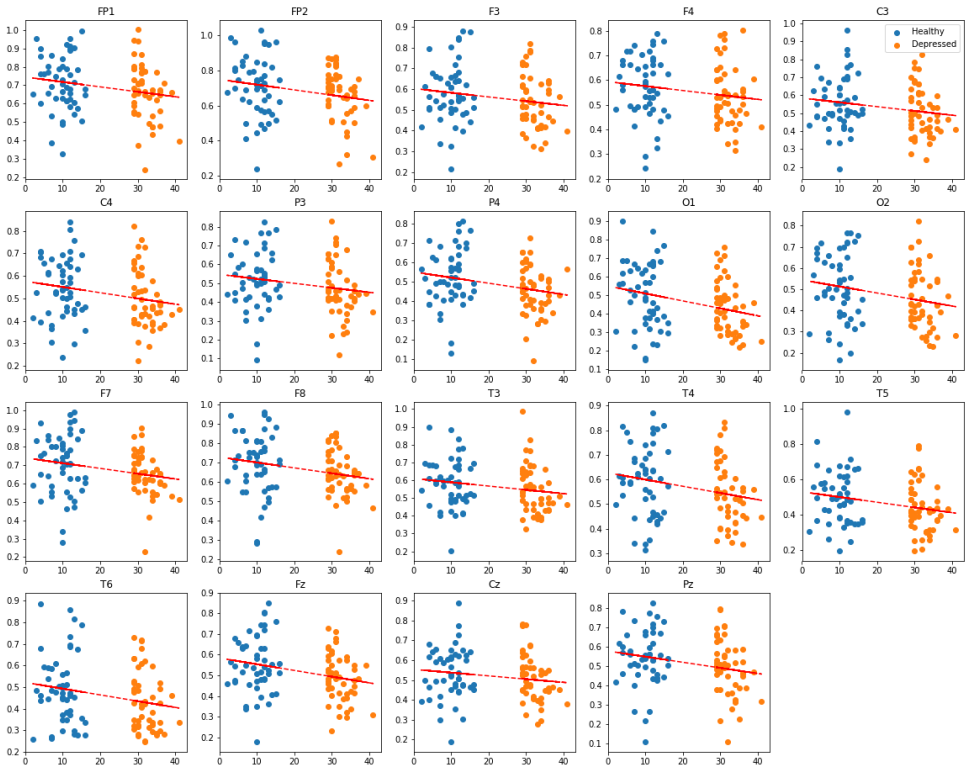


Figure 2.25: Trend of values of DFA as a function of depression score. The correlation is not significantly ($p < 0.05$) negative for F3, F4, P3, Cz. Similar trend is present in Hurst exponent, Higuchi fractal dimension and sample entropy.

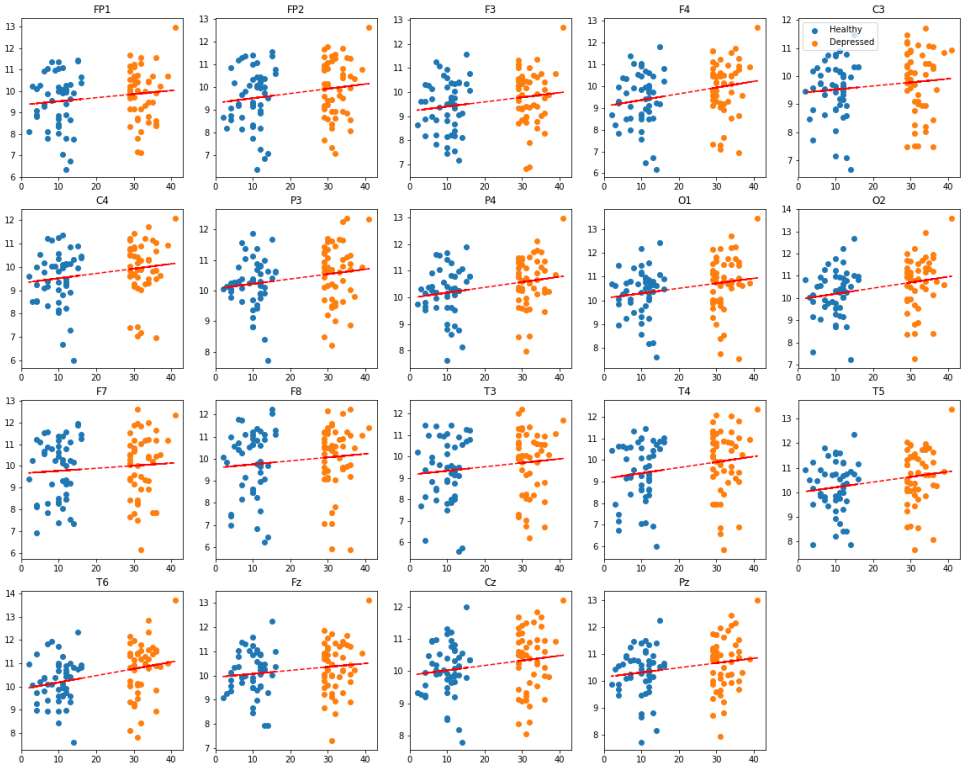


Figure 2.26: Trend of values of largest Lyapunov exponent as a function of depression score. The correlation is not significantly ($p < 0.05$) positive for all channels with exception of FP1, FP2, C3, F7, F8, T3.

Channel	Healthy	Depressed	p-value	Sig.
mean	0.576 ± 0.127	0.524 ± 0.104	0.095	
std	0.105 ± 0.027	0.110 ± 0.030	0.508	
FP1	0.712 ± 0.150	0.665 ± 0.152	0.508	
FP2	0.710 ± 0.168	0.663 ± 0.145	0.358	
F3	0.583 ± 0.141	0.538 ± 0.126	0.155	
F4	0.573 ± 0.126	0.538 ± 0.120	0.155	
C3	0.561 ± 0.146	0.507 ± 0.135	0.056	
C4	0.548 ± 0.135	0.498 ± 0.128	0.056	
P3	0.522 ± 0.148	0.473 ± 0.146	0.017	*
P4	0.526 ± 0.145	0.454 ± 0.124	0.095	
O1	0.502 ± 0.178	0.431 ± 0.141	0.056	
O2	0.509 ± 0.162	0.450 ± 0.141	0.032	*
F7	0.709 ± 0.162	0.652 ± 0.115	0.017	*
F8	0.696 ± 0.154	0.643 ± 0.117	0.009	**
T3	0.583 ± 0.134	0.548 ± 0.132	0.241	
T4	0.596 ± 0.141	0.544 ± 0.123	0.009	**
T5	0.496 ± 0.152	0.437 ± 0.137	0.032	*
T6	0.489 ± 0.160	0.433 ± 0.136	0.056	
Fz	0.554 ± 0.138	0.487 ± 0.113	0.095	
Cz	0.534 ± 0.127	0.504 ± 0.116	0.155	
Pz	0.547 ± 0.145	0.490 ± 0.146	0.017	*

(a) DFA

Channel	Healthy	Depressed	p-value	Sig.
mean	0.604 ± 0.091	0.574 ± 0.080	0.241	
std	0.068 ± 0.022	0.076 ± 0.025	0.241	
FP1	0.679 ± 0.088	0.654 ± 0.102	0.358	
FP2	0.674 ± 0.104	0.662 ± 0.101	0.841	
F3	0.609 ± 0.098	0.582 ± 0.084	0.241	
F4	0.604 ± 0.094	0.586 ± 0.089	0.241	
C3	0.596 ± 0.105	0.568 ± 0.093	0.155	
C4	0.587 ± 0.097	0.563 ± 0.093	0.358	
P3	0.567 ± 0.119	0.536 ± 0.115	0.508	
P4	0.573 ± 0.108	0.527 ± 0.106	0.155	
O1	0.549 ± 0.133	0.505 ± 0.116	0.155	
O2	0.561 ± 0.128	0.519 ± 0.112	0.032	*
F7	0.695 ± 0.105	0.667 ± 0.083	0.155	
F8	0.677 ± 0.103	0.658 ± 0.082	0.032	*
T3	0.611 ± 0.092	0.597 ± 0.087	0.508	
T4	0.625 ± 0.091	0.594 ± 0.087	0.009	**
T5	0.555 ± 0.114	0.509 ± 0.106	0.155	
T6	0.544 ± 0.115	0.502 ± 0.111	0.032	*
Fz	0.594 ± 0.096	0.553 ± 0.088	0.056	
Cz	0.594 ± 0.093	0.571 ± 0.084	0.241	
Pz	0.588 ± 0.114	0.554 ± 0.117	0.095	

(b) Hurst exponent

Channel	Healthy	Depressed	p-value	Sig.
mean	9.848 ± 0.947	10.236 ± 1.043	0.056	
std	0.681 ± 0.264	0.670 ± 0.291	0.841	
FP1	9.538 ± 1.257	9.888 ± 1.202	0.241	
FP2	9.551 ± 1.274	9.946 ± 1.275	0.358	
F3	9.413 ± 1.081	9.812 ± 1.097	0.241	
F4	9.363 ± 1.234	9.963 ± 1.254	0.032	*
C3	9.518 ± 1.010	9.803 ± 1.155	0.056	
C4	9.513 ± 1.131	9.956 ± 1.172	0.095	
P3	10.226 ± 0.806	10.569 ± 0.961	0.032	*
P4	10.200 ± 0.898	10.593 ± 0.941	0.095	
O1	10.321 ± 0.976	10.736 ± 1.201	0.056	
O2	10.240 ± 1.059	10.700 ± 1.205	0.032	*
F7	9.814 ± 1.401	9.988 ± 1.466	0.841	
F8	9.772 ± 1.528	10.061 ± 1.471	0.358	
T3	9.336 ± 1.454	9.731 ± 1.498	0.017	*
T4	9.408 ± 1.378	9.922 ± 1.520	0.155	
T5	10.216 ± 1.039	10.649 ± 1.142	0.155	
T6	10.200 ± 0.929	10.787 ± 1.157	0.001	***
Fz	10.097 ± 0.901	10.358 ± 1.040	0.358	
Cz	10.048 ± 0.816	10.339 ± 0.963	0.095	
Pz	10.343 ± 0.888	10.680 ± 1.016	0.032	*

(c) Largest Lyapunov exponent

Channel	Healthy	Depressed	p-value	Sig.
mean	0.797 ± 0.123	0.760 ± 0.125	0.056	
std	0.087 ± 0.046	0.076 ± 0.044	0.508	
FP1	0.843 ± 0.180	0.788 ± 0.154	0.241	
FP2	0.841 ± 0.179	0.790 ± 0.157	0.241	
F3	0.848 ± 0.149	0.799 ± 0.145	0.095	
F4	0.852 ± 0.171	0.792 ± 0.160	0.056	
C3	0.833 ± 0.141	0.801 ± 0.146	0.095	
C4	0.832 ± 0.151	0.791 ± 0.155	0.017	*
P3	0.740 ± 0.101	0.714 ± 0.102	0.095	
P4	0.742 ± 0.109	0.705 ± 0.105	0.095	
O1	0.729 ± 0.123	0.704 ± 0.128	0.358	
O2	0.739 ± 0.130	0.705 ± 0.126	0.032	*
F7	0.817 ± 0.174	0.799 ± 0.179	0.841	
F8	0.824 ± 0.199	0.796 ± 0.189	0.155	
T3	0.873 ± 0.195	0.834 ± 0.188	0.155	
T4	0.864 ± 0.179	0.816 ± 0.194	0.155	
T5	0.750 ± 0.120	0.715 ± 0.125	0.095	
T6	0.743 ± 0.109	0.704 ± 0.115	0.009	**
Fz	0.771 ± 0.116	0.739 ± 0.119	0.056	
Cz	0.775 ± 0.101	0.745 ± 0.108	0.155	
Pz	0.734 ± 0.109	0.701 ± 0.105	0.155	

(d) Sample entropy

Channel	Healthy	Depressed	p-value	Sig.
mean	1.408 ± 0.129	1.357 ± 0.131	0.017	*
std	0.093 ± 0.042	0.086 ± 0.042	0.241	
FP1	1.474 ± 0.186	1.406 ± 0.166	0.095	
FP2	1.468 ± 0.182	1.405 ± 0.176	0.056	
F3	1.465 ± 0.160	1.405 ± 0.158	0.017	*
F4	1.466 ± 0.180	1.398 ± 0.167	0.056	
C3	1.450 ± 0.135	1.406 ± 0.140	0.032	*
C4	1.446 ± 0.149	1.393 ± 0.139	0.032	*
P3	1.346 ± 0.109	1.305 ± 0.111	0.095	
P4	1.345 ± 0.119	1.295 ± 0.106	0.009	**
O1	1.322 ± 0.138	1.279 ± 0.136	0.155	
O2	1.330 ± 0.136	1.283 ± 0.141	0.095	
F7	1.450 ± 0.178	1.412 ± 0.181	0.508	
F8	1.462 ± 0.190	1.405 ± 0.183	0.095	
T3	1.474 ± 0.191	1.431 ± 0.184	0.508	
T4	1.468 ± 0.168	1.414 ± 0.192	0.241	
T5	1.333 ± 0.117	1.290 ± 0.127	0.095	
T6	1.327 ± 0.116	1.274 ± 0.121	0.032	*
Fz	1.381 ± 0.137	1.340 ± 0.147	0.095	
Cz	1.404 ± 0.115	1.356 ± 0.125	0.032	*
Pz	1.337 ± 0.118	1.295 ± 0.118	0.155	

(e) Higuchi fractal dimension

Channel	Healthy	Depressed	p-value	Sig.
mean	10.591 ± 0.879	10.816 ± 0.716	0.507	
std	0.664 ± 0.197	0.651 ± 0.154	0.377	
FP1	10.939 ± 1.125	11.222 ± 0.915	0.116	
FP2	11.015 ± 1.038	11.278 ± 0.978	0.257	
F3	10.616 ± 1.053	10.986 ± 0.911	0.108	
F4	10.614 ± 0.975	10.892 ± 0.944	0.250	
C3	10.147 ± 1.136	10.425 ± 0.864	0.035	*
C4	10.218 ± 1.120	10.504 ± 0.981	0.111	
P3	10.226 ± 1.053	10.596 ± 0.928	0.264	
P4	10.169 ± 0.973	10.420 ± 0.825	0.363	
O1	10.690 ± 1.007	11.014 ± 1.144	0.261	
O2	10.725 ± 1.088	10.863 ± 0.983	0.577	
F7	10.913 ± 0.962	10.948 ± 0.818	0.556	
F8	10.923 ± 0.892	11.087 ± 0.953	0.771	
T3	11.015 ± 1.113	10.995 ± 0.812	0.750	
T4	11.059 ± 1.002	11.109 ± 0.958	0.905	
T5	10.787 ± 1.048	10.831 ± 0.913	0.991	
T6	10.740 ± 0.946	10.998 ± 0.764	0.511	
Fz	10.151 ± 1.118	10.585 ± 0.906	0.182	
Cz	10.297 ± 1.258	10.451 ± 1.038	0.602	
Pz	9.991 ± 1.073	10.295 ± 0.891	0.363	

(f) Correlation dimension

Table 2.8: Comparison of mean values of measures computed for 50 healthy (depression score ≤ 16) and 50 depressed (depression score ≥ 28) patients.

deviation 3.145637. Most values range from 1 to 5, with a few outliers improving their symptoms 14 and 16-fold respectively. Only 9 patients stagnated exactly or slightly worsened their symptoms (change ≤ 1).

We performed Kolmogorov-Smirnov test to see the differences in the computed measures between the two groups in individual channels. The p-value cutoffs for significance ratings are 0.05, 0.01, 0.005. The two classes were selected to contain 50 least and 50 most remitting patients respectively, i.e. 50 patients with the lowest and 50 patients with the highest value of change. Mean change for such selected non-remitters is 1.19 ± 0.19 (range from 0.70 to 1.5), and 3.32 ± 2.37 (range from 2 to 16) for remitters.

Note that many of the patients classified as non-remitting actually improved their symptoms. In fact, symptoms of only 9 patients of the whole dataset worsened or stayed stagnant. No significant differences between before and after treatment recordings of the non-remitting patients, were found. This suggests considering the after treatment recordings for the non-remitting group to increase the number of strictly non-remitting patients. However, this would

Why not actually do that? I tried and it destroyed the differences.

Hence, we considered only before treament recordings in order to avoid the possible confounding effects of treatment. The most significant differences found were in Lyapunov exponent, especially in frontal, parietal, and right temporal areas. Aside from the largest Lyapunov exponent, Higuchi fractal dimension then also showed significant differences in frontal areas, and sample entropy in areas above corpus collosum. The results are shown in Table 2.9.

Of course, analyzing effectivity of treatment is difficult problem, and we realize the many limitations of this analysis. For example, many variables, including age, sex, starting depression score, behavioral changes occuring in the interim period and (again) the kind of treatment, were not accounted for.

2.7 Classification

We used two classifiers: logistic regression (LR) and support vector machine (SVM). One third of randomly selected samples was held out as a test set, the rest was used for training and cross validation. Feature selection was performed on LR with regularization strength 1 and SVM with regularization strenth 1 and linear kernel (i.e. $k(\boldsymbol{x_1}, \boldsymbol{x_2}) = \boldsymbol{x_1} \cdot \boldsymbol{x_2}$) using

- recursive feature elimination with 3-fold cross validation based on coefficients of the linear model,
- elimination of features with below-mean coefficients of the linear model,
- selection of 5 features with the highest χ^2 statistics between values of the feature and corresponding class,

Am I justified in using χ^2?

- genetic algorithm with 5-fold cross validation (scoring models based on ROC AUC, population size 80, 80 generations, crossover probability 0.8, mutation probability 0.2, and tournament size 5),
- manual selection of channels with significantly different means of corresponding features between the two considered classes, as reported by the Kolmogorov-Smirnov test.

Note that from the algorithmic techniques, genetic algorithm was by far the most effective. However, most of the best performing and thus reported classifiers were found by combination of the last two techniques, i.e. by applying genetic selection algorithm to the features marked as having diffening means between the two groups.

Evaluation was performed using 5-fold cross-validation. The best performing classifiers (based on accuracy, precision, recall, f-score, and number of features) were selected for each measure and for all

Channel	Remitting	Retaining	p-value	Sig.
mean	0.566 ± 0.122	0.554 ± 0.098	0.652	
std	0.106 ± 0.034	0.108 ± 0.026	0.652	
FP1	0.689 ± 0.154	0.697 ± 0.121	0.652	
FP2	0.690 ± 0.150	0.704 ± 0.127	0.822	
F3	0.579 ± 0.152	0.566 ± 0.129	0.946	
F4	0.567 ± 0.141	0.569 ± 0.112	0.946	
C3	0.548 ± 0.134	0.526 ± 0.124	0.652	
C4	0.541 ± 0.136	0.503 ± 0.110	0.139	
P3	0.515 ± 0.162	0.493 ± 0.126	0.480	
P4	0.499 ± 0.157	0.499 ± 0.123	0.822	
O1	0.492 ± 0.161	0.462 ± 0.136	0.480	
O2	0.497 ± 0.154	0.502 ± 0.156	0.480	
F7	0.687 ± 0.144	0.685 ± 0.092	0.480	
F8	0.688 ± 0.134	0.677 ± 0.103	0.652	
T3	0.600 ± 0.157	0.576 ± 0.115	0.220	
T4	0.576 ± 0.127	0.591 ± 0.127	0.652	
T5	0.486 ± 0.154	0.454 ± 0.130	0.480	
T6	0.485 ± 0.154	0.460 ± 0.148	0.139	
Fz	0.537 ± 0.129	0.514 ± 0.110	0.652	
Cz	0.544 ± 0.124	0.522 ± 0.100	0.083	
Pz	0.536 ± 0.165	0.534 ± 0.119	0.480	

(a) DFA

Channel	Remitting	Retaining	p-value	Sig.
mean	0.603 ± 0.093	0.593 ± 0.072	0.333	
std	0.071 ± 0.027	0.072 ± 0.018	0.220	
FP1	0.672 ± 0.103	0.669 ± 0.071	0.139	
FP2	0.680 ± 0.098	0.675 ± 0.076	0.652	
F3	0.610 ± 0.103	0.602 ± 0.086	0.652	
F4	0.606 ± 0.100	0.609 ± 0.080	0.946	
C3	0.597 ± 0.100	0.580 ± 0.091	0.652	
C4	0.594 ± 0.104	0.563 ± 0.087	0.048	*
P3	0.564 ± 0.129	0.550 ± 0.098	0.333	
P4	0.558 ± 0.127	0.555 ± 0.098	0.652	
O1	0.549 ± 0.126	0.527 ± 0.105	0.220	
O2	0.556 ± 0.128	0.555 ± 0.112	0.946	
F7	0.684 ± 0.097	0.685 ± 0.059	0.480	
F8	0.684 ± 0.086	0.678 ± 0.070	0.652	
T3	0.626 ± 0.104	0.617 ± 0.077	0.220	
T4	0.622 ± 0.088	0.623 ± 0.080	0.652	
T5	0.546 ± 0.123	0.522 ± 0.097	0.048	*
T6	0.547 ± 0.122	0.519 ± 0.116	0.220	
Fz	0.587 ± 0.099	0.577 ± 0.080	0.480	
Cz	0.600 ± 0.093	0.579 ± 0.080	0.139	
Pz	0.582 ± 0.130	0.583 ± 0.085	0.480	

(b) Hurst exponent

Channel	Remitting	Retaining	p-value	Sig.
mean	10.123 ± 0.766	10.458 ± 0.952	0.014	*
std	0.628 ± 0.217	0.604 ± 0.252	0.480	
FP1	9.762 ± 1.057	10.052 ± 1.110	0.220	
FP2	9.740 ± 1.132	10.088 ± 1.150	0.139	
F3	9.688 ± 0.917	10.156 ± 1.000	0.001	***
F4	9.802 ± 0.941	10.241 ± 1.117	0.014	*
C3	9.821 ± 0.931	10.115 ± 1.004	0.083	
C4	9.921 ± 0.909	10.217 ± 1.076	0.003	***
P3	10.410 ± 0.638	10.700 ± 0.884	0.048	*
P4	10.423 ± 0.700	10.754 ± 0.918	0.014	*
O1	10.609 ± 0.769	10.815 ± 1.218	0.139	
O2	10.443 ± 0.943	10.863 ± 1.167	0.083	
F7	10.092 ± 1.289	10.386 ± 1.358	0.333	
F8	10.045 ± 1.116	10.459 ± 1.329	0.007	**
T3	9.885 ± 1.324	10.126 ± 1.221	0.652	
T4	9.777 ± 1.164	10.176 ± 1.418	0.048	*
T5	10.531 ± 0.824	10.773 ± 1.114	0.014	*
T6	10.500 ± 0.860	10.981 ± 1.086	0.007	**
Fz	10.238 ± 0.801	10.570 ± 0.913	0.048	*
Cz	10.213 ± 0.770	10.447 ± 0.835	0.220	
Pz	10.447 ± 0.702	10.775 ± 0.876	0.139	

(c) Largest Lyapunov exponent

Channel	Remitting	Retaining	p-value	Sig.
mean	0.764 ± 0.102	0.749 ± 0.106	0.026	*
std	0.075 ± 0.038	0.064 ± 0.041	0.003	***
FP1	0.806 ± 0.154	0.789 ± 0.129	0.652	
FP2	0.810 ± 0.162	0.787 ± 0.136	0.652	
F3	0.809 ± 0.128	0.776 ± 0.122	0.026	*
F4	0.799 ± 0.134	0.772 ± 0.130	0.139	
C3	0.797 ± 0.116	0.781 ± 0.122	0.048	*
C4	0.785 ± 0.122	0.767 ± 0.126	0.048	*
P3	0.718 ± 0.084	0.711 ± 0.086	0.333	
P4	0.720 ± 0.096	0.709 ± 0.089	0.220	
O1	0.699 ± 0.080	0.705 ± 0.119	0.333	
O2	0.718 ± 0.102	0.702 ± 0.119	0.220	
F7	0.781 ± 0.163	0.775 ± 0.156	0.333	
F8	0.783 ± 0.146	0.766 ± 0.159	0.139	
T3	0.810 ± 0.169	0.792 ± 0.151	0.333	
T4	0.818 ± 0.164	0.795 ± 0.168	0.220	
T5	0.719 ± 0.085	0.715 ± 0.116	0.220	
T6	0.722 ± 0.104	0.702 ± 0.117	0.139	
Fz	0.753 ± 0.100	0.729 ± 0.097	0.048	*
Cz	0.758 ± 0.090	0.744 ± 0.090	0.480	
Pz	0.720 ± 0.098	0.705 ± 0.087	0.333	

(d) Sample entropy

Channel	Remitting	Retaining	p-value	Sig.
mean	1.378 ± 0.113	1.348 ± 0.113	0.083	
std	0.080 ± 0.036	0.074 ± 0.037	0.333	
FP1	1.442 ± 0.165	1.400 ± 0.138	0.333	
FP2	1.445 ± 0.169	1.405 ± 0.143	0.139	
F3	1.431 ± 0.138	1.381 ± 0.133	0.026	*
F4	1.420 ± 0.143	1.380 ± 0.139	0.083	
C3	1.413 ± 0.115	1.385 ± 0.120	0.048	*
C4	1.403 ± 0.124	1.376 ± 0.126	0.220	
P3	1.328 ± 0.101	1.304 ± 0.104	0.083	
P4	1.322 ± 0.103	1.302 ± 0.103	0.048	*
O1	1.298 ± 0.106	1.284 ± 0.139	0.333	
O2	1.311 ± 0.125	1.281 ± 0.128	0.220	
F7	1.418 ± 0.159	1.385 ± 0.152	0.083	
F8	1.419 ± 0.144	1.387 ± 0.160	0.014	*
T3	1.421 ± 0.169	1.388 ± 0.143	0.480	
T4	1.422 ± 0.166	1.398 ± 0.165	0.652	
T5	1.312 ± 0.099	1.290 ± 0.118	0.083	
T6	1.308 ± 0.110	1.278 ± 0.130	0.139	
Fz	1.367 ± 0.123	1.330 ± 0.119	0.026	*
Cz	1.383 ± 0.106	1.359 ± 0.103	0.333	
Pz	1.328 ± 0.111	1.301 ± 0.106	0.083	

(e) Higuchi fractal dimension

Channel	Remitting	Retaining	p-value	Sig.
mean	10.546 ± 0.755	10.696 ± 0.848	0.639	
std	0.649 ± 0.191	0.653 ± 0.162	0.466	
FP1	10.900 ± 0.951	11.006 ± 0.834	0.639	
FP2	11.005 ± 0.967	11.043 ± 1.010	0.941	
F3	10.659 ± 1.000	10.648 ± 1.049	0.994	
F4	10.486 ± 0.796	10.785 ± 1.056	0.210	
C3	10.058 ± 0.918	10.308 ± 1.145	0.131	
C4	10.100 ± 0.990	10.412 ± 1.180	0.466	
P3	10.272 ± 0.987	10.402 ± 0.993	0.639	
P4	10.245 ± 1.036	10.341 ± 0.903	0.994	
O1	10.677 ± 0.933	10.968 ± 1.251	0.466	
O2	10.728 ± 0.875	10.789 ± 1.021	0.941	
F7	10.859 ± 0.915	10.968 ± 0.948	0.812	
F8	10.864 ± 0.869	11.088 ± 0.952	0.210	
T3	10.760 ± 0.920	10.769 ± 0.914	0.941	
T4	10.837 ± 1.070	10.992 ± 1.040	0.210	
T5	10.719 ± 0.884	10.743 ± 1.041	0.466	
T6	10.739 ± 0.939	10.847 ± 0.993	0.994	
Fz	10.301 ± 0.979	10.402 ± 1.023	0.639	
Cz	10.197 ± 1.114	10.472 ± 1.167	0.210	
Pz	9.964 ± 0.917	10.249 ± 1.132	0.131	

(f) Correlation dimension

Table 2.9: Comparison of mean values of measures computed from recordings obtained during the first session for 50 remitting (patients responding positively to treatment) patients and 50 patients retaining (or worsening their) symptoms.

measures by varying the maximum number of features considered by the genetic algorithm from 3 to the 1/10 of the corresponding training set size. Then, a brute force grid search with 5-fold cross validation was performed on each classifier to select

- the optimal regularization stength, and norm for LR, and
- the optimal regularization strength and kernel type (linear, polynomial, or radial basis function with coefficients $\gamma = 1/n_f$, where n_f is the number of selected features) for SVM.

Am I justified in changing the kernel type?

This resulted in slight improvement in accuracy, and correspondingly slight bias of the reported classifiers.

2.7.1 Depression

The recordings were separated into two classes as follows:

Healthy : 50 recordings with associated depression score at most 16.

Depressed : 50 recordings with associated depression score at least 28.

The results are shown in Table 2.10. The best performing classifiers in this section were SVMs. The largest Lyapunov exponent was the most predictive out of all considered non-linear measures, both achieving the highest accuracy out of the single-measure classifiers (0.72 ± 0.04), and being one of the measures in majority of the best performing combined-measure classifiers. It was followed, perhaps surprisingly considering the results obtained in Section 2.6.2, by correlation dimension (0.71 ± 0.05). Although the accuracy of the remaining classifiers, whose features were obtained using the Kolmogorov-Smirnov test from Section 2.6.2, was slightly lower (with higher variance), they are also simpler in terms of the number of selected channels.

All the channels in the combined-measure classifiers were found using the genetic algorithm, as described in the opening to this section. The best overall accuracy was achieved by combination of the largest Lyapunov exponent and sample entropy (0.75 ± 0.10). However, second to it was a combination of the largest Lyapunov exponent and correlation dimension, which has lower variance (0.74 ± 0.04). Other measures performing well together with the largest Lyapunov exponent are the Hurst exponent, and sample entropy together with DFA. The best combination not inluding the largest Lyapunov exponent is correlation dimension and Higuchi fractal dimension.

There seems to be little consistency in the selected features of the same measures across classifiers. A possible explanation is that different measures complement themselves in such a way that different channels are relevant when classification is performed based on single measure as opposed to a combination of measures.

2.7.2 Remission

Let us remind the reader of the definition of change from Section 2.6.3 as the ratio of the depression score reported on the second session (after administration of drugs) to the depression score reported on the first session (before administration of drugs). The recordings were separated into two classes as follows:

Retaining : 50 recordings made before administration of drugs with the lowest change.

Remitting : 50 recordings made before administration of drugs with the highest change.

The results can be seen in Table 2.11.

Measure	Classifier	Accuracy	Precision	Recall	F-score	Channels
LLE, SE	SVM (lin.)	0.75 ± 0.10	0.77 ± 0.09	0.75 ± 0.10	0.75 ± 0.10	*LLE*: C4, T3, T6, Pz *SE*: C3, P4
LLE, CD	SVM (lin.)	0.74 ± 0.04	0.76 ± 0.04	0.74 ± 0.04	0.74 ± 0.05	*LLE*: F3, F7, T6 *CD*: O1, O2, T5
LLE, HE	SVM (lin.)	0.73 ± 0.06	0.74 ± 0.06	0.73 ± 0.06	0.73 ± 0.06	*LLE*: P3, T3, T6, Pz *HE*: C3, T3
LLE, SE, DFA	SVM (lin.)	0.73 ± 0.09	0.74 ± 0.10	0.73 ± 0.09	0.73 ± 0.09	*LLE*: T6, Fz *SE*: T6 *DFA*: P4
CD, HD	LR	0.73 ± 0.10	0.74 ± 0.11	0.73 ± 0.10	0.73 ± 0.10	*CD*: F3, Fz *HD*: P3, Cz
LLE	SVM (lin.)	0.72 ± 0.04	0.73 ± 0.04	0.72 ± 0.04	0.72 ± 0.04	T3, T5, T6, Pz
CD	SVM (lin.)	0.71 ± 0.05	0.72 ± 0.05	0.71 ± 0.05	0.71 ± 0.05	F3, C4, P3, F8, T5, T6, Fz, Cz
SE	LR	0.68 ± 0.12	0.69 ± 0.12	0.68 ± 0.12	0.68 ± 0.12	C4, O2, T6
HD	SVM (rbf)	0.67 ± 0.11	0.67 ± 0.12	0.67 ± 0.11	0.67 ± 0.11	C3, C4, P4, T6, Cz
DFA	LR	0.67 ± 0.16	0.68 ± 0.17	0.67 ± 0.16	0.67 ± 0.16	F8, O2
HE	LR	0.67 ± 0.17	0.68 ± 0.18	0.67 ± 0.17	0.67 ± 0.17	O2, T4

Table 2.10: Evaluation of depression classification. The two classes consist of 50 / 50 recordings with the smallest / highest associated depression score out of recordings performed both before and after administration of drugs.

Measure	Classifier	Accuracy	Precision	Recall	F-score	Channels
LLE, SE	SVM (lin.)	0.75 ± 0.10	0.77 ± 0.09	0.75 ± 0.10	0.75 ± 0.10	*LLE*: FP2, F3, O1, T4, T6 *SE*: F3, C3, T6
LLE, CD	SVM (lin.)	0.75 ± 0.11	0.76 ± 0.11	0.75 ± 0.11	0.75 ± 0.11	*LLE*: F3, O2, T5, T6 *CD*: FP2, F4, O2
LLE	LR	0.71 ± 0.08	0.73 ± 0.08	0.71 ± 0.08	0.70 ± 0.09	F3, F4, T5, T6
CD	LR	0.67 ± 0.09	0.70 ± 0.11	0.67 ± 0.09	0.65 ± 0.10	F3, F4, O2, Pz
HD	LR	0.66 ± 0.05	0.72 ± 0.08	0.66 ± 0.05	0.64 ± 0.05	F3, F8
SE	LR	0.66 ± 0.09	0.66 ± 0.09	0.66 ± 0.09	0.65 ± 0.10	FP1, F3, P3, Cz
DFA	SVM (lin.)	0.64 ± 0.15	0.65 ± 0.15	0.64 ± 0.15	0.63 ± 0.15	T3, T4, Cz
HE	SVM (rbf)	0.63 ± 0.09	0.64 ± 0.10	0.63 ± 0.09	0.62 ± 0.09	C3, T6

Table 2.11: Evaluation of remission classification. Only recordings obtained before drug administration were considered. The two classes consist of the 50 patients with the highest and least improvement in depression score after the drug administration (as measured by ratio of the two depression scores).

Chapter 3

Feature extraction approach

3.1 Convolutional Neural Networks

3.1.1 Mathematical background

Definition 13. *Let I be an image function, K a kernel. A (discrete)* ***convolution*** *of I and K is a functional defined as*

$$(I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n)\,. \tag{3.1}$$

Note that some machine learning libraries (such as Tensorflow) implement **cross-correlation** instead of convolution, but preserving the term convolution for the operation. Cross-correlation corresponds to convolution with kernel rotated by 90 degrees:

$$(I * K)(i, j) = \sum_m \sum_n I(m, n)K(i + m, j + n)\,. \tag{3.2}$$

Unlike convolution, cross-correlation is not commutative, but this property is not required for neural network applications.

Definition 14. *Let f be arbitrary function, and $\mathcal{D}$ its degradation operator. We say f is* ***invariant*** *under $\mathcal{D}$ if*

$$\mathcal{D}(f) \equiv f\,. \tag{3.3}$$

For the following, the reader needs to understand the term **equivariance**.

Definition 15 ([71])**.** *Let G be a group and X, Y its G-sets. Then $F : X \to Y$ is called an* ***equivariant function*** *if*

$$F(g(x)) = g(F(x)) \tag{3.4}$$

for all G actions g and $x \in X$.

For our purposes, we can view G as a group of transformations, and then equivariance as a commutative property of a function with regards to the transformations. In other words, computing the function and then applying the transformation has the same effect as applying the transformation and then computing the function.

Algorithm 1 Gradient descent algorithm.

1: Initialize random $x_0 \in D(f)$
2: $n \leftarrow 0$
3: step_size $\leftarrow 1$
4: **while** step_size < threshold and $n <$ iters_limit **do**
5: $\quad x_{n+1} = x_n - \epsilon \nabla_{x_n} f$
6: $\quad$ step_size $\leftarrow |x_{n+1} - x_n|$
7: $\quad n \leftarrow n + 1$
8: **end while**

Gradient descent is a first order iterative method of finding an extremum a differentiable function $f : \mathbb{R} \to \mathbb{R}^n$, $f \in C^1$, based on continually moving a point in its domain in the direction of negative of its gradient at that point, until the absolute value of the gradient (or the step size) is below a certain threshold. See Algorithm 1.

Add description of stochastic gradient descent, Nesterov and momentum?

3.1.2 History

The classical approach to image pattern recognition consists of the following stages:

preprocessing: supressing unwanted distortions and noise, enhancement beneficient for further processing,

object segmetation: separating disparate objects from the background,

feature extraction: gathering relevant information about the properties of the objects, removing irrelevant variations,

classification: categorizing segmented objects based on obtained features into classes.

The preprocessing step may require additional assumptions about the data or further processing, which are potentially too restrictive or too broad. Getting around this limitation requires dealing with complications such as high dimensionality of the input (number of pixels) and desirability of invariance towards a number of allowable distortions and geometrical transformations.

Artificial neural networks in combination with gradient-based learning are one possible solution to the problem. By gradually optimizing a set of weights based on a training data set using a differentiable error function, they provide a framework for learning a suitable set of assumptions automatically from the data.

One of the oldest neural network architectures, fully connected multi-layer perceptron (FC-MLP), can be used for image pattern recognition. However, it has the following drawbacks:

parameter explosion: the number of parameters of such network is exponential in the number of layers, increasing the capacity of the network and therefore need for more data,

no invariance: no invariance even with respect to common geometrical transformation such as translation, rotation and scaling,

ignoring input topology: natural images exhibit strong local structure and high correlation between intensities of neighboring pixels, but FC-MLPs are unstructed - inputs can be presented in any order.

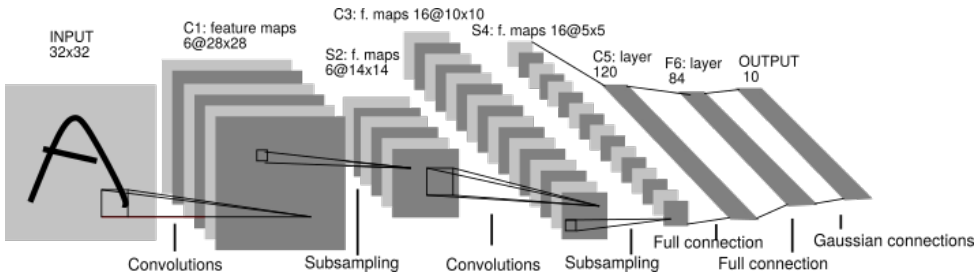


Figure 3.1: LeNet-5 architecture [54].

Although the main idea dates back to 1980, when K. Fukushima introduced neocognitron [29], the back-propagation algorithm was not known at the time. The first convolutional architecture successfuly applied on an image pattern recognition problem by attempting to solve the aforementioned problems, dubbed LeNet-5, was proposed in 1998 by Y. LeCun, L. Bottou, Y. Bengio and P. Haffner [53].

3.1.3 Description

Bearing resemblence to visual processing in biological organisms [1], LeNet-5 proposed the following design principles to enforce *shift, scale and distortion invariance*: [54]

local receptive fields: each neuron in a layer receives input from a small neighborhood in the previous layer,

shared weights: each layer is composed of neurons organized in planes within which each neuron have the same weight vector (feature map),

spatial subsampling: adding a subsampling layers, which reduce the resolution of the previous layer by averaging or taking the maximal value of neighboring pixels in the previous layer.

3.1.3.1 Local receptive fields

Local receptive fields enable the network to synthesize filters that produce strong response to elementary salient features in the early layers (such as lines, edges and corners in a visual input, and their equivalents in other modalities), and then learn to combine them in the subsequent layers to produce higher-order feature detectors.

For a visual explanation of the concept of receptive field, see Figure 3.2. The locality of of those receptive fields implies sparser connectivity, and hence more efficient computations in comparison with fully connected neural networks. A fully connected neural network with no hidden layers with m inputs and n outputs has $m \times n$ weight parameters, and the correspoding feed forward pass (matrix multiplication) is of $O(m \times n)$ time complexity per input. If the number of connections per output unit is limited to $k < m$, the achieved runtime is $O(k \times n)$, where k is usally in practice several orders of magintude smaller than m. [31]

In shallow neural networks, locality of receptive fields implies locality of "influence" of each input unit on the output. In deep neural networks, on the other hand, units in the deeper layers can be indirectly

[1] As early as in 1968, D. H. Hubel and T.N. Wiesel discovered that some cells (called simple cells) in cat's primary visual cortex (V1) with small receptive fields (shared by neighboring neurons) are sensitive to straight lines and edges of light of particular orientation, and other cells (called complex cells) with larger receptive fields further in the visual cortex also respond to straight lines and edges, but with invariance to translation [41].

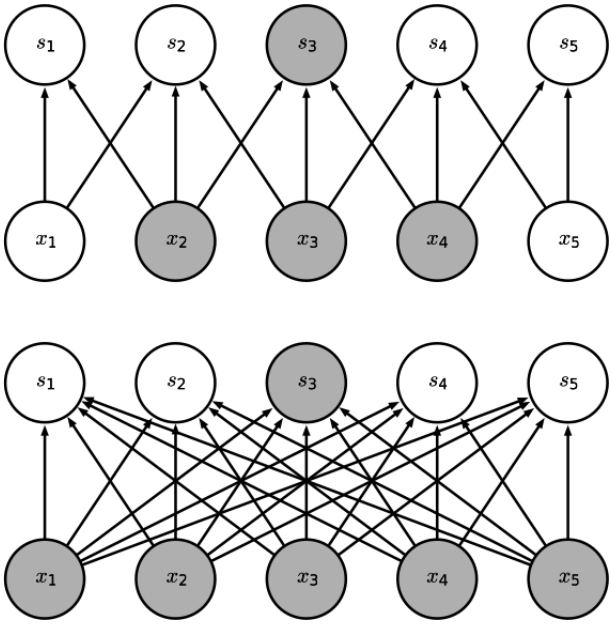


Figure 3.2: Receptive field. [31]

connected to some or all units of the input, thus enabling them to achieve aforementioned effect of combining more complex features from simpler ones.

3.1.3.2 Shared weights

With *shared weights*, neural units in a layer with differing receptive fields have the same feature map and the same feature detecting operation (convolution with feature map kernel followed by additive bias and a application of a non-linear function) is performed on differing parts of the image (see Figure 3.3). A single convolutional layer is composed of multiple feature detecting planes.

Shared weights principle exploits the fact that in natural images, a function of small number of neighboring pixels can be useful in multiple parts of the image. For example, an edge detector can be used accross the entire image to detect edges in the first layer, an object detector can then be used to detect presence of edges in particular arrangements in the next layer, etc.

Although it does not reduce the time complexity of the feedforward pass, it does reduce the memory requirements. If the kernel size is k, m the number of inputs, n the number of outputs, the number of parameters per layer is k instead of $m \times n$ (per feature detecting plane) in a fully connected case. Since k is usually in practice several orders of magnitude smaller than m, and usually m and n are comparable in size, the memory savings are highly significant. [31]

One of the drawbacks of classical CNNs is that although convolution in combination with weight sharing causes layer output to be equivariant to translation of the input, this is not the case for scaling and rotation. Moreover, equivariance to input may not be always desirable. Consider a case of face detection, where all training and test images are centered. Then, the relative positions of individual features are important, and it may be favorable to fix feature detectors (and thus weights) to certain locations in the image.

3.1.3.3 Pooling

The final output activations of a convolutional layer are computed in subsequent stages:

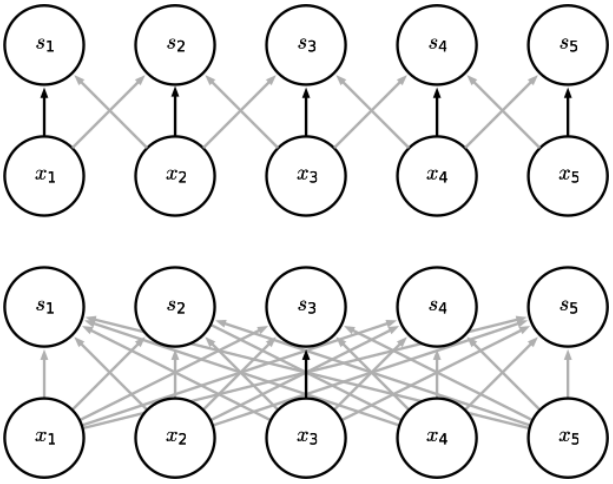


Figure 3.3: Shared weights. [31]

1. linear unit activations are computed via the convolution operation,

2. a non-linear activation function is applied to the activations,

3. a spatial subsampling (pooling) operation is applied.

The rationale behind applying a non-linearity is it makes the network capable of modelling non-linear functions. Common activation functions include rectified linear $\max(0, x)$, sigmoid $\frac{1}{1+\exp(-x)}$, hyperbolic tangent tanh, and many others. They have varying properties making them useful in different situations. We will not explore them further here.

Pooling operation splits the neural units into sets of multiple adjacent activations and computes a summary statistic, such as the maximum element (max pooling) or the average (average pooling), per such set and outputs the result. If the stride between the sets is greater than one, the spatial dimension of output is decreased relative to input (subsampling).

The purpose of spatial subsampling is to ensure scale and distortion invariance[2] by reducing the precision at which a feature is encoded in a feature map by reducing its resolution - when scale and distortion invariance is assumed, the exact location of a feature becomes less important and is allowed to exhibit slight positional variance - roughly speaking, an "approximate" translation invariance.

Although the combination of convolution and pooling performs well in many practical situations, it has multiple drawbacks. For example, the learned representations are not rotation invariant and thus, to mitigate this, the capacity of the network has to be increased and the training dataset must be enhanced to contain examples of rotated features, often extending the amount of data necessary and training time. A number of alternative approaches were suggested in the litarature.[3] For another example of a limitation, see Figure 3.4.

[2]Whether it achieves this goal has been famously doubted by Geoffrey Hinton: "The pooling operation used in convolutional neural networks is a big mistake and the fact that it works so well is a disaster." []

[3]For instance, Hinton's *CapsNet*, described e.g. in [82], is an attempt to transform the manifold of images of similar shape (which is highly non-linear in the space of pixel intensities) to a space where it is globally linear by the way of using so called capsules instead of traditional convolutional layers.

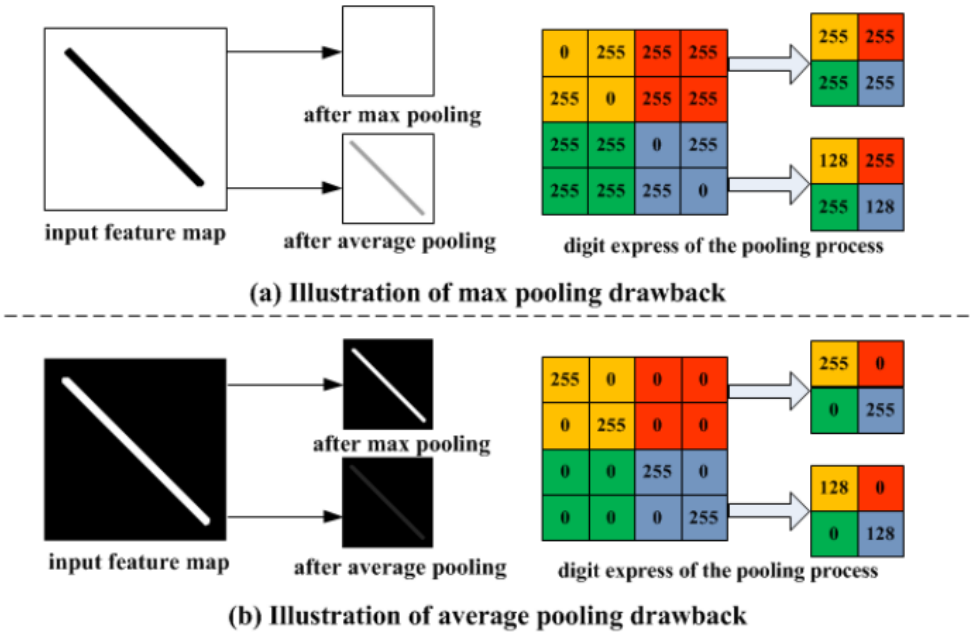


Figure 3.4: Examples of drawbacks of the pooling operation. Max pooling discards all except the maximum element, and valuable information may thus be lost. Average pooling considers all the values, and the information about their contrast is reduced. Moreover, extreme values may have undesired effects on the result. [101]

3.2 Common Spatial Patterns (CSP)

The method of Common Spatial Patterns (CSP) was originally proposed for people with impeded motor control (e.g. disabled people) in context of brain-computer interfaces, and thus most studies focus on its use in classification of motion performed or visualized by the subject. In our study, we will apply convolutional neural network architectures inspired by FBCSP for depression diagnosis and prediction of future remission of the disease.

As mentioned repeatedly in the previous text, the task of finding patterns in EEG signal associated with particular mind state or motor action present us with numerous challenges. CSP, and in FBCSP (see Section 3.2.2) in particular, are methods devised in attempt to overcome mainly two of them. Firstly, information about different temporally overlapping brain activities is conveyed in parallel in multiple frequency bands. For example, resting wakeful state comprises distinct idle rhythms over different cortical areas (such as α-rhythm characteristic of idling visual cortex in the occipital area), which are overlapping with μ-rhythms produced in sensorimotor areas both during imagined and performed movement. Secondly, the spatial origin of those signals is important for associating them with said mind states or motor actions. For example, different parts of the sensorimotor cortex over the central sulcus map directly to movements of distinct bodyparts. This is further complicated by the fact that EEG apparatus has inherently low spatial resolution due to small number of electrodes and poor volume conduction.

Spatial filtering, then, is process of addressing this second challenge by accentuating signals from some areas, while attenuating others. And CSP analysis is a data-driven approach of achieving this by mutually maximizing the variance of spatially filtered signal associated with one activity, while minimizing the variance of filtered signal associated with other activity, thus making the signals independent (as Gaussian random processes). [13] In the following section, we will explain the process in detail.

3.2.1 CSP analysis

Let C be the number of channels, and $\mathbf{x}(t) \in \mathbb{R}^C$ be a band-passed, de-meaned and scaled multichannel EEG recording. CSP analysis yields a projection of $\mathbf{x}(t)$ of the signal from the original signal space to $\mathbf{x}_{\text{CSP}}(t) \in \mathbb{R}^C$ by finding a matrix $W \in \mathbb{R}^{C\times C}$, where

$$\mathbf{x}_{\text{CSP}}(t) = W^T\mathbf{x}(t).$$

Each column vector of W is referred to as spatial filter. Thus, CSP decomposes the original signals into additive subcomponents, column vectors of $A := (W^{-1})^T$, referred to as spatial patterns, giving name to the technique.

The matrix W is found under optimization criteria, which we will describe in the following text. Firstly, let $\Sigma^{(+)} \in \mathbb{R}^C$ and $\Sigma^{(-)} \in \mathbb{R}^C$ be estimates of the inter-channel covariance matrices, corresponding to signals recorded in the two conditions c we aim to distinguish, + and −:

$$\Sigma^{(c)} = \frac{1}{|I_c|}\sum_{i\in I_c} X_i X_i^T, \qquad c \in \{+,-\},$$

where I_c is the set of time indeces matching the two conditions.[4] Since variance of band-pass filtered is the power present in the frequency band, the diagonal elements of $\Sigma^{(c)}$ represent the fraction of the total band power in each channel, and the off-diagonal elements represent the fractional covariance. [51] CSP then performs simultaneous decomposition

$$\begin{aligned} W^T\Sigma^{(+)}W &= \Lambda^{(+)}, \\ W^T\Sigma^{(-)}W &= \Lambda^{(-)}, \qquad \Lambda^{(c)} \text{ diagonal} \end{aligned}$$

under the condition that $\Lambda^{(+)} + \Lambda^{(+)} = I$, which is equivalent to solving the generalized eigenvalue problem

$$\Sigma^{(+)}\mathbf{w} = \lambda\Sigma^{(-)}\mathbf{w}$$

for generalized eigenvectors $\mathbf{w}$ and their eigenvalues λ. The resulting eigenvectors $\mathbf{w}_j$, $j \in \{1,\dots,C\}$ then are the column vectors of W, and corresponding eigenvalues $\lambda_j^{(c)} = \mathbf{w}_j^T\Sigma^{(c)}\mathbf{w}_j$ are the diagonal elements of $\Lambda^{(c)}$. Then, $\lambda_j = \lambda_j^{(+)}/\lambda_j^{(-)}$, and $\lambda_j^{(+)} + \lambda_j^{(-)} = 1$. This means that high variance the direction of $\mathbf{w}_j$ of signal in class + results in small variance in signal in class −, and vice versa (see Figure 3.5). [13]

This method, although loosely based on PCA, is better suited for supervised classification, since, unlike PCA, it is guaranteed to find components which are responsible for the maximum differences in variance between the two classes. These eigenvectors are an orthonormal set which spans $\mathbb{R}^C$, and are optimal for the amount of variance they account for in the least squares sense. [51]

3.2.2 Filter Bank Common Spatial Patterns (FBCSP)

Although CSP usually yields good performance when the signals have been filtered in frequency range carefully tuned for the particular subject and classification problem at hand, its performance rapidly decreases when measurements are either unfiltered or filtered in inappropriate frequency range. [7] Thus, an improvement has been suggested, called Filter Bank Spatial Patterns (FBCSP). It comprises of four stages: frequency filtering, spatial filtering, feature selection and classification (see Figure 3.6). In Stage

[4] Here we suppose that two separate events happened during a single recording to simplify notation. This is not strictly necessary.

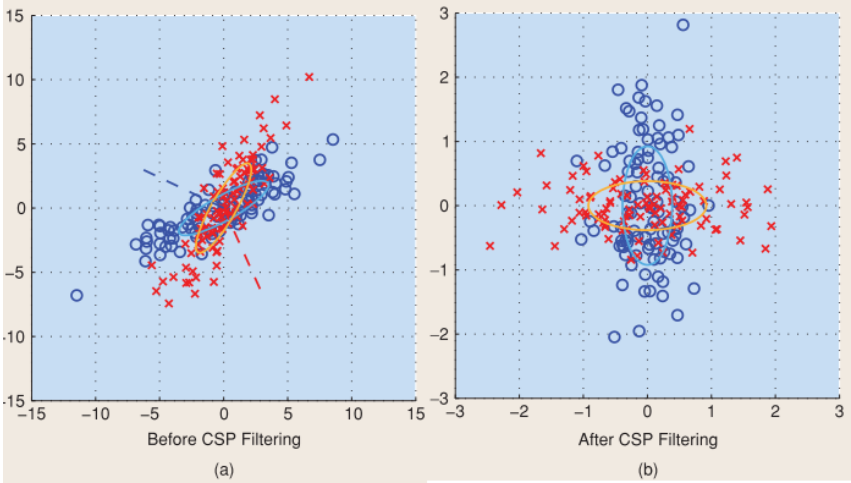


Figure 3.5: CSP

1, multiple band-pass filters are applied to split the signal into distanct filter banks. Then, in Stage 2, spatial filters are found for each of the respective filter banks using CSP analysis, as described in Section 3.2.1. These filters characterize features present in the signal specific to the corresponding frequency band. In Stage 3, a feature selection algorithm is employed to select the most discriminative features of all the filters found in previous step. Finally, in Stage 4, a classification algorithm uses the selected features for classiying the input signal into a class.

Using the fact that magnitudes of the CSP eigenvalues are proportional to the amount of variance explained by corresponding signal in the direction corresponding to the eigenvector, the CSP algorithm in Stage 2 is slightly modified to to order the eigenvectors according to the magnitued of their eigenvalues, and only the first and last m filtered signals are selected for further classification. This means selecting only the first and last m rows from the matrix $X_{\text{CSP}=W^T X}$, yielding a matrix $Z \in (2m) \times T$ with row vectors Z_j, $j = 1, \ldots, 2m$. The final feature vector $\mathbf{f}$ is composed as logarithm of the contribution of variance of each row vector to the total variance as follows: [7]

$$f_j = \log\left(\frac{\text{var}(Z_j)}{\sum_{i=1}^{2m} \text{var}(Z_i)}\right). \tag{3.5}$$

3.3 Experiment

3.3.1 Input representation

Before applying any machine learning technique to the classification problem at hand, the question of optimal input representation needs to be answered. To this end, multiple factors need to be considered. Firstly, what is the dimensionality of the input relative to the resulting number of samples, and does it allow construction of sufficiently complex architectures compared to complexity of the classification problem? Secondly, does each input sample contain enough information to perform successful classification? Thirdly, is the input representation appropriate for the kind of data, i.e. does it help or hinder successful classification? In our case, for all the methods considered, answer to the first question is a function of recording slice length used to generate the input. Answering the remaining questions,

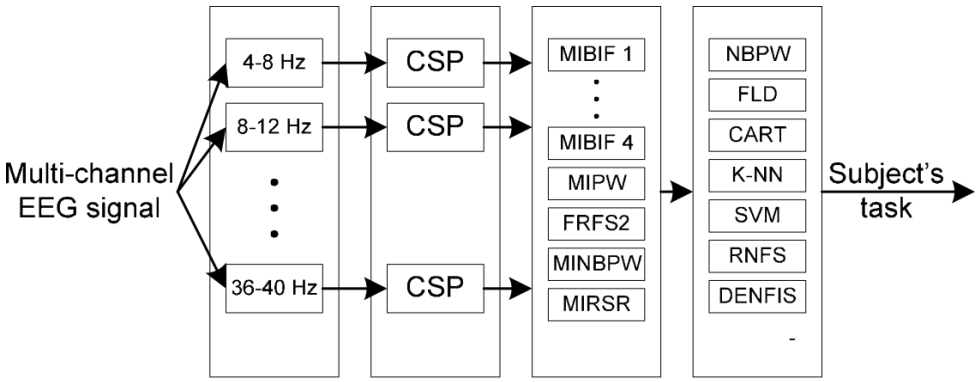


Figure 3.6: FBCSP
[7]

however, is difficult without prior experiments on similar datasets. For this reason, research on applying known techniques to new problems is useful.

Since well designed neural networks are characteristically able to learn feature maps given enough data, one obvious possibility is to use raw data. This approach has multiple benefits. First one, as we will see, is relatively low dimensionality. Global, and local features, no prior bias. Our first choice, then, was then to select

As mentioned in Chapter 1, brain is a non-linear dynamical system. Another possibility, therefore, is to represent the input data as recurrence plots, mentioned in Section 1.3.1. The obvious drawbacks are redundancy due to symmetry and high dimensionality. On the other hand, recurrence plots are known to capture properties of the system which are difficult to obtain using other methods in some cases. [21] Morever, they have already been applied with success to classification of physical activites using convolutional neural networks [30], and even some qualitative differences in recurrence plots have been observed between depressed and healthy patients [2]. We have discussed more applications in 1.7.

For our computation of recurrence matrices, we used the Chebyshev norm. Chebyshev norm has multiple benefits , and we observed more subtle patterns on matrices computed using Chebyshev norm as opposed to those computed using Euclidean L_2 norm. For comparison, see Figure 3.7.

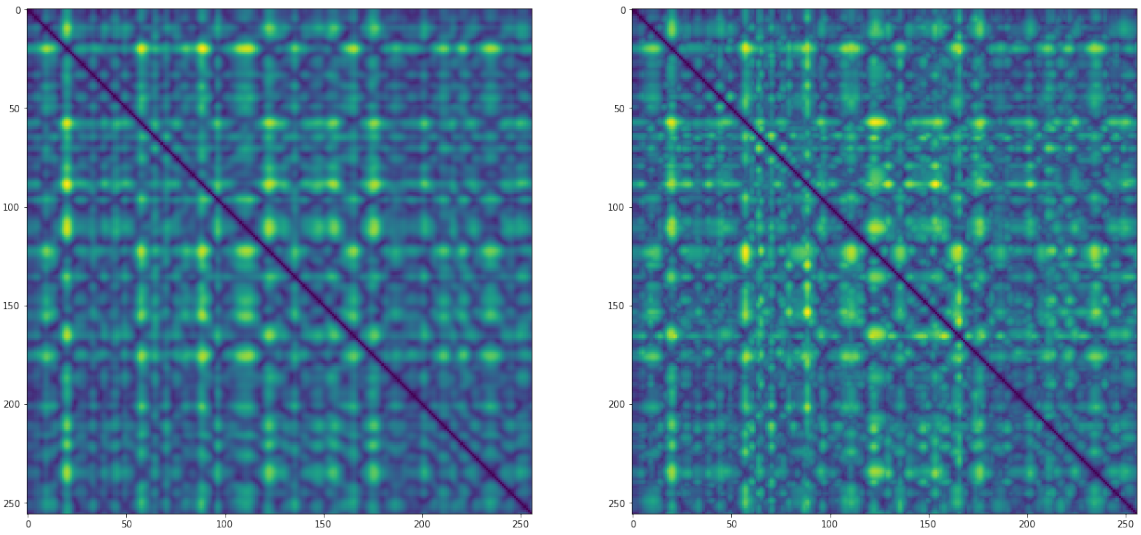
mention some

Our last method of input representation is inspired by success of Gramian Angular Fields (GAFs) for sequence classification [97] using convolutional neural networks. To obtain GAF matrix from a scalar time series $x_1, x_2, \ldots, x_N$, one first scales the time series into interval $(-1, 1)$, and then each value x_i of the time series is converted into complex number with mode and radius given as

$$\phi_i = \arccos(x_i),$$
$$r_i = i/N.$$

This way, temporal dependencies are conserved through the radius. Then, instead of scalar product, an operation $\oplus$ is defined as

$$x_i \oplus x_j = \cos(\phi_i + \phi_j)$$



(a) Euclidean norm

(b) Chebyshev norm

Figure 3.7: Recurrence plots computed using different norms. We can see that the figure on the right hand side has sligthly crisper patterns.

and a quasi-Gram NxN matrix G is computed as

$$G = \begin{pmatrix} \cos(\phi_1 + \phi_1) & \cos(\phi_1 + \phi_2) & \dots & \cos(\phi_1 + \phi_N) \\ \cos(\phi_2 + \phi_1) & \cos(\phi_2 + \phi_2) & \dots & \cos(\phi_2 + \phi_N) \\ \vdots & \vdots & \dots & \vdots \\ \cos(\phi_N + \phi_1) & \cos(\phi_N + \phi_2) & \dots & \cos(\phi_N + \phi_N) \end{pmatrix}.$$

Since GAFs which are defined only for single channel time-series, we modify this approach, use spatial embedding, thus obtaining a multi-channel time series $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$, and then compute cosine similarities between each pair of those vectors as

$$G = \begin{pmatrix} \frac{\mathbf{x}_1 \cdot \mathbf{x}_1}{\|\mathbf{x}_1\| \|\mathbf{x}_1\|} & \frac{\mathbf{x}_1 \cdot \mathbf{x}_2}{\|\mathbf{x}_1\| \|\mathbf{x}_2\|} & \dots & \frac{\mathbf{x}_1 \cdot \mathbf{x}_N}{\|\mathbf{x}_1\| \|\mathbf{x}_N\|} \\ \frac{\mathbf{x}_2 \cdot \mathbf{x}_1}{\|\mathbf{x}_2\| \|\mathbf{x}_1\|} & \frac{\mathbf{x}_2 \cdot \mathbf{x}_2}{\|\mathbf{x}_2\| \|\mathbf{x}_2\|} & \dots & \frac{\mathbf{x}_2 \cdot \mathbf{x}_N}{\|\mathbf{x}_2\| \|\mathbf{x}_N\|} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\mathbf{x}_N \cdot \mathbf{x}_1}{\|\mathbf{x}_N\| \|\mathbf{x}_1\|} & \frac{\mathbf{x}_N \cdot \mathbf{x}_2}{\|\mathbf{x}_N\| \|\mathbf{x}_2\|} & \dots & \frac{\mathbf{x}_N \cdot \mathbf{x}_N}{\|\mathbf{x}_N\| \|\mathbf{x}_N\|} \end{pmatrix}.$$

Since both recurrence plot and cosine similarity matrix are symmetric, we applied the following procedure for computing them. For a given subseries length l_s, we computed recurrence plot of subseries $2l_s$, and considered only lower left quadrant. This way, the inherent redundancy was completely removed, while preserving some of the information - the lower left quadrant contains relationships (i.e. distances or similarities) between time states occuring in the previous subseries of length l_s.

Another possibility is to learn on flattened scalp images of topographical distributions of different band powers. However, as explained in [85], this presents two main challanges. As we have verified in the previous chapter (see Section 2.5.7), the relevant variance is probably spatially global in nature, and not hierarchically compositional to make use of CNN. On the other hand, the temporal patterns are more likely to be hierarchically compositional.

3.3.2 Used architectures

Our choice of CNN architectures was heavily inspired by, and almost identical to, those used in [85]. These architectures, and in particular the second one (in order of description below), were designed by the authors to be analogous to the FBCSP pipeline described in detail in Section 3.2.1.

The first architecture, called *deep* (see Figure 3.8), is more generic of the two architectures used, bearing resemblence to the architectures which proved successful in traditional computer vision tasks. It consists of four convolutional blocks with batch normalization ($\epsilon = 10^{-5}$, momentum = 0.1) and ELU non-linearity, followed by max pooling and dropout ($p = 0.5$). The batch normalization was applied before the activation function. The convolution was performed only along the temporal dimension, with kernel size $(1, 3)$, stride 1. The pooling operation was also performed only along the temporal dimension, with kernel size $(1, 3)$, stride 3. For image input, we used traditional 2D convolution with kernel size $(3, 3)$. The first convolutional layer is an exception - to explicitly seperate the linear transformation into combination of temporal and spatial convolution, this layer is split into two layers with no activation function in between. First, a temporal convolution with kernel size $(1, 10)$ is performed, followed by a spatial convolution across all channels with kernel size $(19, 1)$. Note that the first operation can be seen as analogue of band-pass filtering, and the second as spatial filtering, as performed by CSP algorithm, with the difference that the filters are "constructed" by gradient descent. Batch normalization and pooling operations are also performed as described above.

The second architecture, called *shallow* (see Figure 3.9), is more specialized, tailored to mimic the transformations performed by the FBCSP pipeline. The first and only convolutional layer is split in the same way as in the deep architecture described above, and batch normalization is also applied before the activation. However, squaring non-linearity was used as activation function for the layer instead, followed by average pooling. This can be seen as approximation of computing mean power. Moreover, following the recommendations mentioned in [85], larger kernel size $(1, 25)$ is used for the temporal filtering in this network. Then, logarithm non-linearity is applied, analogous to the mean log-variance computation in FBCSP, see (3.5). One of the advantages of this architecture over FBCSP that it can learn the structure of temporal changes in the representation of "band-powers".

In both deep and shallow architectures, the classification is performed by classification layer with 2D convolution of kernel size of the last layer, 2 filters, and logistic activation to produce probability estimate for each class. For optimization, we used stochastic gradient descent with Nesterov momentum 0.99, decay 10^{-5}, learning rate 0.01 for batch size 128 (which we used for raw data), and learning rate 0.001 for batch size 64 (which we used for image data due to hardware limitations). This last change was made because lower batch size leads to more updates per epoch.

We also attempted different configurations: increasing or decreasing the number of layers in the deep network, increasing or decreasing the kernel sizes, ReLU activation functions, and Adam or rmsprop optimizers. However, we found any of these changes leading to degradation in performance.

For image-encoded data, i.e. recurrence plots and cosine similarities, we also tried the architecture (along with the same hyperparameters) used in [30], which resulted in overall accuracy of 0.942 and 0.804 recall on classification task of 6 activies using recurrence plots and CNNs (as mentioned in Section 1.7), evaluated using 10-fold cross validation on over 10 000 samples. However, we were unable to replicate the result. This may be because of the difference in input image sizes, number of used input channels (the authors had only 4 electrodes available, and used all of them as input channels, whereas we used spatial embedding).

Moreover, we also evaluated multiple simpler architectures. The best performing (both on image-encoded and raw data) was a VGG-like model with 3 convolution-pooling modules (convolution kernels $(3, 3)$, pooling kernels $(2, 2)$, ReLU activation functions) with 8, 16 and 16 filters respectively. These were followed by dropout ($p = 0.5$), and fully connected sigmoid classification layer. This model, optimized by rmsprop, achieved 73% accuracy on stand-out test set on raw data, and below 60% on the image-encoded data. All attempts of modifying capacity and regularization, i.e. adding batch normalization, adding or removing layers, increasing or decreasing the number of filters, as well as adding weight normalization or changing the optimizer, lead to deterioration of performance.

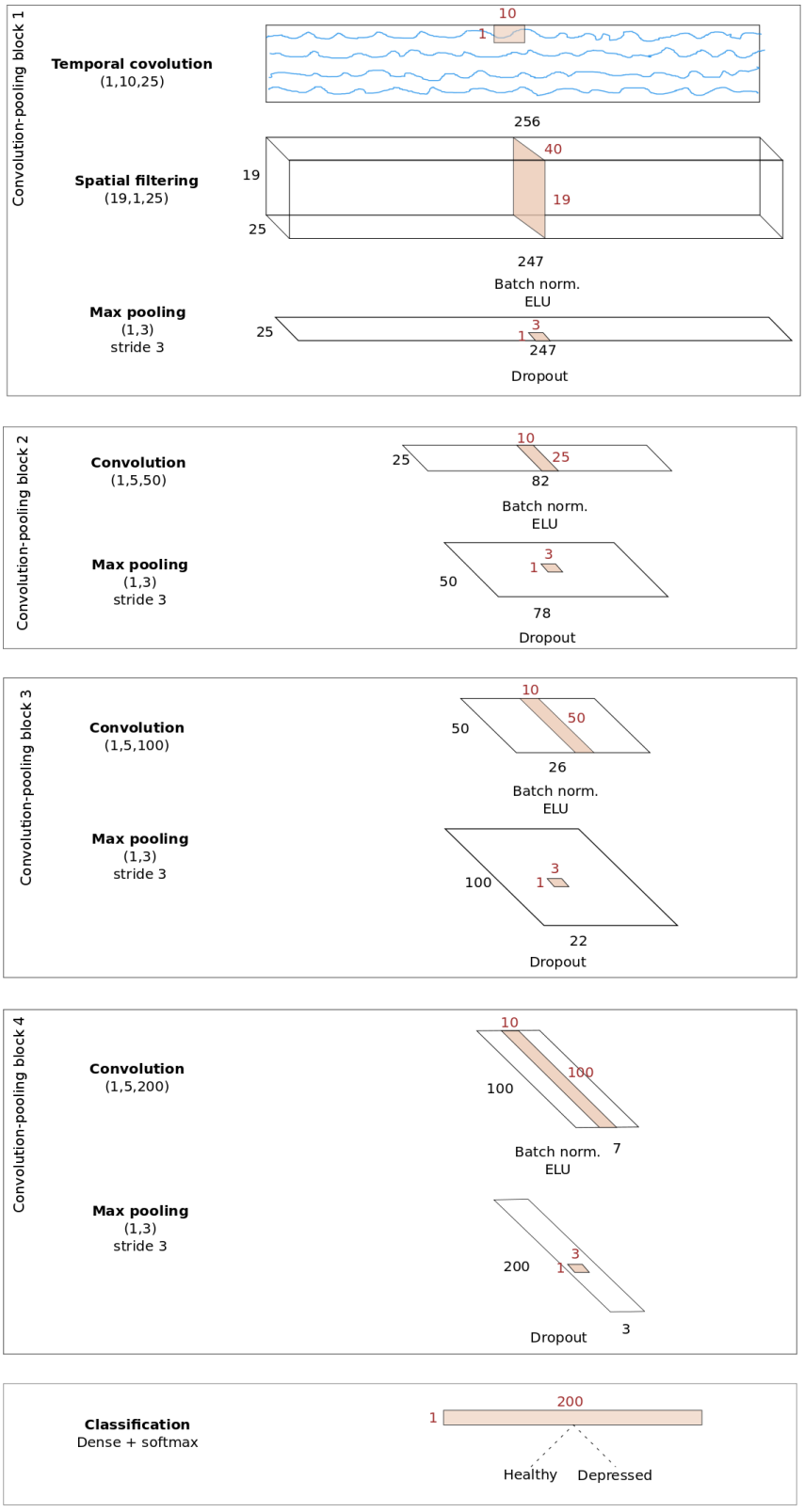


Figure 3.8: Deep architecture for evaluation on the raw data. For evaluation on image-encoded data, the kernel sizes were changed - see text for details.

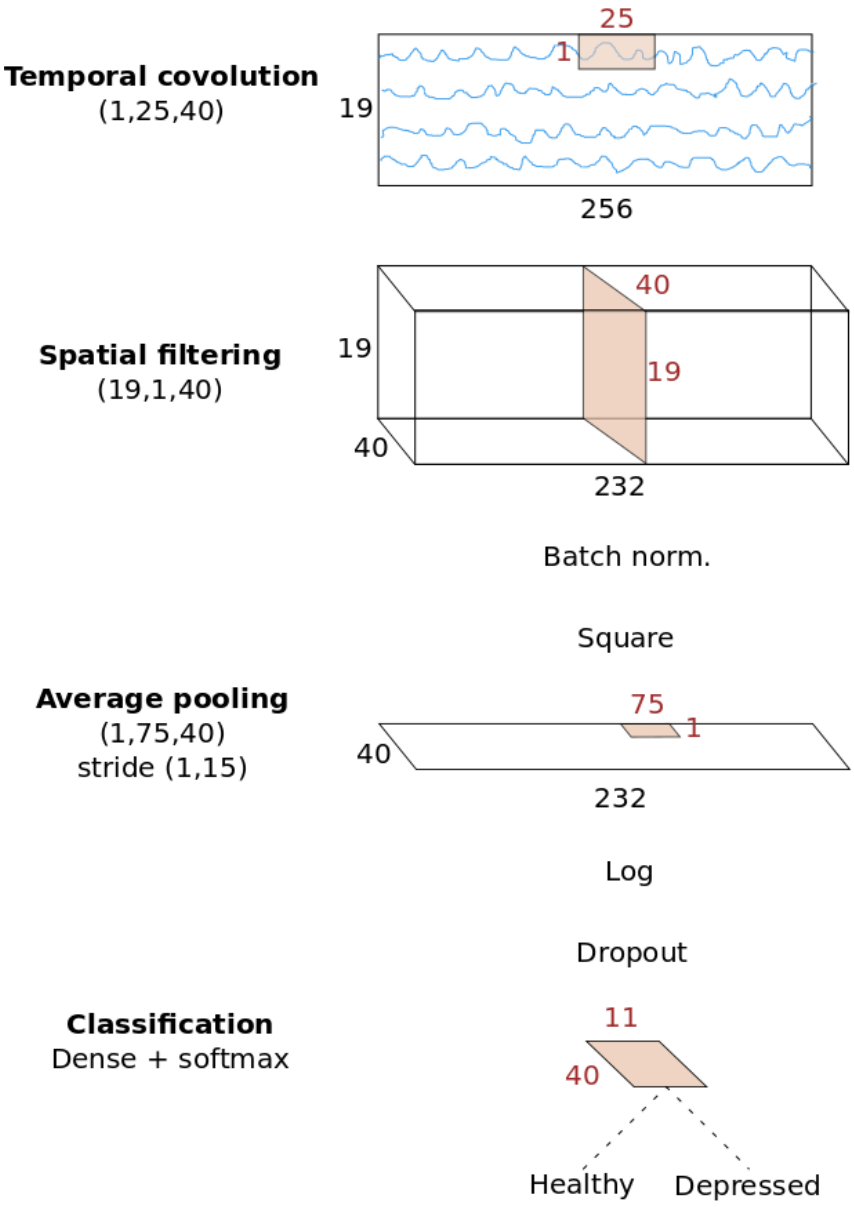


Figure 3.9: Shallow architecture, inspired by FBCSP algorithm, which achieved outstanding performance on the raw data.

3.3.3 Dataset

For experiments in this chapter, we decided to use the entire recordings, in contrast to our approach in the previous chapter, where we used only the beginning in each recording. This is mainly because the classification algorithms we used in this chapter have larger variance, and thus are easily overfit on small datasets. Thus, each of the recordings, after downsampling to 250 Hz (see Section 2.1) was cut into multiple subrecordings of length 256, each subrecording forming a data sample. The subrecording length was selected as a tradeoff between the amount of obtained samples and informationcontained within each sample. For this sampling frequency, this corresponds to approximately a second of recording, which was shown to contain enough information to classify depression with satisfactory accuracy.Moreover, some GPU cards are optimized for working on data chunks multiples of two in size.We also tried increasing the length to 512, but found no improvement in performance.

Is information the right term? The amount of variance?

Cite this!!!

Make more precise.

After splitting the recordings, we assigned positive, neutral or negative label to each subrecording in order to split the dataset into three groups based on depression score of the subject at the time of the recording for depression classification, or based on the subject's before to after treatment depression score for remission classification (in which case only before treatment recordings were further used). The neutral class was then removed and not further considered.

The threshold values separating these classes were selected such that the classes remained relatively balanced and that enough samples were present in each class to train and evaluate a model of moderate capacity. In the case of depression classification, the amount of inter-class variance is inherently limited by nature of the provided data - patients were not randomly sampled, but visited the institution to seek professional help. In attempt to partially remedy this issue, the depression score threshold was set such that 71 patients remained in each depressed and healthy classes, leaving 124 neutral subjects. This corresponds to depressions score ranges $\langle 0, 17\rangle$ for healthy, $(17, 27)$ for neutral, and $\langle 27, 34\rangle$ for depressed. In the case of remission classification, our ability to potentially increase inter-class variance in this way is more limited due the amount of available data, since only before treatment recordings are used. Thus, we removed only 14 neutral subjects, leaving 59 non-remitting and 60 remitting subjects.

Preprocessing: Filtering, running average and standard deviation. Signals were preprocessed before either image-encoding or direct classification. First, the electrode voltages were converted to mV to improve numerical stability. Then, optionally, a high-pass butterworth filter of order 3 with 4 Hz cutoff frequency was applied. It has been suggested that in some cases, filtering the singals may improve classification performance.In image-encoded case, the signals were encoded at this stage. Finally, Welford's algorithm for running mean and standard deviation was used to compute the mean and variance over the whole dataset, and the dataset was then centered and normalized according to the found values.

Elaborate, cite, mayber refer to discussion in previous chapter.

3.3.4 Results

Evaluation: We used 5-fold cross validation, selected the best performing model by evaluation on validation set, run for 200 epochs.

Results: SHAL performs noticably better on REM than DEEP. Filtering has not much effect, but helps slightly for REM, but does not help on DEP.

Conclusions: SHAL has state of the art performance on REM. It would be nice to evaluate against pure FBCSP, and try more traditional configuration to see difference in performance. Combination of CNN and recurrence plots is likely not especially effective.

We might want to show the missclasifications - how close were they? Are people acting, or is the measurement relatively objective? Or maybe confusing matrices.

How about seeing hidden layer activations typical of particular class?

Batchsize 64 128 doesn't matter. The learning rate was decreased with batch size.

Dataset	DEP		REM	
	Neg.	**Pos.**	**Neg.**	**Pos.**
Training	3278	3230	2684	2705
Validation	826	802	686	662
Test	1038	997	830	855
Overall	5142	5029	4200	4222

Table 3.1: Number of negative / positive samples in training, validation, test sets.

Label	Freq.	Arch.	Acc.
DEP	$0 - f_{\text{fin}}$	SHAL	84.53 ± 1.29%
	$4 - f_{\text{fin}}$	SHAL	84.16 ± 1.05%
	$0 - f_{\text{fin}}$	DEEP	**85.65** ± 0.71%
	$4 - f_{\text{fin}}$	DEEP	84.63 ± 2.02%
REM	$0 - f_{\text{fin}}$	SHAL	**93.90** ± 2.34%
	$4 - f_{\text{fin}}$	SHAL	93.47 ± 2.54%
	$0 - f_{\text{fin}}$	DEEP	88.03 ± 1.02%
	$4 - f_{\text{fin}}$	DEEP	86.33 ± 2.30%

(a) Raw data

Label	Freq.	Meth.	Acc.
DEP	$0 - f_{\text{fin}}$	RP	**63.02** ± 1.48%
	$4 - f_{\text{fin}}$	RP	61.31 ± 0.48%
	$0 - f_{\text{fin}}$	CS	59.05 ± 1.52%
	$4 - f_{\text{fin}}$	CS	58.65 ± 0.69%
REM	$0 - f_{\text{fin}}$	RP	60.96 ± 3.05%
	$4 - f_{\text{fin}}$	RP	**64.95** ± 1.78%
	$0 - f_{\text{fin}}$	CS	55.32 ± 1.68%
	$4 - f_{\text{fin}}$	CS	62.78 ± 1.18%

(b) Image-econded data

Table 3.2: Evaluation of accuracies of the shallow (SHAL) and deep (DEEP) architectures on the raw and image-encoded data in classification of depression state (DEP) or prediction of future remission (REM). RP - recurrence plot, CS - cosine similarity.

CHanges to number of any parameter, including adding layers made the results worse. Interestingly, simpler models overfit the training set very quickly, and regularization hurt performance. Hence, this model is probably at least close to local optima in hyperparameter space.

Maybe measure shallow model with more traditional activations to see how much performance is due to FCP.

Conclusion

Bibliography

[1] Henry Abarbanel. *Analysis of observed chaotic data*. Springer Science & Business Media, 2012.

[2] U Rajendra Acharya, Vidya K Sudarshan, Hojjat Adeli, Jayasree Santhosh, Joel EW Koh, and Amir Adeli. Computer-aided diagnosis of depression using eeg signals. *European neurology*, 73(5-6):329–336, 2015.

[3] U Rajendra Acharya, Vidya K Sudarshan, Hojjat Adeli, Jayasree Santhosh, Joel EW Koh, Subha D Puthankatti, and Amir Adeli. A novel depression diagnosis index using nonlinear features in eeg signals. *European neurology*, 74(1-2):79–83, 2015.

[4] Mehran Ahmadlou, Hojjat Adeli, and Amir Adeli. Fractality analysis of frontal brain in major depressive disorder. *International Journal of Psychophysiology*, 85(2):206–211, 2012.

[5] AM Albano and PE Rapp. On the reliability of dynamical measures of eeg signals. In *The 2nd Annual Conference on Nonlinear Dynamics Analysis of the EEG, World Scientific, Singapore*, pages 117–139, 1993.

[6] Galka Andreas. *Topics in nonlinear time series analysis, with implications for EEG analysis*, volume 14. World Scientific, 2000.

[7] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, and Cuntai Guan. Filter bank common spatial pattern (fbcsp) in brain-computer interface. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 2390–2397. IEEE, 2008.

[8] A Babloyantz. Strange attractors in the dynamics of brain activity. In *Complex systems—Operational approaches in neurobiology, physics, and computers*, pages 116–122. Springer, 1985.

[9] Maie Bachmann, Jaanus Lass, Anna Suhhova, and Hiie Hinrikus. Spectral asymmetry and higuchi's fractal dimension measures of depression electroencephalogram. *Computational and mathematical methods in medicine*, 2013, 2013.

[10] Fatemeh Bahari and Amin Janghorbani. Eeg-based emotion recognition using recurrence plot analysis and k nearest neighbor classifier. In *Biomedical Engineering (ICBME), 2013 20th Iranian Conference on*, pages 228–233. IEEE, 2013.

[11] Jan Beran. *Statistics for long-memory processes*. Routledge, 2017.

[12] Peter J Bickel and Peter Bühlmann. What is a linear process? *Proceedings of the National Academy of Sciences*, 93(22):12128–12131, 1996.

[13] Benjamin Blankertz, Ryota Tomioka, Steven Lemm, Motoaki Kawanabe, and K-R Muller. Optimizing spatial filters for robust eeg single-trial analysis. *IEEE Signal processing magazine*, 25(1):41–56, 2008.

[14] Eugene N Bruce, Margaret C Bruce, and Swetha Vennelaganti. Sample entropy tracks changes in eeg power spectrum with sleep state and aging. *Journal of clinical neurophysiology: official publication of the American Electroencephalographic Society*, 26(4):257, 2009.

[15] György Buzsáki and Andreas Draguhn. Neuronal oscillations in cortical networks. *science*, 304(5679):1926–1929, 2004.

[16] Th Buzug and G Pfister. Optimal delay time and embedding dimension for delay-time coordinates by analysis of the global static and local dynamical behavior of strange attractors. *Physical review A*, 45(10):7073, 1992.

[17] Ryan T Canolty, Erik Edwards, Sarang S Dalal, Maryam Soltani, Srikantan S Nagarajan, Heidi E Kirsch, Mitchel S Berger, Nicholas M Barbaro, and Robert T Knight. High gamma power is phase-locked to theta oscillations in human neocortex. *science*, 313(5793):1626–1628, 2006.

[18] Liangyue Cao. Practical method for determining the minimum embedding dimension of a scalar time series. *Physica D: Nonlinear Phenomena*, 110(1-2):43–50, 1997.

[19] Martin Casdagli, Stephen Eubank, J Doyne Farmer, and John Gibson. State space reconstruction in the presence of noise. *Physica D: Nonlinear Phenomena*, 51(1-3):52–98, 1991.

[20] Ahmad Diab, Mahmoud Hassan, Brynjar Karlsson, and Catherine Marque. Effect of decimation on the classification rate of non-linear analysis methods applied to uterine emg signals. *IRBM*, 34(4-5):326–329, 2013.

[21] J.-P Eckmann, S. Oliffson Kamphorst, and D Ruelle. Recurrence Plots of Dynamical Systems. *Europhysics Letters (EPL)*, 4(9):973–977, 1987.

[22] J-P Eckmann, S Oliffson Kamphorst, David Ruelle, and S Ciliberto. Liapunov exponents from time series. *Physical Review A*, 34(6):4971, 1986.

[23] J-P Eckmann and David Ruelle. Ergodic theory of chaos and strange attractors. In *The Theory of Chaotic Attractors*, pages 273–312. Springer, 1985.

[24] J-P Eckmann and David Ruelle. Fundamental limitations for estimating dimensions and lyapunov exponents in dynamical systems. *Physica D: Nonlinear Phenomena*, 56(2-3):185–187, 1992.

[25] Kenneth Falconer. *Fractal geometry: mathematical foundations and applications*. John Wiley & Sons, 2004.

[26] Jens Feder. *Fractals*. Springer Science & Business Media, 2013.

[27] Andrew M Fraser. Reconstructing attractors from scalar time series: A comparison of singular system and redundancy criteria. *Physica D: Nonlinear Phenomena*, 34(3):391–404, 1989.

[28] Andrew M Fraser and Harry L Swinney. Independent coordinates for strange attractors from mutual information. *Physical review A*, 33(2):1134, 1986.

[29] Kunihiko Fukushima and Sei Miyake. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer, 1982.

[30] Enrique Garcia-Ceja, Md Zia Uddin, and Jim Torresen. Classification of recurrence plots' distance matrices with a convolutional neural network for activity recognition. *Procedia computer science*, 130(C):157–163, 2018.

[31] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[32] Peter Grassberger. Do climatic attractors exist? *Nature*, 323(6089):609, 1986.

[33] Peter Grassberger. Grassberger-Procaccia algorithm. `http://www.scholarpedia.org/article/Grassberger-Procaccia_algorithm`, 2007. [Online; accessed 20-December-2018].

[34] Peter Grassberger, Thomas Schreiber, and Carsten Schaffrath. Nonlinear time sequence analysis. *International journal of bifurcation and chaos*, 1(03):521–547, 1991.

[35] Matti Hämäläinen, Riitta Hari, Risto J Ilmoniemi, Jukka Knuutila, and Olli V Lounasmaa. Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Reviews of modern Physics*, 65(2):413, 1993.

[36] Richard Hardstone, Simon-Shlomo Poil, Giuseppina Schiavone, Rick Jansen, Vadim V Nikulin, Huibert D Mansvelder, and Klaus Linkenkaer-Hansen. Detrended fluctuation analysis: a scale-free view on neuronal oscillations. *Frontiers in physiology*, 3:450, 2012.

[37] Erik R Hauge, Jan Øystein Berle, Ketil J Oedegaard, Fred Holsten, and Ole Bernt Fasmer. Nonlinear analysis of motor activity shows differences between schizophrenia and depression: a study using fourier analysis and sample entropy. *PloS one*, 6(1):e16291, 2011.

[38] N Herrmann, SE Black, J Lawrence, C Szekely, and JP Szalai. The sunnybrook stroke study: a prospective study of depressive symptoms and functional outcome. *Stroke*, 29(3):618–624, 1998.

[39] Tomoyuki Higuchi. Approach to an irregular time series on the basis of the fractal theory. *Physica D: Nonlinear Phenomena*, 31(2):277–283, 1988.

[40] Behshad Hosseinifard, Mohammad Hassan Moradi, and Reza Rostami. Classifying depression patients and normal subjects using machine learning techniques and nonlinear features from eeg signal. *Computer methods and programs in biomedicine*, 109(3):339–345, 2013.

[41] D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of Physiology*, 1968.

[42] Harold E Hurst. The problem of long-term storage in reservoirs. *Hydrological Sciences Journal*, 1(3):13–27, 1956.

[43] Harold Edwin Hurst. A suggested statistical model of some time series which occur in nature. *Nature*, 180(4584):494, 1957.

[44] Heinz Isliker and Juergen Kurths. A test for stationarity: finding parts in time series apt for correlation dimension estimates. *International Journal of Bifurcation and Chaos*, 3(06):1573–1579, 1993.

[45] Mainak Jas, Eric Larson, Denis-Alexander Engemann, Jaakko Leppakangas, Samu Taulu, Matti Hamalainen, and Alexandre Gramfort. MEG/EEG group study with MNE: recommendations, quality assessments and best practices. *bioRxiv*, page 240044, 2017.

[46] John M Kane. Factors which can make patients difficult to treat. *The British Journal of Psychiatry*, 169(S31):10–14, 1996.

[47] Holger Kantz and Eckehard Olbrich. Scalar observations from a class of high-dimensional chaotic systems: Limitations of the time delay embedding. *Chaos*, 7(3):423–429, 1997.

[48] Holger Kantz and Thomas Schreiber. *Nonlinear time series analysis*, volume 7. Cambridge university press, 2004.

[49] Alexander Ya Kaplan, Andrew A Fingelkurts, Alexander A Fingelkurts, Sergei V Borisov, and Boris S Darkhovsky. Nonstationary nature of the brain activity as revealed by eeg/meg: methodological, practical and conceptual challenges. *Signal processing*, 85(11):2190–2212, 2005.

[50] Matthew B Kennel, Reggie Brown, and Henry DI Abarbanel. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Physical review A*, 45(6):3403, 1992.

[51] Zoltan J Koles, Michael S Lazar, and Steven Z Zhou. Spatial patterns underlying population differences in the background eeg. *Brain topography*, 2(4):275–284, 1990.

[52] Dimitris Kugiumtzis. State space reconstruction parameters in the analysis of chaotic time series-the role of the time window length. *arXiv preprint comp-gas/9602002*, 1996.

[53] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2323, 1998.

[54] Yann LeCun, Patrick Haffner, Léon Bottou, and Yoshua Bengio. Object recognition with gradient-based learning. In *Shape, contour and grouping in computer vision*, pages 319–345. Springer, 1999.

[55] John Lee. *Introduction to topological manifolds*, volume 202. Springer Science & Business Media, 2010.

[56] Jun-Seok Lee, Byung-Hwan Yang, Jang-Han Lee, Jun-Ho Choi, Ihn-Geun Choi, and Sae-Byul Kim. Detrended fluctuation analysis of resting eeg in depressed outpatients and healthy controls. *Clinical Neurophysiology*, 118(11):2489–2496, 2007.

[57] Klaus Linkenkaer-Hansen, Simo Monto, Heikki Rytsälä, Kirsi Suominen, Erkki Isometsä, and Seppo Kähkönen. Breakdown of long-range temporal correlations in theta oscillations in patients with major depressive disorder. *Journal of Neuroscience*, 25(44):10131–10137, 2005.

[58] Rodolfo R Llinás, Urs Ribary, Daniel Jeanmonod, Eugene Kronberg, and Partha P Mitra. Thalamocortical dysrhythmia: a neurological and neuropsychiatric syndrome characterized by magnetoencephalography. *Proceedings of the National Academy of Sciences*, 96(26):15222–15227, 1999.

[59] JM Martinerie, Alfonso M Albano, AI Mees, and PE Rapp. Mutual information, strange attractors, and the optimal estimation of dimension. *Physical Review A*, 45(10):7058, 1992.

[60] Norbert Marwan, M Carmen Romano, Marco Thiel, and Jürgen Kurths. Recurrence plots for the analysis of complex systems. *Physics reports*, 438(5-6):237–329, 2007.

[61] JH McAuley and CD Marsden. Physiological and pathological tremors and rhythmic central motor control. *Brain*, 123(8):1545–1567, 2000.

[62] AI Mees, PE Rapp, and LS Jennings. Singular-value decomposition and embedding dimension. *Physical Review A*, 36(1):340, 1987.

[63] Kieran J Murphy and James A Brunberg. Adult claustrophobia, anxiety and sedation in mri. *Magnetic resonance imaging*, 15(1):51–54, 1997.

[64] Jean Louis Nandrino, Laurent Pezard, Jacques Martinerie, Farid El Massioui, Bernard Renault, Roland Jouvent, Jean François Allilaire, and Daniel Widlöcher. Decrease of complexity in EEG as a symptom of depression. *NeuroReport*, 5(4):528–530, 1994.

[65] Paul L Nunez, Ramesh Srinivasan, et al. *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press, USA, 2006.

[66] David Nutt, Sue Wilson, and Louise Paterson. Sleep disorders as core symptoms of depression. *Dialogues in clinical neuroscience*, 10(3):329, 2008.

[67] World Heatlh Organization. Depression. `http://www.who.int/en/news-room/fact-sheets/detail/depression`, 2018. [Online; accessed 18-August-2018].

[68] Norman H Packard, James P Crutchfield, J Doyne Farmer, and Robert S Shaw. Geometry from a time series. *Physical review letters*, 45(9):712, 1980.

[69] Laurent Pezard, Jean Louis Nandrino, Bernard Renault, Farid El Massioui, Jean François Allilaire, Johannes Müller, Francisco J. Varela, and Jacques Martinerie. Depression as a dynamical disease. *Biological Psychiatry*, 39(12):991–999, 1996.

[70] Jan Pieter M Pijn, Demetrios N Velis, Marcel J van der Heyden, Jaap DeGoede, Cees WM van Veelen, and Fernando H Lopes da Silva. Nonlinear dynamics of epileptic seizures on basis of intracranial eeg recordings. *Brain topography*, 9(4):249–270, 1997.

[71] Andrew M Pitts. *Nominal sets: Names and symmetry in computer science*. Cambridge University Press, 2013.

[72] Maurice Bertram Priestley. Non-linear and non-stationary time series analysis. 1988.

[73] Itamar Procaccia. Complex or just complicated? *Nature*, 333:498–499, 1988.

[74] Paul E Rapp, Alfonso M Albano, TI Schmah, and LA Farwell. Filtered noise can mimic low-dimensional chaotic attractors. *Physical review E*, 47(4):2289, 1993.

[75] Joshua S Richman and J Randall Moorman. Physiological time-series analysis using approximate entropy and sample entropy. *American Journal of Physiology-Heart and Circulatory Physiology*, 278(6):H2039–H2049, 2000.

[76] Germán Rodríguez-Bermúdez and Pedro J García-Laencina. Analysis of EEG Signals using Non-linear Dynamics and Chaos : A review. *Applied Mathematics & Information Sciences*, 9(5):2309–2321, 2015.

[77] Nicholas Rohrbacker. Analysis of Electroencephologram Data Using Time-Delay Embeddings to Reconstruct Phase Space. *Dynamics at the Horsetooth*, 1:1–11, 2009.

[78] J Röschke, Juergen Fell, and P Beckmann. Nonlinear analysis of sleep eeg in depression: calculation of the largest lyapunov exponent. *European archives of psychiatry and clinical neuroscience*, 245(1):27–35, 1995.

[79] Michael T. Rosenstein, James J. Collins, and Carlo J. De Luca. A practical method for calculating largest Lyapunov exponents from small data sets. *Physica D: Nonlinear Phenomena*, 65(1-2):117–134, 1993.

[80] Michael T Rosenstein, James J Collins, and Carlo J De Luca. Reconstruction expansion as a geometry-based framework for choosing proper delay times. *Physica D: Nonlinear Phenomena*, 73(1-2):82–98, 1994.

[81] J. R??schke, J. Fell, and P. Beckmann. Nonlinear analysis of sleep eeg in depression: Calculation of the largest lyapunov exponent. *European Archives of Psychiatry and Clinical Neuroscience*, 245(1):27–35, 1995.

[82] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic Routing Between Capsules. (Nips), 2017.

[83] Tim Sauer, James A Yorke, and Martin Casdagli. Embedology. *Journal of statistical Physics*, 65(3-4):579–616, 1991.

[84] Timothy D. Sauer. Attractor reconstruction. `http://www.scholarpedia.org/article/Attractor_reconstruction`, 2006. [Online; accessed 28-November-2018].

[85] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[86] Teal L Schultz. Technical tips: Mri compatible eeg electrodes: advantages, disadvantages, and financial feasibility in a clinical setting. *The Neurodiagnostic Journal*, 52(1):69–81, 2012.

[87] Vladimir Shusterman and William C Troy. From baseline to epileptiform activity: a path to synchronized rhythmicity in large-scale neural networks. *Physical Review E*, 77(6):061911, 2008.

[88] Ramesh Srinivasan. Methods to improve the spatial resolution of eeg. *International Journal of Bioelectromagnetism*, 1(1):102–111, 1999.

[89] C. J. Stam. Nonlinear dynamical analysis of EEG and MEG: Review of an emerging field. *Clinical Neurophysiology*, 116(10):2266–2301, 2005.

[90] Steven H Strogatz and Donald E Herbert. Nonlinear dynamics and chaos. *Medical Physics-New York-Institute of Physics*, 23(6):993–995, 1996.

[91] Floris Takens. Detecting strange attractors in turbulence. In *Dynamical systems and turbulence, Warwick 1980*, pages 366–381. Springer, 1981.

[92] James Theiler. Estimating fractal dimension. *JOSA A*, 7(6):1055–1073, 1990.

[93] James Theiler. On the evidence for low-dimensional chaos in an epileptic electroencephalogram. *Physics Letters A*, 196(1-2):335–341, 1994.

[94] James Theiler, Stephen Eubank, André Longtin, Bryan Galdrikian, and J Doyne Farmer. Testing for nonlinearity in time series: the method of surrogate data. *Physica D: Nonlinear Phenomena*, 58(1-4):77–94, 1992.

[95] Sven Vanneste, Jae-Jin Song, and Dirk De Ridder. Thalamocortical dysrhythmia detected by machine learning. *Nature communications*, 9(1):1103, 2018.

[96] Paul M Vespa, Val Nenov, and Marc R Nuwer. Continuous eeg monitoring in the intensive care unit: early findings and clinical efficacy. *Journal of Clinical Neurophysiology*, 16(1):1–13, 1999.

[97] Zhiguang Wang and Tim Oates. Imaging time-series to improve classification and imputation. *arXiv preprint arXiv:1506.00327*, 2015.

[98] Hassler Whitney. Differentiable manifolds. *Annals of Mathematics*, pages 645–680, 1936.

[99] Janet BW Williams and Kenneth A Kobak. Development and reliability of a structured interview guide for the montgomery-åsberg depression rating scale (sigma). *The British Journal of Psychiatry*, 192(1):52–58, 2008.

[100] Alan Wolf, Jack B Swift, Harry L Swinney, and John A Vastano. Determining lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*, 16(3):285–317, 1985.

[101] Dingjun Yu, Hanli Wang, Peiqiu Chen, and Zhihua Wei. Mixed pooling for convolutional neural networks. In *International Conference on Rough Sets and Knowledge Technology*, pages 364–375. Springer, 2014.

[102] Kyongsik Yun, Hee-Kwon Park, Do-Hoon Kwon, Yang-Tae Kim, Sung-Nam Cho, Hyun-Jin Cho, Bradley S Peterson, and Jaeseung Jeong. Decreased cortical complexity in methamphetamine abusers. *Psychiatry Research: Neuroimaging*, 201(3):226–232, 2012.