# Contents

# Introduction

# Chapter 1

# An Introduction to Brain Imaging

In this section, a resume of differnt important aspects in brain modelling and imaging is presented.

Starting from an overview on *resting state* studies, the importance of the study of the *deafult mode network* is underlined especially as far as the disruption of its topological and dynamical properties in several forms of brain disease are concerned.

Then, an introduction to *fMRI* and *BOLD* signal analysis techniques is made in order to define the framework of the data used to train the model proposed to solve the classification problem treated in this thesis.

## 1.1 The resting State

### 1.1.1 Earlier Studies

Many definitions have been brought forth as far as brain's resting state is concerned: a sign of the increasing interest of the scientific community over the years, after the obsversations of its properties.

As a matter of facts, unlike the equilibrium state of an unperturbed noisy physical system, the spontaneous state of the brain does not show a meaningless random activity, as expected by the scientists until two decades ago.

Since the early studies of cerebral metabolism it has been noted that, although the human brain amounts to just 2% of the total body mass, it consumes 20% of the body's energy; these meaurements were made over brains in resting state.

This has led to asking a crucial question: whether cerebral metabolism changes globally when one goes from a quiet rest state to performing a challenging arithmetic problem.

Surprisingly, metabolism remained constant; the local changes were too small (usually less than 5% compared with the resting energy consumption) to be detected by methods designed to measure the energy consumption of the brain as a whole [1].

For several years though, spontaneous brain activity has been systematically overlooked.

As a matter of facts, neuroimaging practices were largely based on the assumption that ongoing activity is sufficiently random and can be averaged out in statistical analysis.

As all the efforts of the scientific community were focused on understanding cognitive behaviour, scans of resting state brain activity were often acquired across these studies for mere control comparison and noise averaging practices, but researchers routinely began noticing that some brain regions showed more activity in resting state condition than during the execution of tasks.

A major step in defining the importance of resting state's studies has been made by the series of pubblication of Raichle, Gusnard and collagues in 2001 [2].

In this study they isolate a set of brain regions, the *Default Mode Network (DMN)*, characterized by surprisingly high metabolic rate during rest, and, on the other hand, by the greatest deactivation during externally imposed cognitive tasks.

Their work propose that DMN is to be studied as a foundamental neurobiological system, like the motor system or the visual system.

It contains a set of interacting brain areas that are *tightly functionally connected* and distinct from other systems within the brain.

An example of DMN anatomy studied via BOLD (blood oxygen level dependant) signal is shown in fig.(1.1).

After this discovery, other patterns of activities were found, leading to the definition of many resting state networks (RSNs) [4].

Different RSNs found by subsequent studies are visible in figure(1.2).

The resting state can thus be defined as a cognitive state in which a subject is quietly awake and alert but does not engage in any specific cognitive or behavioural task [5].

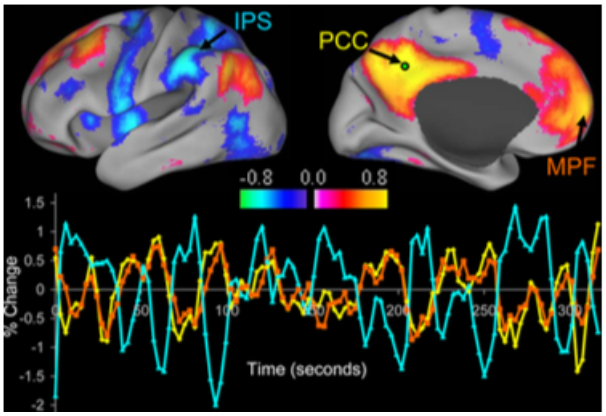


Figure 1.1: *Regions of a single subject's brain that are correlated (positive values) and anticorrelated (negative values) during resting fixation in a functional MRI study.* ***Source: Fox et al (2005). [3]***

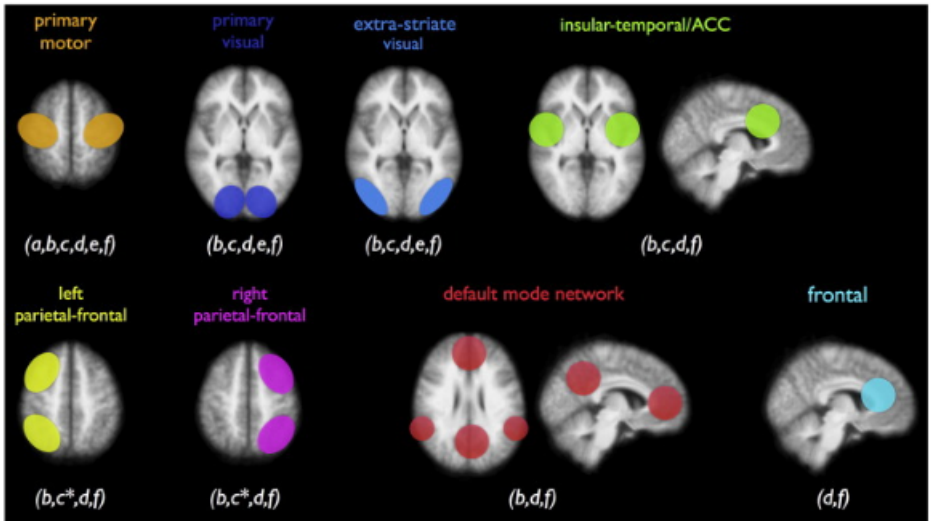


Figure 1.2: *Different Resting State Networks found in literature and summarized by Van Den Heuvel et colleagues, 2010 [4]*

### 1.1.2 The Resting State in Brain Disease

Most, if not all, physiological and psychiatric diseases have been found to have disrupted large-scale functional and/or structural properties.

This fact opens a wide array of possibilities as far as characterization, modelling and predictive studies are concerned for different types of disease.

Disorders like autism, schizophrenia and Alzheimer's disease have all been correlated to resting state network's alterations.

In example, Alzheimer's disesase diagnosed subjects have been found to have enhanced local network properties while having disrupted global properties with respect to non-diagnosed subjects [6].

Results of these type, based on the observations of altered topological properties of functional networks, are shown in fig.(1.3).

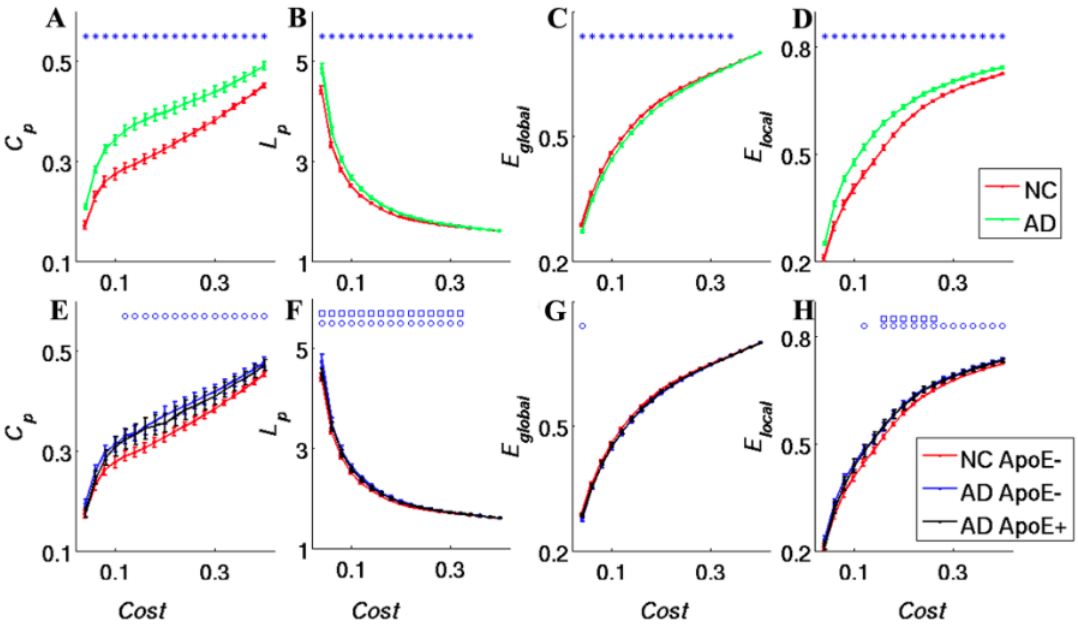


Figure 1.3: *Change of network parameters as a function of connection density (Cost). Clustering coefficient (A), shortest path length (B), global efficiency (C) and local efficiency (D) of the AD (green line) and NC (red line) groups as a function of Cost. Clustering coefficient (E), shortest path length (F), global efficiency (G) and local efficiency (H) of the AD ApoE4+ (black line), AD ApoE4- (blue line) and NC ApoE4- (red line) groups as a function of Cost. The error bars correspond to the standard error of the mean.* ***Source: Xiaohu Zhao , Yong Liu et al. 2012 [6]***

We can see that *clustering coefficients*, the *shortest path length*, l*ocal efficiency*, and *connection density* are all enhanced in Alzheimer's disease diagnosed patients, whereas global efficiency is lower.

As a matter of fact, $C_p$ is a measure of local network connectivity: it reflects the local efficiency and error tolerance of a network.

Higher network clustering coefficients indicate more concentrated clustering of local connections and stronger local information processing capacity [5].

The $C_p$ of brain functional networks was found to be higher in Alzheimer's disease diagnosed patients, indicating that these patients have stronger local information processing capacity [6].

The average shortest path length ($L_p$) of a network reflects how the network connects internally. In brain networks, the shortest path ensures the effective integration and fast transmission of information between distant brain areas.

If the average shortest path of the brain functional networks in Alzheimer's disease diagnosed patients is significantly greater than that in non-diagnosed, it can be stated that the long distance information integration and transmission capacity of neurons is reduced in Alzheimers disease diagnosed patients.

Together with the lower global efficiency in Alzheimer's disease diagnosed subjects, these results may suggest that information transfer between brain regions is more difficult for these subjects [6].

Looking at which brain's regions show a significant variation in topological functional newtork parameters, all of the typical default mode network can be identified as shown in fig.(1.4).

This and many other studies relating to different types of disorders, show how investigating and modeling the default mode networks is of great importance in classification and preventive diagnosis methods implementation.

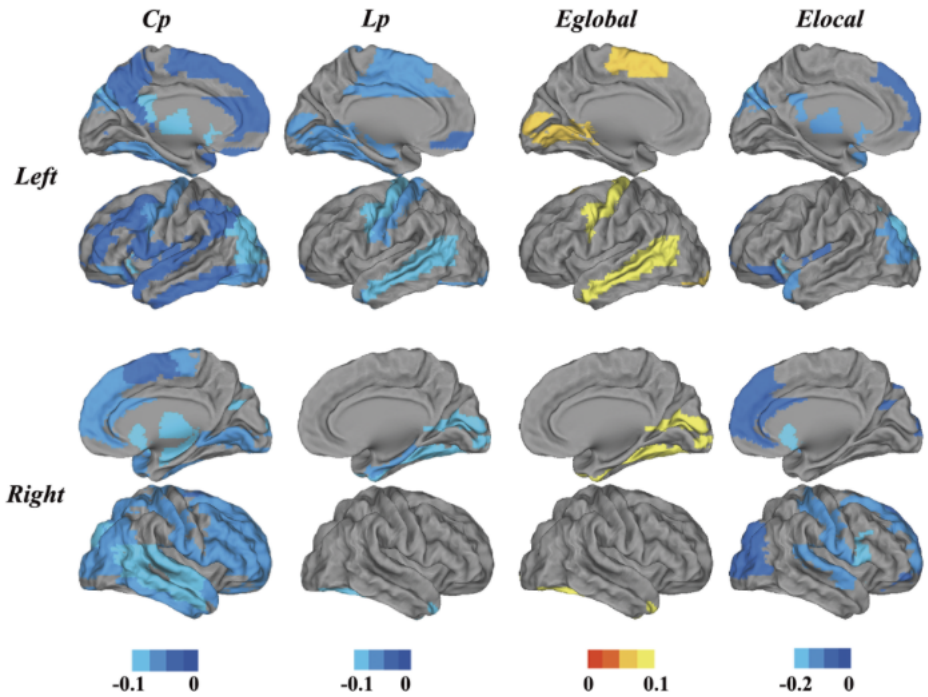


Figure 1.4: *Surface rendering of the distribution of altered nodes at a connection density of 22%. Colored bars indicate differences in network properties between the NC and AD groups. Blue indicates regions showing an increase in the AD group but not the NCs. Yellow indicates regions showing a decrease in the AD group but not the NCs. In the AD group, the regions showing significant increases in $C_p$, $L_p$ and $E_{local}$ are widely distributed across the brain, especially in default mode network regions such as the ACC, PCC, MPFC, HIP and IPL.* ***Source: Xiaohu Zhao , Yong Liu et al. 2012 [6]***

## 1.2 Investigating the Resting State

### 1.2.1 The fMRI Technique

The magnetic resonance imaging (MRI) is a non-invasive method largely used to obtain images of inner structures such as the human body.

The method is based on the magnetic properties of materials composed of nuclei having a non-zero spin.

Such nuclei, when placed in a magnetic field $B_0$, arrange themselves over energetic levels according to the Boltzmann distribution, with the *total magnetization* characterizing the order.

After introducing a perturbating pulse, which has to satisfy the *resonance condition* of the system, the magnetization tends to realign itself with $B_0$ after a characteristic time, in which nuclei make transitions to set back the equilibrium.

The MRI follows the evolution of the system during the return to the equilibrium, obtaining informations about a system's properties and components via their characteristic time.

Let us consider an atomic nucleus with a non-zero total nuclear spin $\overrightarrow{I}$ ; the relation between the magnetic moment $\mu$ and the spin is:

$$\mu = \gamma \hbar I \tag{1.1}$$

where $\gamma$ is the magnetogyric ratio, which is tied to each nuclear isotope. Thus, the component along the $z$ direction of the magnetic moment is:

$$\mu_z = \gamma \hbar m \tag{1.2}$$

where $m$ can take one of the $2I + 1$ values in the interval $[-I, I]$.

For $I = \frac{1}{2}$, an homogeneous applied external magnetic field $B_0$ induces a splitting of the nuclear spin energy level:

$$\Delta E = \gamma \hbar B_0 \tag{1.3}$$

Replacing the Planck-Einstein relation $\Delta E = h\nu$ in the latter equation, the *Larmor resonance frequency* is obtained:

$$\nu_0 = \frac{\gamma}{2\pi} B_0 \tag{1.4}$$

The corresponding pulsation $\omega_0$ is given by:

$$\omega_0 = \gamma B_0 \tag{1.5}$$

The collective motion of a set of $N$ nuclei can then be observed by means of the *total magnetization* $\overrightarrow{M} = N\langle\overrightarrow{\mu}\rangle$.

The evolution in time of the magnetization of a set of nuclei placed a magnetic field $B_0$ is:

$$\frac{d\overrightarrow{M}}{dt} = \gamma\overrightarrow{M} \times B_0 \tag{1.6}$$

The equation describes the precession of $\overrightarrow{M}$ around $\overrightarrow{B_0}$ at the angular velocity $\omega_0$, when $\overrightarrow{M}$ is not aligned with $\overrightarrow{B_0}$.

At the equilibrium, the total magnetization of a paramagnetic material placed in a magnetic field $\overrightarrow{B_0}$, shares the same direction of $\overrightarrow{B_0}$ as stated by *Curie's law*:

$$\overrightarrow{M_0} = C\frac{\overrightarrow{B_0}}{T} \tag{1.7}$$

where $T$ is the absolute temperature and $C$ is the *Curie costant* that tied to material characteristics.

For the sake of simplicity $\overrightarrow{B_0}$ and $\overrightarrow{M_0}$ are considered aligned with the $z$ axis.

Applying a magnetic field $\overrightarrow{B_1}$ orthogonal to $\overrightarrow{B_0}$ with frequency $\nu_0$ causes the magnetization vector to move away from the $z$ axis; the angle between the $z$ axis and the new position of the magnetization vector depends on the duration of the radio frequency (RF) field $\overrightarrow{B_1}$ applied, generated by a coil.

At the end of the pulse application, the spin precession on the transverse plane induces an oscillatory electromotive force in the coil by electromagnetic induction, thus originating a current in the probe.

The detectable signal is called *Free Induction Decay* (FID), which has an oscillating trend with exponential decaying, and it is originated by photons in the radio-wave range emitted by the set of nuclei getting back to equilibrium.

After the RF pulse, the deacy of the NMR signal is analyzed in terms of two separate processes, the longitudinal one and the trasverse one, each with their own time constants.

The underlying process that leads the longitudinal component of the magnetization (along $z$) to reach its equilibrium value $M_0$, is the redistribution of nuclear spin populations according to the Boltzman distribution; such process takes place by energy exchanges between the nuclei and the surroundings.

The longitudinal component of the magnetization decreases in time, as defined by:

$$\frac{dM_z(t)}{dt} = -\frac{(M_z(t) - M_0)}{T_1} \tag{1.8}$$

and thus:

$$M_z(t) = M_z(0)e^{-\frac{t}{T_1}} + M_0(1 - e^{-\frac{t}{T_1}}) \tag{1.9}$$

The underlying process that leads the trasverse component of the magnetization to reach its equilibrium value, i.e. zero, is the decoherence of the transverse nuclear spin magnetization.

Random fluctuations of the local magnetic field lead to random variations in the instantaneous NMR precession frequency of different spins.

As a result, the starting phase coherence of the nuclear spins is lost and the total $xy$ magnetization is null.

The transverse component of the magnetization decays to zero in time according to:

$$\frac{dM_{xy}(t)}{dt} = -\frac{(M_{xy}(t)}{T_2} \tag{1.10}$$

and thus:

$$M_{xy}(t) = M_{xy}(0)e^{-\frac{t}{T_2}} \tag{1.11}$$

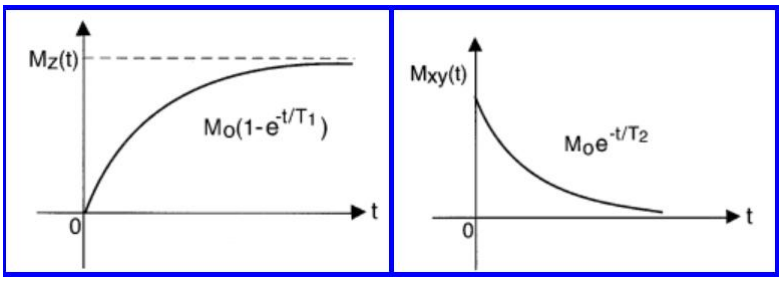


Figure 1.5: *Evolution in time, after the RF pulse, of the longitudinal magnetization (left) and of the transverse magnetization (right) in the rotating frame.* ***Source: https://warwick.ac.uk/fac/sci/physics/research/condensedmatt***

## 1.3 The BOLD Signal

The *Blood Oxygen Level Dependant* signal is a measure of the amount of the oxygen contained in blood flowing towards neural regions.

To function properly, the brain needs energy in the form of *Adenine-TriPhosphate* (ATP), which is in turn produced through a chemical reactions involving glucose and oxygen.

As neither glucose nor oxygen are stored in the brain by default, they need to be carried to the brain via circuclatory system.

Oxygen is transported by *haemoglobin*, in a chemical form known as oxy-haemoglobin, in contrast to *deoxy-haemoglobin*, the form haemoglobin assumes when it releases the transported oxygen.

These two molecular forms differ by their magnetic properties: oxy-haemoglobin is paramagnetic, whereas deoxy-haemoglobin is diamagnetic (fig.(1.6)).

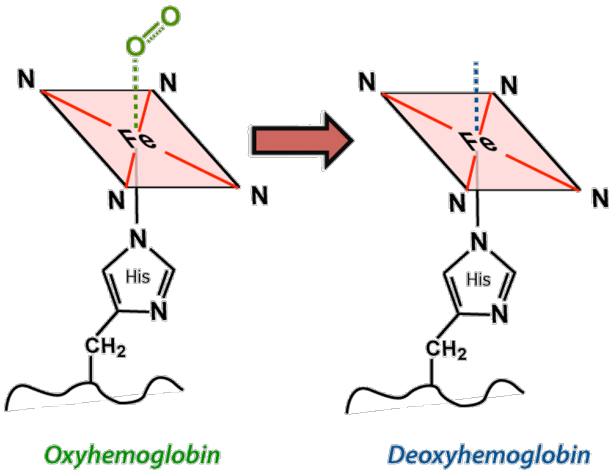


Figure 1.6: *Deoxy-haemoglobin is trongly paramagnetic due to 4 unpaired electrons at each iron center.* ***Source: mriquestion.com/bold-contrast.html***

When energy is required in a particular area, or in other terms a particular cerebral area is activated due to a given cognitive task, the amount of incoming oxygen (oxyhaemoglobin) is much higher than the oxygen being consumed to form ATP.

As a result these areas show an increase in signal intensity.

More precisely, since the deoxyhemoglobin is paramagnetic, it is able to reduce the NMR signal in $T_2$ weighted images: indeed the rate of loss of proton spin phase coherence, measured through $T_2$, can be modulated by

the presence of intravoxel deoxyhaemoglobin. On the contrary, being the oxyhemoglobin diamagnetic, it does not modify the NMR signal.

Thus, during the neural activation of a brain area, an higher incoming blood flux is observed with respect to the blood incoming during rest; in such area blood vessels expand and the transported oxygen rate is higher than oxygen consumed rate in burning glucose.

Therefore, although paradoxical, in the active brain region the concentration of oxygenated blood increases, and the concentration of deoxygenated blood decreases with respect to the neighbour inactive brain areas. Such process is shown in fig.(1.7).

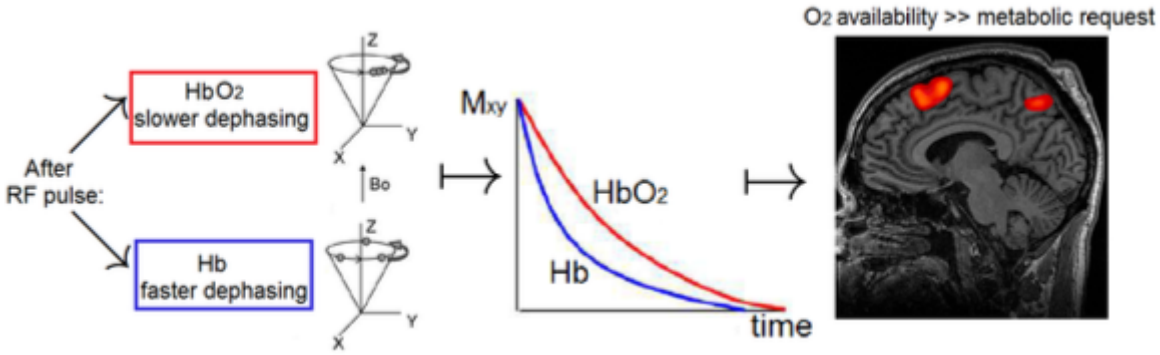


Figure 1.7: *BOLD signal formation process*

It has to be pointed out, however, that the oxygen influx iunderlying the BOLD signal is not an immediate consequence of neural activity;they are rather parallel processes.

In fact *Glutamate-generated Calcium influx* releases many vasodilators.

Blood flow is related more to local *field potentials* than individual neurons spiking [7]; it can therefore be statede that the signal is increased over an area larger than the one with specific neuronal activity.

Given the high number of processes underlying global blood flow changes, a model tying BOLD signals to neural activity is required for those fMRI aplications whose goal is to observe and characterize neural processes.

The link between the the BOLD signal and the effective neural signal lies in the *haemodynamic response function (HRF)*.

Formally, the BOLD signal can be interpreted as a *convolution* between the actual neural signal and the HRF; the system's response to the stymulus is in ths way obtained.

If we take $N(t)$ as the neural activity signa and $h(t)$ as HDF we have:

$$B(t) = N(t) * h(t) = \int_0^t N(\tau)h(t-\tau)d\tau \tag{1.12}$$

In order for eq.(1.12) to be true, the assumption that the system's response is *linear and time invariant* has to be made. We then define the neural signal as:

$$N(t) = \int_{t=0}^{\infty} \delta(t-\tau)n_\tau d\tau \tag{1.13}$$

Given that the BOLD function is a function of neural activity $B(t) = f(N(t))$, linearity implies that:

$$f(\int_{t=0}^{\infty} \delta(t-\tau)n_\tau d\tau) = \int_{t=0}^{\infty} f[\delta(t-\tau)]n_\tau d\tau \tag{1.14}$$

thus proving eq.(1.7).

Although many evidences indicate that BOLD signal is non-linear, deviations from linearity are often small and linearity assumptions are quite valid in many cases and apllications.

As the HRF depends on the ways oxygen is consumed when energy is required by neurons, it is complex to model as a function.

Even though eq.(1.7) holds true, most of the possible issues derive from $N(t)$ and $h(t)$ being both unknown in a large number of problems.

As a result, the HRF is often *estimated* in order to be able to calculate the neural signal.

Many of the commonly used estimation methods rely upon recording the response to a given neural input, deconvolving eq.(1.7) while assuming a model for $N(t)$, or trying to guess the function and smooth it with some kind of parameter fitting.

Whichever the technique employed, one main feature of the HRF Must be taken into account for the consequences it induces on the BOLD signal $B(t)$: $h(t)$ has a low response, due to oxygen being quite slow (approximately 10 seconds) to reach its maximum value after neural activation.

As a consequence, for short and close neural impulses, the BOLD response does not have the chance to decrease, because of the convolution with the HRF.

It thus can be stated that BOLD *always smooths* the real underlying neural signal.

## 1.4 Data Preprocessing

The aim of data preprocessing is to reduce the statistical noise, in order to better extract the true signal.

This is most important for resting state analysis, given that there is no peak in the time series compared to the average value of the entire series.

Looking at fig.(1.2) in example, we can see that, as a matter of facts,changes in BOLD signal rarely exceed the 1%.

A resume of the most common preprocessing practices for fMRI studies is done in the following section.

- **Slice-timing correction**: All fMRI data are collected in *slices*, which in turn contain arrays of voxels.

  The *Repetition Time TR* is the time which separates the onsets of consecutive whole brain scans; thus is the time needed to collect data from all the brain voxels at a given point in time.

  This causes some problems to arise, i.e. not all the brain voxels in a given TR are acquired simultaneously: a TR time is needed to take all the slices between the first and the last.

  The bias is dependent on the order in which slices are taken.

  The most common approach to deal with such an issue is a form of *temporal interpolation*, which can be linear, spline, or sinc.

  Linear interpolation is good when data do not vary much rapidly from one time acquisition to the other. This approach simply consists of estimating a continuous BOLD function from the discrete sampling; when using a linear function, a simple line is estimated between two points, and the value at the point of interest is taken.

- **Head motion correction**: It is probably the most important preprocessing step.

  During task studies, a movement of $5mm$ can increase activation values by a factor of 5, and it can completely mix up signal from different voxels in resting state studies.

  All the corrections are based on the assumption that during movements the brain does not change size or shape; as a result, the only changes are due to rigid body movement.

As a matter of fact, the movement can be characterised by *six parameters*, 3 translational and 3 rotational, as described by classical mechanics.

Voxels are defined by their position within the scanner,*rather than by position within the subject's brain.*

To correct for this a *rigid body registration* is performed.

- **Normalization**: each individual presents morphologically different brains, with different global size and different local regions sizes too.

  This leads to many problems while studying if a signal observed in one region is observed in the same region in another patient.

  Moreover, when performing group analyses, brains have to be overlapped, in order to increase the signal to noise ratio and the statistical significance of the analysis.

  However, if we overlap two significally different regions, the signal is quite likely to average out.

  Therefore, warping one subject's structural image to a *standard brain atlas* is a required step.

  The most common reference system is the MNI space, developed by the Montreal Neurological Institute [8].

- **Coregistration**: functional data have to be mapped onto structural data in order to as- sess the exact region the signal is coming from.

  These is not straightforward due to the two images being taken with different spatial resolutions. Given that functional images have to be taken within a few seconds, as a consequence of a speed-accuracy tradeoff, they often have poor spatial resolution.

  Structural images, on the other hand, can take up to 10 minutes in order to be acquired if a precise mapping of every region is to be obtained.

  This results in different voxel sizes: a typical fMRI voxel is $(3 \times 3 \times 3.5)mm$, whereas sMRI can have voxels with sizes down to $(0.86.86 \times 0.89)mm$. After coregistration, however, structural resolution can be employed to improve functional resolution.

  Early methods aimed at identifying key landmarks in the two different images and then trying to align them, but given the scarce automation reliabiity of this process most methods now relies upon the minimisation of mutual information between histograms of the images.

- **Data smoothing**: intensity values from individual voxel have an embedded component of noise.

  In order to reduce this noise spatial smoothing is needed; basically the intensity value of a voxel is replaced with a weighted average of the values of neighbouring voxels, through the convolution between the voxels and a function representing the neighbourhood known as *kernel*.

  In this way, close voxels contribute much more than distant voxels.

  Other than increasing SNR, this process is useful since, as explained by the *central limit theorem*, it allows the distribution of intensities to become normal thus helping the multiple comparison analysis in task studies.

## 1.5 The Functional Connectivity Matrix

One of the most common approaches in resting state studies gravitates around the notion of *Functional Connectivity*.

Functional Connectivity is defined as *the statistical association or dependency among two or more anatomically distinct time-series* [9].

In FC analyses, there is no inference about coupling between regions; that is it does not tell *how* regions are coupled.

In fact, it only tests some form of correlation against the null hypothesis.

FC is however useful to *discover patterns* (which regions are coupled), and compare patterns, especially between groups.

In practice FC can be represented by a matrix whose entry $a_{ij}$ is a correlation between the intrinsic activity of neural source $i$ and neural source $j$.

Common examples of correlations measures computed on time-series data types are the *cross-correlation* and *cross-coherence* [9].

Cross correlation between regions 1 and 2 with a time delay $t$ is given by:

$$R(t) = \frac{cov(1, 2+t)}{\sqrt{var(s1) + var(s2+t)}} \tag{1.15}$$

Cross-coherence can be defined as equivalent to cross-correlation but in the *frequency domain*.

# Chapter 2

# The Wishart Distribution

## 2.1 Definition

The *wishart distribution* $W_p(n, \Sigma)$ is a probability distribution of random nonnegative-definite $p \times p$ matrices that is used to model random covariance matrices.

The parameter $n$ is the number of degrees of freedom, and $\Sigma$ is a nonnegative-definite symmetric $p \times p$ matrix, called the *scale matrix*.

*Def.* Let $X_1 ... X_n$ be $N_p(0, \Sigma)$ distribuited vectors, forming a data matrix $p \times n$, $X = [X_1 ... X_n]$. The distribution of a $p \times p$, $M = XX' = \Sigma_{i=1}^{n} X_i X_i'$ *random matrix* is a Wishart distribution. [10]

We have then by definition:

$$M \sim W_p(n, \Sigma) \sim \Sigma_{i=1}^{n} X_i X_i' \qquad X_i \sim N_p(0, \Sigma) \tag{2.1}$$

so that $M \sim W_p(n, \Sigma)$ is the distribution of a sum of $n$ rank-one matrices defined by independent normal $X_i \in R^p$ with $E(X) = 0$ and $Cov(X) = \Sigma$.

In particular, it holds for the present case:

$$E(M) = nE(X_i X_i') = nCov(X_i) = n\Sigma \tag{2.2}$$

## 2.2 PDF Computation for Invertible $\Sigma$

In general, any $X \sim N(\mu, \Sigma)$ can be represented as

$$X = \mu + AZ, \quad Z \sim N(0, I_p) \tag{2.3}$$

so that

$$\Sigma = Cov(X) = ACov(Z)A' = AA' \tag{2.4}$$

The easiest way to find $A$ in terms of $\Sigma$ is the LU-decomposition, which finds a unique lower diagonal matrix $A$ with $A_{ii} \geqslant 0$ such that $AA' = \Sigma$.

Then by 2.1 and 2.4, with $\mu = 0$ we have:

$$W_p(n, \Sigma) \sim \sum_{i=1}^{n} (AZ_i)(AZ_i)' \sim A(\sum_{i=1}^{n} Z_i Z_i')A' \sim AW_p(n)A' \tag{2.5}$$

where $Z_i \sim N(0, I_p)$ and $W_p(n) = W_p(I_p, n)$.

Assuming that $n \geq p$ and $\Sigma$ is invertible, the density of the random $p \times p$ matrix $M$ in 2.1 can be written [1] :

$$f(M, n, \Sigma) = \frac{1}{2^{\frac{np}{2}} \Gamma_p(\frac{n}{2}) \|\Sigma\|^{\frac{n}{2}}} \|M\|^{\frac{n-p-1}{2}} exp[-\frac{1}{2} trace(\Sigma^{-1} M)] \tag{2.6}$$

so that $f(M, n, \Sigma) = 0$ unless $M$ is *symmetric and positive-definite.* [11]

Note that in 2.6 we define $\Gamma_p(\alpha)$ as the *generalized gamma function* $\Gamma_p(\alpha) = \pi^{\frac{p(p-1)}{4}} \prod_{i=1}^{p} \Gamma(\frac{2\alpha+1-i}{2})$

### 2.2.1 Visualizing the Wishart Distribution

The Wishart distribution is a generalization to multiple dimensions of the *chi-squared distribution*, or in the case of non-integer degrees of freedom, of the *gamma distribution.*

We show as a proof in fig.2.1 that for a 1-dimensional and equal to 1 $\Sigma$ scale matrix, the Wishart distribution $W_1(n, 1)$ collapses to the $\chi^2(n)$ distribution.

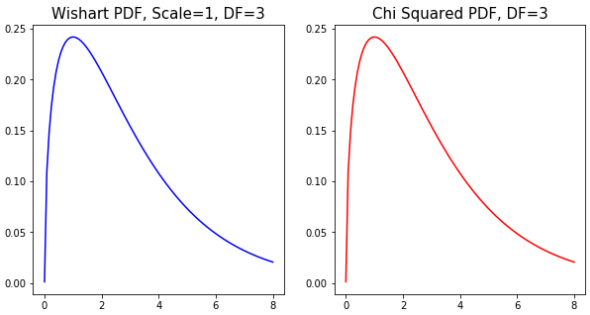


Figure 2.1: *Monodimensional Wishart Distribution and* $\chi^2(n)$ *distribution comparison*

[1] *Note:* $\|\Sigma, N\| = det(\Sigma, M)$

Save for this simple case, being the Wishart a distribution over matrices, it is a generally hard task to visualize it as a density function.

We can however sample from it and use the eigenvectors and eigenvalues of the resulting sampled matrix to define an ellipse.

An example of this technique is shown in fig.2.2. A set of five sampled matrices is drawn for each plot. While the parameter $n = 2$ (*degrees of freedom*) is the same for both the samplings shown, a different *scale matrix* $\Sigma$ is used for each plot.

Note that for $\Sigma = I_2$ (left plot in fig.2.2) the sample would look *on average* like circles.

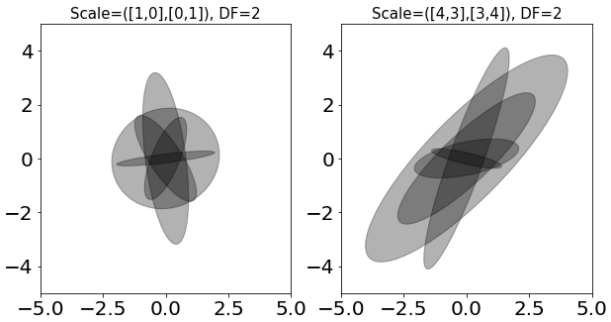


Figure 2.2: *Plot of eigenvalue and eigenvectors defined ellipses, drawn from different scale matrix defined Wishart-sampled distribution.*

## 2.3 The Wishart Distribution in Bayesian Conjugate Prior Analysis

An important use of the Wishart distribution is as a conjugate prior for *multivariate normal sampling*. We now recall some basics concepts about Bayesian inference and prediction in order to show the application of the Wishart in those fields.

### 2.3.1 Bayesian Inference and Priors Distributions

The distinctive feature of the Bayesian approach underlies in its way of defining probability.

Probability is treated as *belief* and not as *frequency*, thus introducing a fundamental difference between the Bayesian and the *frequentist* approach and shifting the goal toward the analysis and statement of a *belief* [12].

We can sum up the process of Bayesian inference as follows:

- A probability density called *prior distribution* $\pi(\theta)$ is chosen, expressing the *beliefs* about a parameter $\theta$ before any data are seen.
- A statistical model $p(x \mid \theta)$ is chosen, which must reflect the beliefs about $x$ given $\theta$.
- After observing the data $D_n = [X_1...X_n]$, the beliefs is updated and the *posterior distribution* $p(\theta \mid D_n)$ is computed.

By Bayes' theorem the posterior distribution can be written as

$$p(\theta \mid X_1...X_n) = \frac{p(X_1...X_n \mid \theta)\pi(\theta)}{p(X_1...X_n)} = \frac{L_n(\theta)\pi(\theta)}{c_n} \propto L_n(\theta)\pi(\theta) \tag{2.7}$$

where $L_n(\theta) = \prod_{i=1}^{n} p(X_i \mid \theta)$ is the likelihood function and the *normalizing constant* $c_n$ is defined as follows:

$$c_n = p(X_1...X_n) = \int p(X_1...X_n \mid \theta)\pi(\theta)d\theta = \int L_n(\theta)\pi(\theta)d\theta \tag{2.8}$$

the normalizing constant is also called the *evidence*.

We now define the general properties of a *conjugate prior*.

If, for a given problem, the posterior distribution $p(\theta \mid X_1...X_n)$ and the prior $\pi(\theta)$ belong to the same family of distribution, they're called *conjugated distributions* and the prior is said to be a *conjugate prior* for the given *likelihood function* $L_n = p(X_1...X_n \mid \theta)$.

A classical example concerns the Gaussian Distribution: the Gaussian family is conjugate to itself (or self-conjugate) with respect to a Gaussian likelihood function: if the likelihood function is Gaussian, choosing a Gaussian prior over the mean will ensure that the posterior distribution is also Gaussian [13].

Considering the general problem of inferring a distribution for a parameter $\theta$ given some observations $D_n = [X_1...X_n]$ and referring to theorem 2.7, by which we let the likelihood function be considered fixed as it is usually well-determined from a statement of the data-generating process, it is clear that different choices of the prior distribution $\pi(\theta)$ may make the integral in 2.8 more or less difficult to compute. The product $L_n(\theta)\pi(\theta)$ will also be influenced, gaining the possibility to take one algebraic form or another.

If for certain choices of the prior the posterior has the same algebraic form as the prior, those choices are said to yield a *conjugate prior*.

It is then possible to state that a conjugate prior is an algebraic convenience giving a closed-under-sampling-form expression for the posterior.

### 2.3.2 The Wishart Conjugate Prior

We now show how the Wishart Distribution is correlated to the*Inverse Gamma Distribution* in a multidimensional setting, by considering a Gaussian model with known mean $\mu$, so that the free parameter is the variance $\sigma^2$, as in [12].

The likelihood function is defined as follows:

$$p(X_1...X_n \mid \sigma^2) \propto (\sigma^2)^{-\frac{n}{2}} exp(-\frac{1}{2\sigma^2} n\overline{(X-\mu^2)}), \qquad \overline{(X-\mu^2)} = \frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2 \tag{2.9}$$

The conjugate prior is an inverse Gamma distribution. Recall that $\theta$ has an inverse Gamma distribution with parameters $(\alpha, \beta)$ when $\frac{1}{\theta} \sim Gamma(\alpha, \beta)$.

The density is then bound to take the form

$$\pi_{\alpha,\beta}(\theta) \propto \theta^{-(\alpha+1)} e^{-\frac{\beta}{\theta}} \tag{2.10}$$

Using this prior, the posterior distribution of $\sigma^2$ is given by

$$p(\sigma^2 \mid X_1...X_n) \sim InvGamma(\alpha + \frac{n}{2}, \beta + \frac{n}{2}\overline{(X-\mu^2)}) \tag{2.11}$$

An alternative way of parameterization of the prior is given by the *Inverse Scaled $\chi^2$ Distribution*, whose density is defined as

$$\pi_{\nu_0,\sigma_0^2} \propto \theta^{-(1+\frac{n_0}{2})} exp(-\frac{\nu_0\sigma_0^2}{2\theta}) \tag{2.12}$$

Under this kind of parameterization of the prior, the posterior takes the form

$$p(\sigma^2 \mid X_1...X_n) \sim ScaledInv\chi^2(\nu_0 + n, \frac{\nu_0\sigma_0^2}{\nu_0+n} + \frac{n(X-\mu^2)}{\nu_0+n}) \tag{2.13}$$

In the multidimensional setting, the inverse Wishart takes the place of the inverse Gamma.

It has already been stated that the Wishart distribution is a distribution over *symmetric positive semi-definite* $d \times d$ matrices $W$. A more compact form of the density is given by

$$\pi_{\nu_0,S_0}(W) \propto \mid W \mid^{\frac{(\nu_0-d-1)}{2}} exp(-\frac{1}{2} trace(S_0^{-1}W)), \qquad \mid W \mid = det(W) \tag{2.14}$$

where the parameters are the degrees of freedom $\nu_0$ and the positive-definite *scale matrix* $S_0$.

If $W^{-1} \sim Wishart(\nu_0, S_0)$ we can then state that $W$ has an *Inverse Wishart Distribution*, whose density has the form

$$\pi_{\nu_0,S_0}(W) \propto \mid W \mid^{-\frac{(\nu_0+d+1)}{2}} exp(-\frac{1}{2}trace(S_0W^{-1})), \qquad \mid W \mid = det(W) \tag{2.15}$$

Let $X_1...X_n$ be $N(0,\Sigma)$ distributed observed data. Then an inverse Wishart prior multiplying the likelihood $p(X_1...X_n \mid \Sigma)$ yields

$$\begin{aligned} & p(X_1...X_n \mid \Sigma)\pi_{\nu_0,S_0}(\Sigma) \propto \\ \mid \Sigma \mid^{-\frac{n}{2}} exp(-\frac{n}{2}trace(\overline{S}\Sigma^{-1}) & \mid \Sigma \mid^{-\frac{(\nu_0+d+1)}{2}} exp(-\frac{1}{2}trace(S_0\Sigma^{-1})) \\ & = \mid \Sigma \mid^{-\frac{(\nu_0+d+n+1)}{2}} exp(-\frac{1}{2}trace((n\overline{S}+S_0)\Sigma^{-1})) \end{aligned} \tag{2.16}$$

where $\overline{S}$ is the *empirical covariance* $\overline{S} = \frac{1}{n}\sum_{i=1}^{n} X_iX_i^T$.

Thus, a posterior with the form

$$p(\Sigma \mid X_1...X_n) \sim InvWishart(\nu_0 + n, n\overline{S} + S_0) \tag{2.17}$$

is obtained.

Analogally, it can be stated that for the inverse covariance (*precision*) matrix $\Sigma^{-1}$ the conjugate prior is a Wishart distribution.

# Chapter 3

# The WISDoM Multiple Order Classification

In this section, the classification method implemented and used on the ADNI2 and ABIDE databases is described, both in an analytical and technical way.

The Wishart Distributed Matrices Multiple Order Classification is a method that allows both classification and *feature selection* for any classification problem whose elements can be tied to a *symmetric positive-definite* matrix representation (i.e. covariance and correlation matrices).

The *"distance"* used to train the classifier is defined as well as the *feature transformation* undergone by the each of the subject analyzed.

The general pipeline and the validation pipeline are then discussed while also introducing an example of possible parallelization for performance enhancing.

## 3.1 Wishart Sampling

Considering what has been said in the last section, using the Wishart distribution to model and sample the elements of a wide range of problems follows naturally.

As a matter of fact, every calssification problem whose elements take the form of *symmetric positive-definite* matrices can be approached with the method we are about to discuss.

The main idea for the *WISDoM Classifier* is to use the *free parameters* of the Wishart distribution (the scale matrix $S_0$ and the number $n$ of the degree of freedom, as shown in 2.6) to compute an estimation of the distribution for

a certain class of elements, and then assign a single element to a given class by computing some sort of distance between the element being analyzed and the classes.

Furhermore, if we assume that the matrices are somehow representative of the *features* of the system studied (i.e. covariance matrices might be taken into account), a score can be assigned to each feature by estimating the weight of said feature in terms of *Log Likelihood Ratio*.

In other words, a score can be assigned to each feature by analyzing the variation in terms of *LogLikelihood* caused by the deletion of it.

If the deletion of a feature causes significant increase (or decrease) in the *LogLikelihood* computed with respect to the *estimated distributions* for the classes, it can be stated that said feature is highly representative of the system analyzed.

It is now clear that the simplest usable objects to estimate the distribution for a class and to represent its elements is the *covariance matrix*. Further proofs for this statement will be given later on.

Thus, the aim of the WISDoM classifier is not only to assign a given element to the optimal class, but also to identify the features with the highest *"weights"* in the decision process.

### 3.1.1 Computing the Estimated Distribution

Let us briefly recall the parametrization of the Wishart Distribution in order to clearly define the application conditions for classification problems.

Let $X_1...X_n$ be independent $N_p(0, \Sigma)$ distributed vectors, forming a data matrix $p \times n$, $X = [X_1...X_n]$. The distribution of a $p \times p$, $M = XX' = \Sigma_{i=1}^{n} X_i X_i'$ *random matrix* is a Wishart distribution with parameters $W_p(n, S_0)$. In the previous chapter (2.6) it has been proved that for normal distributed data, for $S_0 = \Sigma$, a distribution of *random covariance matrices* is obtained.

In a similar fashion, if a good choice for the scale matrix $S_0$ is made for a given class, a representative distribution for the class can be estimated and samples can be drawn from it.

*Covariance matrices* are a good choice, although not limiting as long as the matrices are symmetric and positive-definite, both for the way they represent a system and for the property that *the mean of a set of covariance matrices is a covariance matrix.*

If each element of a given class $C$ is represented by a covariance matrix $\Sigma$ of its features, this property allows us to estimate a distribution for the

class by choosing

$$S_0 = \hat{\Sigma}_C = \frac{1}{N}\sum_{i=1}^{N}\Sigma_i \tag{3.1}$$

The other necessary parameter for the estimation is the *degrees of freedom* $n$.

Assume that an $X_i = (x_1, ..., x_p)$ vector of $p$ features is associated to each element $i$ of a given class, while having $n$ observation of said vector. The covariance matrix $\Sigma_i$ computed over the $n$ observations will represent the "interactions" between the features of element $i$.

The degrees of freedom $n$ of the Wishart distribution are then given by the number of times $X_i$ is observed.

Let us introduce an example tied to *functional MR brain imaging* in order to further clarify the concepts being introduced.

An image of patient $i$'s brain is acquired; as usual these images are divided in a certain number $p$ of zones (voxel, pixel etc.), each zone being sampled $n$ times over a given time interval in order to observe a certain type of brain activity and functionality.

It is now clear that the features contained in vector $X_i = (x_1, .., x_p)$ associated to patient $i$ are indeed the zones chosen to divide $i$'s brain image, each zone having been sampled $n$ times during an acquisition interval.

The correlation $p \times p$ matrix $\Sigma_i$ computed for $i$'s observation is then representative of the functional correlation between the $p$ zones of $i$'s brain.

Repeating this procedure for $N$ patients belonging to a known class $C$ (i.e. a diagnostic group) and computing the $\hat{\Sigma}_C$ scale matrix for the class as stated before, will allow us to estimate a wishart distribution for that class correlation matrices and draw samples from it.

The module used for Wishart generation and sampling by the WISDoM calssifier is the *SciPy.Stats.Wishart* module of the *SciPy* Python3.6 library.

Further details on the generation and sampling algorithm used by the module can be found in [14].

Some samples drawn from Wishart distributions computed with different $5 \times 5$ scale matrices and degrees of freedom are shown in fig.3.1.

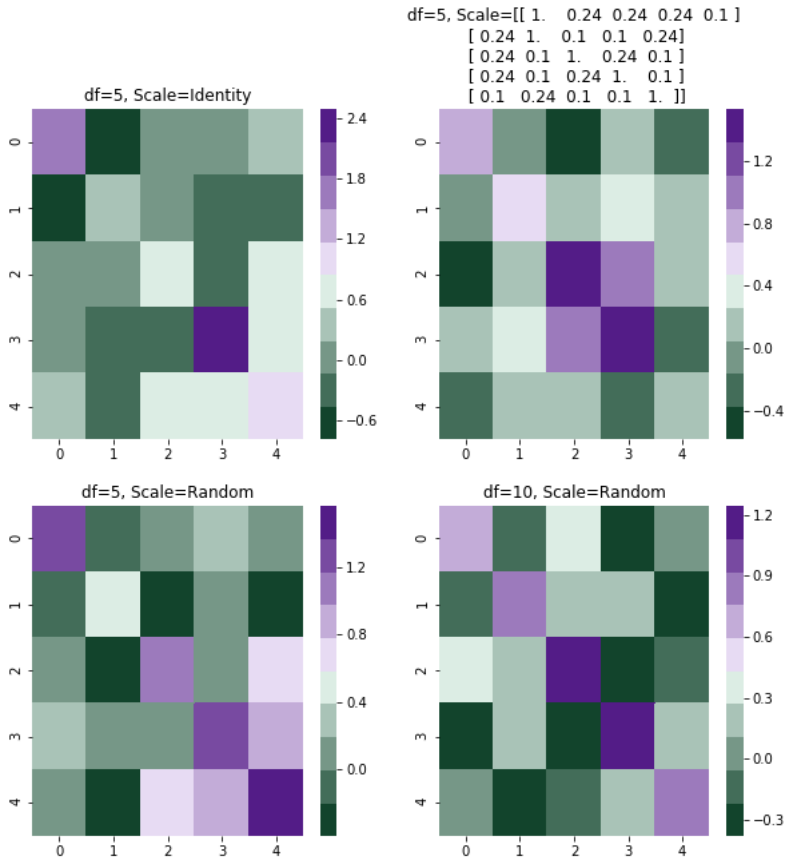


Figure 3.1: *Various sampling from different wishart distribution. A diverging heatmap has been chosen to visualize the values of each sample's elements.*

## 3.2 Log-Likelihood Ratio Distance

After the definition of the role of the Wishart distribution in symmetric positive definite matrices' modeling, it is necessary to define some sort of distance between the estimated distribution for a class $C$ and its hypotetical elements.

As stated before, this will be done in terms of both entire matrices and *single features*, in order to achieve optimal classification and exract information about a system's most meaningful components.

### 3.2.1 Complete Matrix Distance

The scoring system used by the WISDoM Classifier relies on the *logpdf* function from the *SciPy.Stats.Wishart* module in order to compute the LogLikelihood of a matrix $\Sigma_i$ with respect to the Wishart distribution estimated for a class $C$, using $\hat{\Sigma}_C$ as the scale marix.

If a problem concering two given classes $C_A$ and $C_B$ is taken into account, the score assigned to each $\Sigma_i$ upon which the classification decision is based, can be defined as follows:

$$score_i = logP_W(\Sigma_i \mid n, \hat{\Sigma}_A) - logP_W(\Sigma_i \mid n, \hat{\Sigma}_B) \tag{3.2}$$

Where $\hat{\Sigma}_{A,B}$ are the scale matrix computed for the classes $A, B$ and $logP_W(\Sigma_i \mid n, \hat{\Sigma}_{A,B})$ can be seen as the logarithm of the probability of $\Sigma_i$ belonging to the Wishart distribution estimated for one of the two classes $A, B$.

### 3.2.2 Single Feature Distance and Multiple Order Reduction

The aim of the WISDoM classifier is to further increase the informations obtained about the system's features during the classification.

To do this it is then necessary to introduce some matemathical properties of the symmetric positive deifnite matrices, upon which the method relies.

It will be shown that it is indeed possible to access different orders of information by scaling a matrix $A$ to its *principal submatrices.*

**Def.** *Let $A$ be an $n \times n$ matrix. A $k \times k$ submatrix of $A$ formed by deleting $n-k$ rows of $A$, and the same $n-k$ columns of $A$, is called principal submatrix of $A$. The determinant of a principal submatrix of $A$ is called a principal minor of $A$.*

Note that the definition does not specify which $n - k$ rows and columns to delete, only that their indices must be the same.

Let us introduce a $3 \times 3$ example.

For a general matrix $A_{3\times3}$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.3}$$

there are three *first order principal minors*:

- $\mid a_{11} \mid$ formed by deleting the last two rows and columns
- $\mid a_{22} \mid$ formed by deleting the first and third rows and columns
- $\mid a_{33} \mid$ formed by deleting the first two rows and columns

There are three *second order principal minors*:

- $\mid \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mid$ formed by deleting column 3 and row 3
- $\mid \begin{bmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{bmatrix} \mid$ formed by deleting column 2 and row 2
- $\mid \begin{bmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{bmatrix} \mid$ formed by deleting column 1 and row 1

There's one *third order principal minor*, namely $\mid A \mid$.

For the sake of completion, we also recall the following definition.

**Def.** *Let $A$ by an $n \times n$ matrix. The $k^{th}$ order principal sub-matrix of $A$ obtained by deleting the* ***last*** *$n - k$ rows and columns of $A$ is called the $k^{th}$ order* ***leading principal submatrix*** *of $A$, and its determinant is called the $k^th$* ***order leading principal minor*** *of $A$.*

An imporant property for the principal submatrices of a symmetric positive definite matrix is that *any $(n-k) \times (n-k)$ partition is also symmetric and positive definite.*

It is now clear that such properties can be used to reduce both a class scale matrix $\hat{\Sigma}_C$ and any $\Sigma_i$ matrix, in order to study its deviation from a class's estimated Wishart distribution derived from the deletion of one of its components (the features conatined in vector $X_i = (x_1, ..., x_p)$ from which the matrix $\Sigma_{i,p\times p}$ is computed).

Iterating this process over all the features, or in other terms analyzing all of the $(p-1) \times (p-1)$ principal submatrices of $\Sigma_i$ and $\hat{\Sigma}_C$, will allow us to assign a score to each feature, representing its weight in the decision for $\Sigma_i$ to be assigned to one class or another.

Note that for such an order of principal submatrices, the process will reduce the $\Sigma_{i,p\times p}$ matrix to a *score vector* of length $p$ for each element $i$ undergoing the classification.

Let us now introduce the following notation in order to define the score assigned for each of the $x_p$ features of the vector $X_i = (x_1..., x_p)$.

Let $\Sigma_j$ be a principal submatrix of order $(p-1)$, of the matrix $\Sigma$ computed on the observation of $X_i = (x_1, ..., x_p)$ for subject $i$, *obtained by the deletion of the $j^{th}$ row and the $j^{th}$ column*, with $1 \leq j \leq p$.

Let $\hat{\Sigma}_{Cj}$ be a principal submatrix of order $(p-1)$, of the matrix $\hat{\Sigma}_C$ computed for the class $C$ *obtained by the deletion of the $j^{th}$ row and the $j^{th}$ column*, with $1 \leq j \leq p$.

The score assigned to each feature of $X_i = (x_1, ..., x_p)$ is then given by eq.(3.4).

$$Score_j(C) = \Delta logP_{Wj}(C) = logP_W(\Sigma, n \mid \hat{\Sigma}_C, n) - logP_W(\Sigma_j, n \mid \hat{\Sigma}_{Cj}, n) \tag{3.4}$$

In other terms, each partition $\Sigma_j$ represents the matrix $\Sigma$ without the elements tied to feature $x_j$ (the elements in row $j$ and column $j$ of $\Sigma$). Computing the variation in terms of log-likehood between the estimated wishart distribution for the class and the estimated wishart distribution for the class without component $j$, allows us to gain informations about which feature weighs more on both subject $i$'s cassification and the general system structure.

Note that this kind of scoring is *class-dependent*. Computing this score vector with respect to all the classes $C_1..C_n$ of a given problem and performing some sort of score ratio will allow the subject $i$, after a suitable *training*, to be assigned to the most likely of the classes while retaining informations on which features are the most determinant, decision wise.

Let us introduce a *2-classes* example in order to show how this kind of result might be obtained.

Let $C_1$ and $C_2$ be the two classes of a given problem.

Let a set of $N$ matrices $\Sigma_i$ be a set of correlation matrices computed for $N$ subjects $i$ whose class is known.

Let $\hat{\Sigma}_{C1}$ and $\hat{\Sigma}_{C2}$ be the scale matrices computed as seen in eq.(3.1), used to estimate the Whishart distribution for each one of the two classes $C_1$ and $C_2$, and $\hat{\Sigma}_{C1j}$ and $\hat{\Sigma}_{C2j}$ their $(p-1)$ order partitions, as in eq.(3.4).

If from each matrix $\Sigma_i$ the score vector is computed as in eq.(3.4) with respect to each one of the two classes $C_1, C_2$, an *inter-class log-likelihood ratio*

*vector* can be obtained by assigning to each feature a score defined as follows:

$$Ratio_j = \Delta logP_{Wj}(C_1) - \Delta logP_{Wj}(C_2) \tag{3.5}$$

Training a classificator on a set of $N$ subjects whose classes are known, after each matrix $\Sigma_i$ (and as a consequence each feature vector $X_i$) has undergone the transformations defined in eq.(3.4) and (3.5), yields a significant improvement in performance for certain classes of problems, as it will be shown later.

A new subject will be, as a matter of fact, classified according to its *transformed ratio vector* given by eq.(3.5), thus simultanesously retaining information about its class's most significant features: *the score assigned to each feature is a measure of how much the deletion of said feature weighs, in terms of log-likelihood variation, on the decision to assign each matrix $\Sigma_i$ to one class or another.*

The entire process can be seen as a *feature transformation*, which leads to a *feature selection*, whose effect is, for certain types of problems, to enhance the classification performance.

### 3.2.3 Generalizing to $(p-n)$ Order Transformations

As seen in the last section, transforming all the $(p-1) \times (p-1)$ principal submatrices of $\Sigma_i$ by eq.(3.4), yields a vector of score of length $p$ for each element $i$.

Anyway, for any $n < p$, a number of principal submatrices of $\Sigma_i$ can be obtained.

These kind of submatrices can be used to gain informations about the weight of $n$ simoultaneously deleted features on the system structure and classification.

Let us introduce an example for $(p-2)$ order submatrices.

Let $\Sigma_{jk}$ be a principal submatrix of order $(p-2)$, of the matrix $\Sigma_i$ computed on the observation of $X_i = (x_1, ..., x_p)$ for subject $i$, *obtained by the deletion of the $j^{th}$ row and the $j^{th}$ column and the $k^{th}$ row and the $k^{th}$ column* , with $1 \leq j, k \leq p$.

Let $\hat{\Sigma}_{Cjk}$ be a principal submatrix of order $(p-2)$, of the matrix $\hat{\Sigma}_{Cjk}$ computed for the class $C$ *obtained by the deletion of the $j^{th}$ row and the $j^{th}$ column and the $k^{th}$ row and the $k^{th}$ column* , with $1 \leq j, k \leq p$.

Then, eq.(3.4) becomes:

$$Score_{jk}(C) = \Delta logP_{Wjk}(C) = logP_W(\Sigma, n \mid \hat{\Sigma}_C, n) - logP_W(\Sigma_{jk}, n \mid \hat{\Sigma}_{Cjk}, n) \tag{3.6}$$

in this case, a score is assigned to each coupling of the features $j, k$, and transformation (3.6) will yield not a vector, but a $p \times p$ matrix with diagonal elements equals to the scores obtained by (3.4), being the iteration with $j = k$ the coupling the $j^{th}$ feature with itself.

Non-diagonal elements represent the score of the coupling of feature $j$ with feature $k$.

## 3.3 Pipeline

In this section we discuss each step of the feature transformation and classification process.

Given the recursive nature of the method just described, a crucial issue concerning computational time is the strong dependence between it and the analyzed matrix size.

A rough visualization of the entity of such a dependence can be found in fig.(3.2)

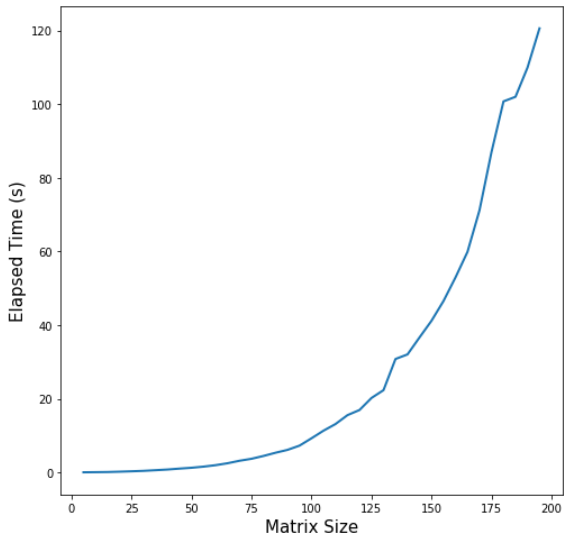


Figure 3.2: *Matrix Size dependence for* $(p-2)$ *order transformations.*

Iterating the $(p-1)$ order transformation described in eq.(3.4) over a large $N$ of observations of size $p \times p$ of a given database, while introducing some kind of *cross-validation* routine may lead to abysmal computational time-wise performances.

A possible solution to this problem is to introduce a high level of automatization for each step, followed by the indroduction of a highly parallelizable overall structure of the pipeline.

### 3.3.1 The Snakemake Environment

The main tool used to achieve such results is the *Snakemake Workflow Management System*, described in [15], a Python-based interface created to build reproducible and scalable data analyses and machine-learning routines.

To briefly some up the the advantages of using such tools and structures, the *Snakemake Workflow* can be described as rules that denote how to create output files from input files. The workflow is implied by dependencies between the rules that arise from one rule needing an output file of another as an input file [15].

A rule definition specifies :

- *a name*, used by the main rule instance *rule all* and main execution environment for identification
- *any number of input and output files*; tipically one rule's output is another rule input, linking the rules alla the way up to main rule instance.
- *either a shell command or Python code*; containing the creation of the output from the input

Input and output files may contain multiple named *wildcards*, whose values are inferred automatically from the files desired by the user.

To further clarify the role of the wildcards, let us introduce a brief example.

Let's say that our aim is to train a classifier over two classes of elements $C_1, C_2$. The training part of the database is then divided in two files, each one containing the name of its elements' class in the filename.

Setting a rule to load these files while expecting a wildcard tied to the class name in the filename, will allow the entire set of rules of the pipeline to be executed automatically for class $C_1$ and class $C_2$.

Considering this example, the real power of the parallelization capabilities offered by the Snakemake environment are quite clear.

With a simple syntax, looking at the example just proposed, each one of two *cores* of a server where our hypothetical pipeline is running can be set to work indipendently on each subset of data belonging to class $C_1$ or class $C_2$.

Building a pipeline whose rules are easily iterable over a set of different *wildcards* will lead to natural and efficient parallelization and automation.

### 3.3.2 The WISDoM Pipeline

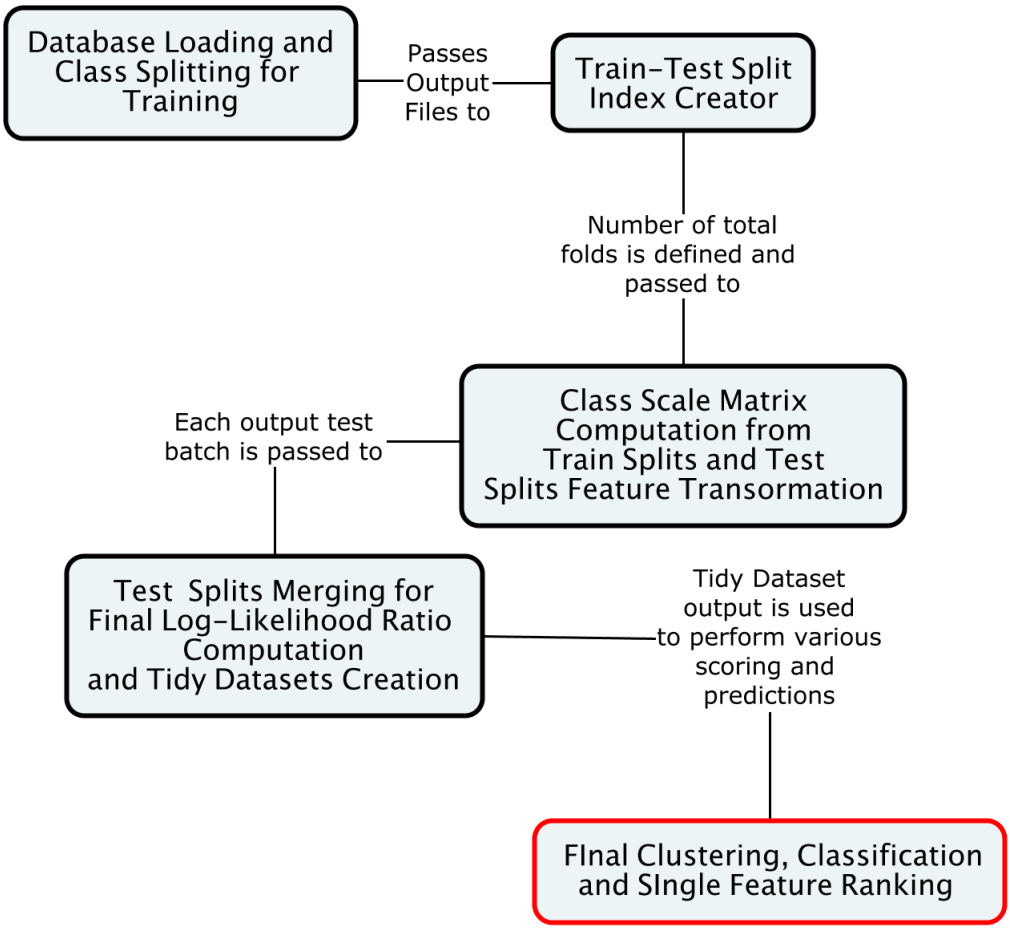


Figure 3.3: *General pipeline workflow.*

In figure (3.3) the main steps required by the WISDoM Multiple Order Classification are reported.

We will now go through each step in detail in order to show how *train and test splitting* are interpreted for the WISDoM pipeline.

- **Database loading and class splitting**
  *rules: case_wrap, seqs_store* ; in this section of the pipeline, data are loaded and divided into the classes defined by the wildcards and main rule instance's inputs. An info sheet containing the classification labels for each observation is needed as an *input* for this step.

  In order to achieve fast reading/writing perfomances for big data, the matrices are stored as the sequence of elements belonging to the upper triangle for each matrix (in *.hdf* format). Being symmetric, the entire matrix can be easily reconstructed when needed. At this step, the files containing observations for each class are created.

- **Train-Test split index creator**
  *rules: split_gen, tt_gen*; in this section of the pipeline, each file containing one class's observations is divided into train-test batches. The total number of train-test folds is defined by wildcards and main rule instance's inputs.

  First, each dataset is divided into sections, then each section is further splitted intoa number of user-specified train-test folds.

- **Class Scale Matrix computation and feature transformation**
  *rules: map_gen*; this is the core section of the pipeline, where the features are transformed according to eq.(3.4).

  Train-test split files for each class are passed as inputs; the train sets of each batch are used to compute the scale matrix $\hat{\Sigma}_C$ as in eq.(3.1). The estimated Wishart for the class is then computed and the features of each test-set element are transformed.

  In order to compute the Ratio described in eq.(3.5), the above process is repeated for each class with respect to each other. A map containing each transformed feature in term of *quantiles* is also created.

- **Test splits merging and tidy datasets creation**
  *rules: t_join, q_join*; in the final step of the pipeline, all of the transformed feature test batches are merged into tidy datasets. This type of data sturcture will allow an easy computation of the ratio in eq.(3.5) for each feature; furhtemore, once such dataset is obtained, evertything needed for the transformed observations to undergo any classification pipeline and/or model selection is ready.

A graphic representation of the plan of rules execution can be obtained by using a *directed acyclic graph* (DAG), as shown in figure (3.4).

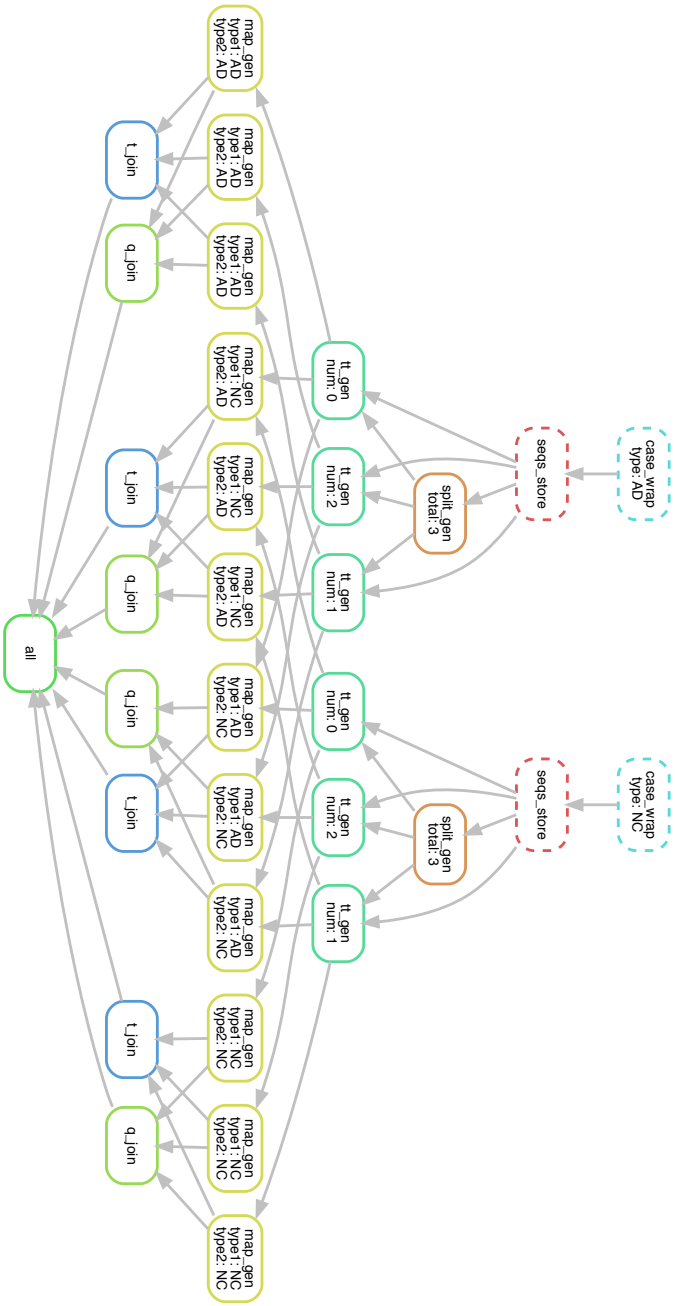


Figure 3.4: *Directed Acyclic Graph representation of the WISDoM pipeline for the observations of ADNI database. Here we have 2 type of subject, labeled AD and NC, undergoing a 3 fold train-test split.*

# Chapter 4

# Results of The WISDoM Multiple Order Classification

As stated in previous chapters, the chance to use the Wishart distribution to estimate covariance matrices distributions makes it extremely suitable for treating brain fMRI data and modelling problems.

Thus, to test WISDoM capabilities on feature selection and classification performances, two major databases of functional brain imaging data, the ADNI2 and ABIDE databases, have been explored and analyzed.

These two databases offer a good number of $p \times p$ *correlation matrices*; one for each of the subjects that have undergone the baseline observations, as well as detailed labelling of different diagnostical groups.

The ADNI2 database contains observations about *Alzheimer's disease* diagnoses and over-time conversions from *mild cognitive impairment*, while the ABIDE study and database focuses on diagnostical groups of *autism spectrum disorder*.

In both cases, data are fed to a WISDoM pipeline and a classification is attempted, while looking for the most significative features in classes' separation.

Furthermore, for comparison purposes, a class separation is attempted using a *Network-Growth* method, while observing the growth of random Wishart-generated networks in order to test the quality of the null-hypothesis.

## 4.1 The ADNI2 Database: Study and Results

ADNI, *Alzheimer's Disease Neuroimaging Initiative*, is an ongoing, longitudinal, multicenter study designed to develop clinical, imaging, genetic, and biochemical biomarkers for the early detection and tracking of Alzheimer's disease (AD). The ADNI study began in 2004 and included 400 subjects diagnosed with mild cognitive impairment (MCI), 200 subjects with early AD and 200 elderly control subjects.

The major goals of ADNI2 are to [16]:

- Determine the relationships among clinical, imaging, genetic, and biochemical biomarker characteristics of the entire spectrum of Alzheimer's Disease (AD), as the pathology evolves from normal aging through very mild symptoms, to mild cognitive impairment (MCI), to dementia.
- Inform the neuroscience of AD, identify diagnostic and prognostic markers, identify outcome measures that can be used in clinical trials, and help develop the most effective clinical trial scenarios.
- Develop improved methods which will lead to uniform standards for acquiring longitudinal multi-site MRI and PET data on patients with AD, MCI, and elderly controls.
- Perform longitudinal clinical, cognitive, MRI, PET (18F-AV-45 and FDG), and blood and CSF biomarker studies on 550 newly enrolled subjects in four diagnostic categories – cognitively normal (CN), early MCI (EMCI), late MCI (LMCI), and mild AD.
- Collect blood samples for DNA and RNA extraction. Newly enrolled subjects will also have samples collected for Cell Immortalization and APOE genotyping.
- Validate the clinical diagnoses and imaging and biomarker surrogates through neuropathological examination of ADNI1, GO and ADNI2 participants who come to autopsy.

A resume of how the clicnical data are collected for 54 months after the baseline is reported in fig. (4.1)

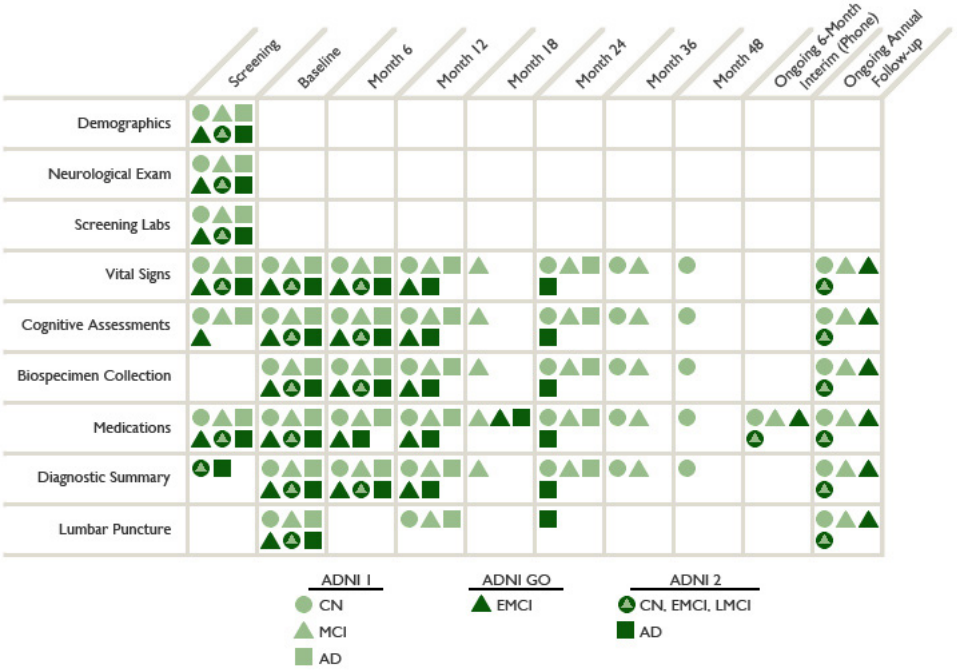


Figure 4.1: *Timeline of clinical data collection procedures. Source:* ***http://adni.loni.usc.edu/data-samples/clinical-data/***

### 4.1.1 Data Exploration and Selection for WISDoM Classification

As a preliminar analysis, the conversion rate of MCI to AD has been checked in order to find correspondence with literature. As in [17], [18] , we expect to find a maximum of the conversions' distribution over time at around 18 months.

This period of time is, as a matter of fact, the cut-off for what has been defined *long-term survival*, after which prediction results for conversion significantly lose stability [18].

As shown in figure (4.2), the maximum for the distribution stands at around 18 months.

Two different fits are reported for comparison in terms of log-likelihood.

Of the 403 total subjects available, only 232 were selected for the final WISDoM Classification run.

The main reason of such a selection is the fact that two distinct data group can be found inside the datasets.

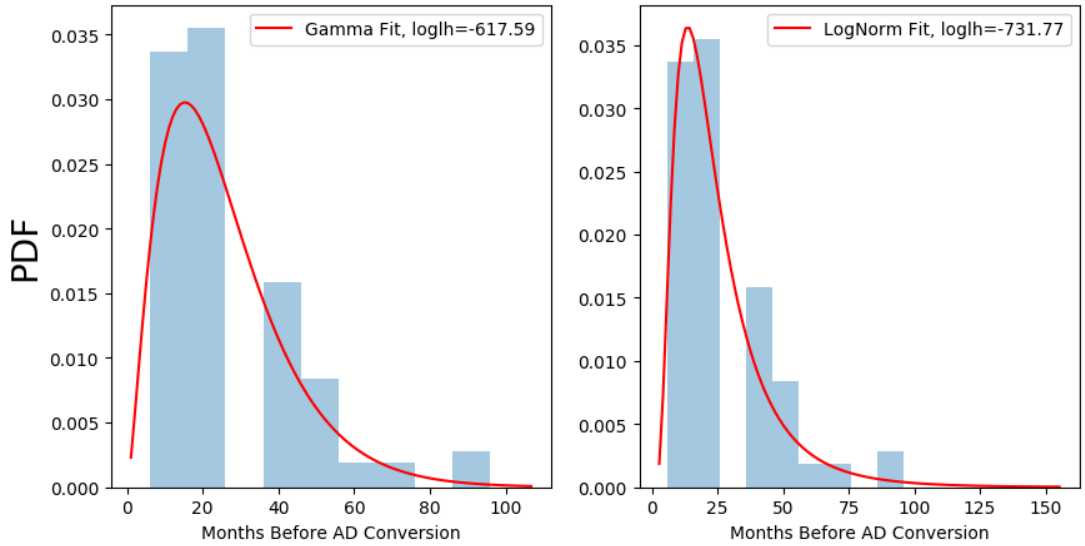


Figure 4.2: *Conversion distribution over time. As expected, we find a maximum at around 18 months. Two different fits (Gamma and Log-normal) are reported for comparison.*

As shown in fig.(4.3), two distinct peaks are obtained if a standard deviation distribution for the diagnostic groups is taken into account.

While no difference in distribution can be found amongst the three different diagnostic grups, the second peak can be a sign of two indipendent data normalization procedures in the dataset.

To support this hypothesis, if we look at the standard deviation distribution by gender in fig.(4.4), we can see that there are two different gender labelling (M-F and Male-Female) generating two distinct peaks.

This fact leads to the hypothesis that two distinct datasets, whos subjects have been examined with silghtly different procedures, might have been merged into a single one.

Given that the Male-Female labelling occurs for the first 98 entries, the remaining 305 entries have been chosen for the final run, being the biggest group with homogeneous data normalization.

Of these 305 entries, 7 have been discarded due to wrong or missing labelling.

Of the remaining entries, subjects of the *Mild Cognitive Impairment (MCI)* diganostic group have been excluded.

This choice has been made in order to train the classifier exclusively on the *Normal Control (NC)* group, and *Alzheimer's Disease (AD)* diagnostic groups.

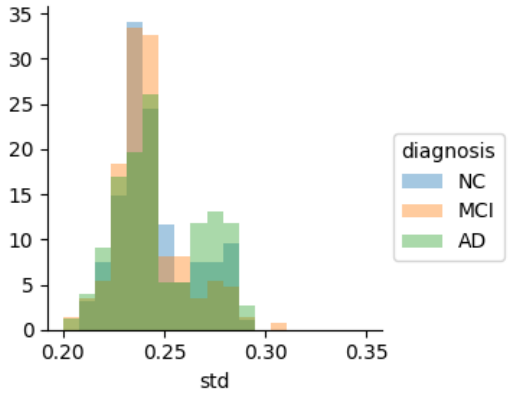


Figure 4.3: *Standard deviation for different diagnostic groups. As shown, different diagnostic groups yields no signficative difference while the second peak at* $\sim 0.27$ *might be due to different normalizations in data.*

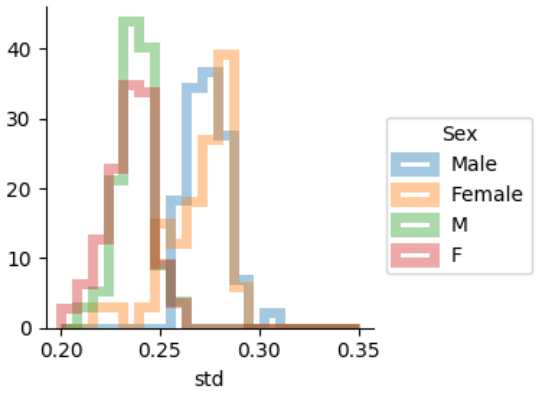


Figure 4.4: *Standard deviation distribution by gender. The two distinct peak are a sign of different normalization for data whose gender is labelled with M-F and Male-Female.*

In this way, the conversion component is excluded form the analysis, whose aim is to classify and select significative features for the NC and AD diagnostic groups.

Of the 232 subjects remaninig, the 63% belongs to diagnostic group AD and the remaining 37% belongs to diagnostic group NC.

### Data Structure and Preprocessing

Each subject's image has been preprocessed and divided into 549 *macrovoxels*, whose *topological correlation* has been computed with respect to each other, each of the 549 macrovoxels being defined over an ensemble of $3 \cdot 10^3$ voxels.

Thus, 232 $N \times N$ , $N = 549$ matrices have been used for training and calssification with the WISDoM classifier.

An example of such a corelation matrix can be seen in fig.(4.5).

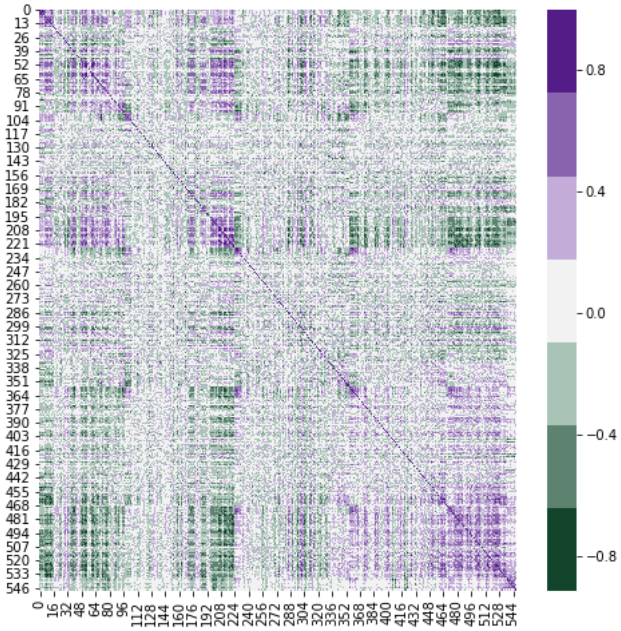


Figure 4.5: *Divergent heatmap representation of a subject's correlation matrix.*

## 4.1.2 Results

Several types of analyses and classifications have been conducted on the outputs of the WISDoM method's features transformations defined in eq.(3.2), eq.(3.4) and eq.(3.5).

First of all, a separation of the two classes AD and NC based on complete matrix distance (eq.(3.2)) is attempted.

Then, single feature distances as in eq.(3.4, 3.5) are computed for all the subjects and various clusterings and calssifications are attempted.

Lastly, a network-growth separation based on transformed features ranking is attempted for comparison.

### Complete Matrix Distance Separation

As a first approach to classification, a score based on simple log-likelihood distance is assigned to each subject's correlation matrix. In fig.(4.6) is reported the attempt to separate the two diagnostic groups AD and NC by computing the distance in terms of log-likelihood from each class's estimated Wishart distribution.

As seen before, one class's WIshart distribution can be estimate by computing the scale matrix as in eq.(3.1).

In this case the parameter *degrees of freedom* $n$ is given by the number of voxels belonging to each macrovoxel. Thus, we take $n = 3 \cdot 10^3$.

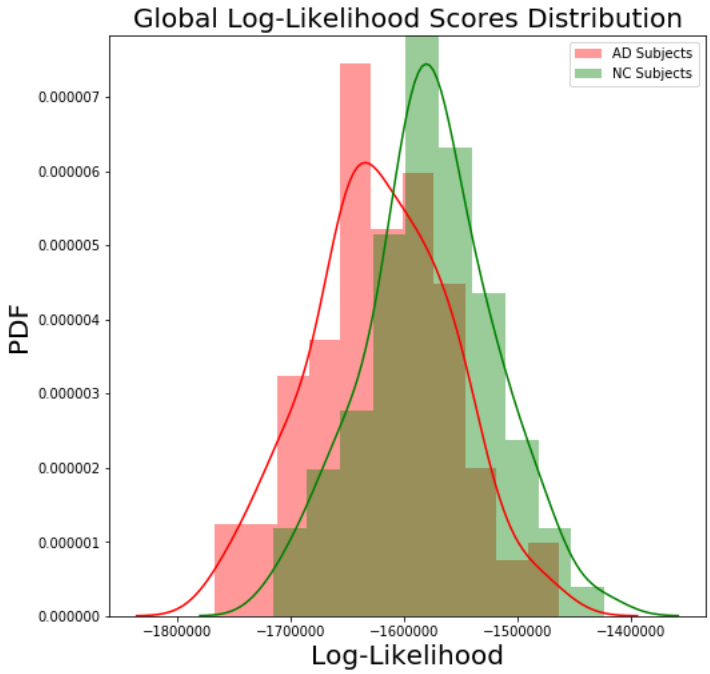


Figure 4.6: *Complete matrix log-likelihood score distribution, separation performance is abysmal.*

Then, a *complete matrix ratio score* as in eq.(3.2) is assigned to each matrix for classifficiation.

Results for this type of process are shown in fig.(4.7).

At this stage, it is clear that separation performances are inadequate and single feature analysis is needed.

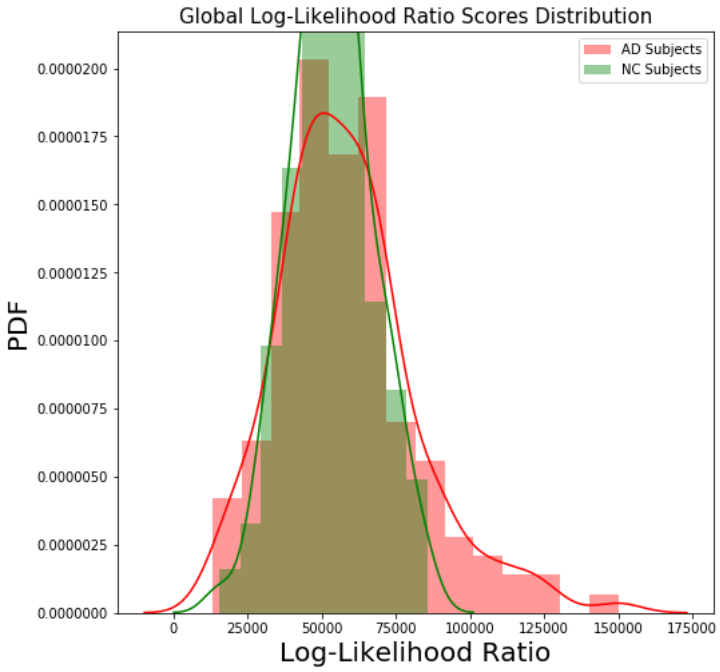


Figure 4.7: *Complete matrix log-likelihood ratio score distribution, separation performance is abysmal.*

### Single Feature Ratio Separation

For each one of the 232 subject, the $(p-1)$ order feature transformation is computed according to eq.(3.4); this is done thanks to the WISDoM Classification pipeline described by figure (3.3).

We set the pipeline to compute the estimated Wishart distribution for the classes from each train batch in a *10-fold cross validation process*; test elements' feature are then transformed and the score vector for order $(p-1)$ transformation is computed.

This leads to a $90\% - 10\%$ *train-test splitting* for each batch.

After the score vector is computed with respect to each class for each subject, scores defined by eq.(3.5) are computed and the vectors merged into a single tidy dataset.

Various classification processes are then attempted.

To compare separation performances after single feature analysis, in a way similar to what has been done for complete matrix distance separation described in fig.(4.6, 4.7), the distribution of single scores assigned for each subject is computed.

This is simply done via a sum of the single feature ratio score vector for each subject.

Results can be seen in fig.(4.8).

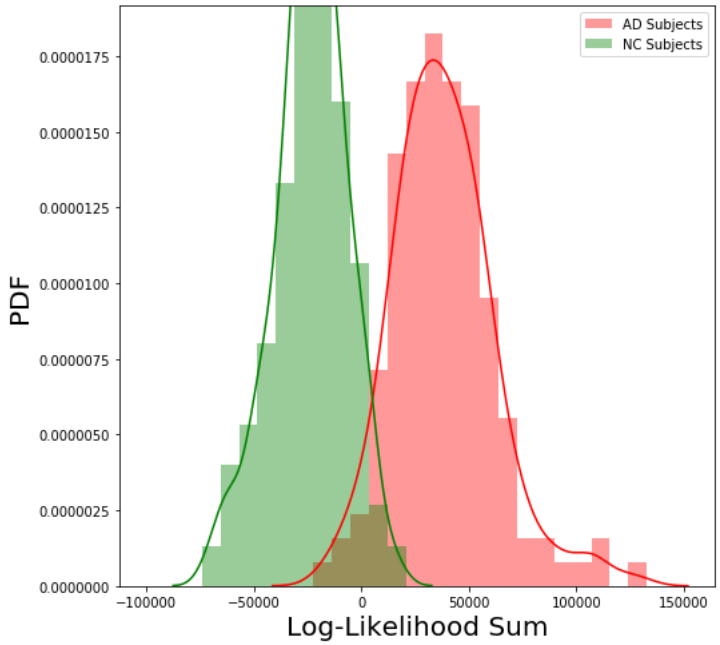


Figure 4.8: *Distribution of the sums of single feature ratio score vector for each subject. In this way, a vector of scores is reduced to a single score. Looking at the distributions we can see that classes are already well separated with this simple reduction.*

Comparing the results shown in fig.(4.8) with the results for complete matrix disatnce separation in fig.(4.6, 4.7), we can state that the $(p-1)$ order feature transformation and ratio score already give a significant separation perfomance enhancement at this stage.

### Clustering

In order to obtain information about which features are the most significative in classification, a *hierachical clustering*, described in [19], [20], is performed.

The metric used is a $L_1$ *City Block*, defined as $\sum_{j=1}^{k} \mid a_j - b_j \mid$ between two $k$ dimensional points $a, b$. In this way, the effect of a large difference in a single dimension is dampened (since the distances are not squared).

Such a clustermap, shown in fig.(4.9), offers a representation of the classification capabilities and weights of single features.

As a matter of facts, features on the left part of the map are seen as the most significative in clustering performance.

We can also state that a clustering made on $(p-1)$ order transformed features yields a good performance for the groups NC (in green) and AD (in red). Only a few subjects tends to be assigned to clusters far from their true classes' main clusters.

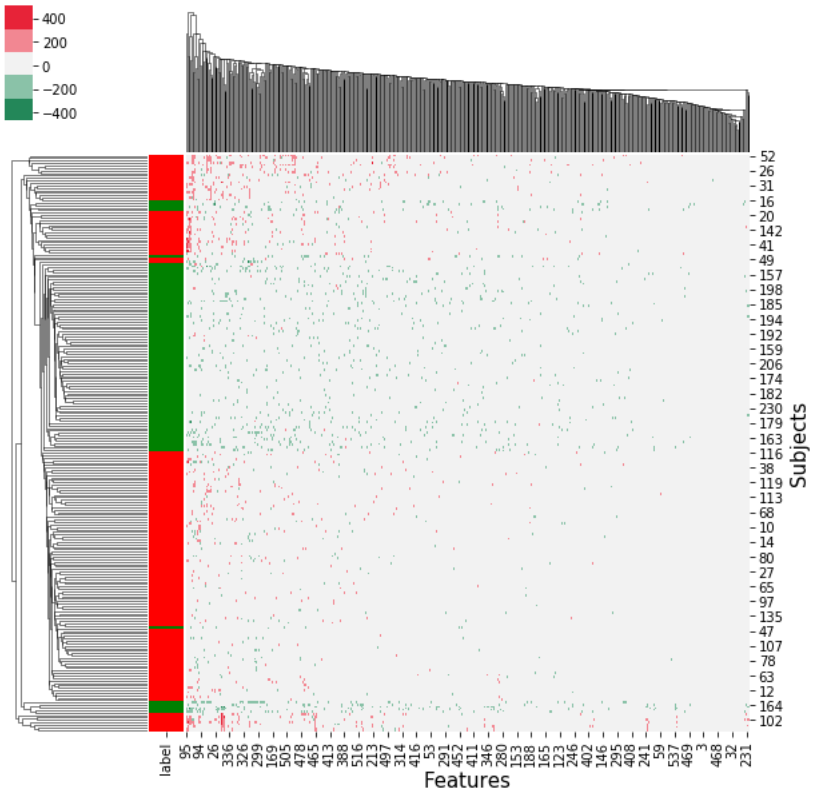


Figure 4.9: *Hierarchical Clustering visualization over an $L_1$ City Block Metric. Features on the left part of the map have the biggest influence in clustering decision. Color green is assigned for NC subjects and red for AD subjects.*

### SVM Classification and Single Feature Logistic Regression Ranking

Given the results obtained with scores' sum distributions and clustering, a classification with a *C-Support Vector Machine* has been attempted.

Using a linear kernel and taking the penalty parameter for error term $C = 1$, a *10 fold stratified cross-validation* is performed.

More details on the *SkLearn* module classifier used can be found at **http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.**

Note that the use of a stratified cross-validation means that the relative proportions of the classes are preserved for each train-test batch, as in [21].

Such a classification yields an *accuracy score* of 100%.

Again, given the results, a *single feature logistic regression classifier* has been tested, in order to observe each feature classification performances and produce a feature ranking.

This is done by computing a logistic regression classification for each one of the 549 transformed features of each subject and computing the average cross-validation *ROC AUC* score for each feature.

More details on the SKlearn Logistic Regression classifier used can be found at **scikit-learn.org/stable/modules/generated/sklearn.linear**/.

In general, the logistic regression model arises from the desire to model the posterior probabilities of the K classes via linear functions in $x$, while at the same time ensuring that they sum to one and remain in $[0, 1]$.

When $K = 2$, this model is especially simple, since there is only a single linear function. It is widely used in biostatistical applications where binary responses (two classes) occur quite frequently [21].

Thus, dealing with the present two-classes problem lets the a logistic regression become the natural choice as a classifier.

At this stage, a logistic regression classification is attempted using a single feature for each iteration.

Then a 10-folds cross validation is computed and the average ROC AUC score is assigned to each feature, as an indicator of classification performance.

A plot of the classification capabilities, in terms of ROC AUC score, over features' ranking is reported in fig.(4.10).

To grasp the meaning of the *Receiving Operating Characteristic* score means, we can think as follows.

A ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) of a classifier at various threshold settings.

It is thus a plot of the *sensitivity* (or *probability of detection*) as a function of the *fall-out* (or *probability of false alarm*).

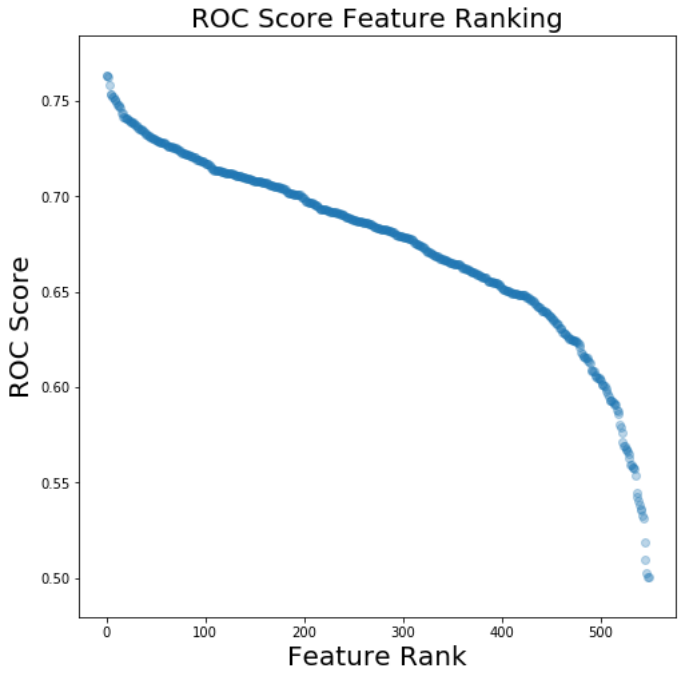


Figure 4.10: *ROC score over ranking. Note that by definition a ROC score of 0.5 means a completely random classification. We can see that classification capabilities rapidly decrease after the first 300 rankings.*

Examples of ROC curves are reported in fig.(4.11).

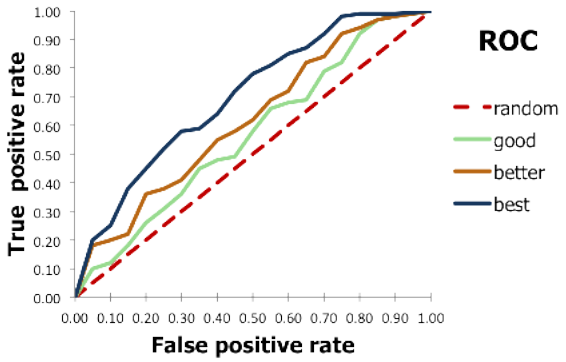


Figure 4.11: *Sample ROC curves. A completely random classifier will migrate to the point* $(0.5, 0.5)$ *while a perfect classification would be located at* $(0, 1)$. Credits: **https://docs.eyesopen.com/toolkits/cookbook/python/plotting/roc.html**

If the *Area Under the Curve* is computed when using normalized units, we obtain a value tied to the informative power of a classifier, with a completely uninformative classifier (i.e a classifier based on completely random choices) yielding a value of 0.5 [22].

Thus, computing a regression for each feature and the relative ROC AUC score will tell how informative a classifier based on that single feature is.

To visualize he informative content of the highest ranking features and take a look at their classification capabilities, we attempted to separate the two classes in the first two highest ranked features' space.

Beside scattering each class's elements, a linear regression for each class is plotted.

Results are shown in fig.(4.12).

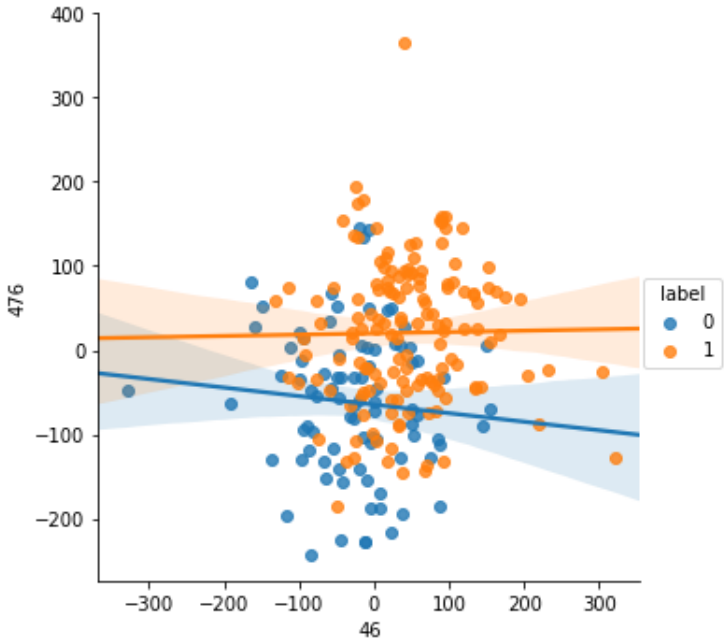


Figure 4.12: *Scatterplot in the first two highest ranked features' space. While the elements of each class are far from being completely separate, we can see that the regressions computed for each one of the classes are not overlapping, even if a confidence interval is plotted. Label 1 is assigned to class AD while label 0 is assigned to class NC.*

While the separation is obviously not comparable to results obtained and shown by fig.(4.8) or fig.(4.9), we can state that reducing the classification to the first two highest features (in other terms, reducing the problem to two

dimensions) doesn't cause the classification to be completely uninformative.

As a matter of fact, we can see that the simple regressions shown in fig.(4.12) already mantain a disctinct trend.

### Network Growth Separation for Non-Transformed Features

In order to emphasize the importance of the features' transformation defined in eqs.(3.4, 3.5) in classification performance enhancement, we attempted to obtain some separation results on non.transformed features via a *threshold-based network growth* process.

First, all matrices have been standardized. Then a ranking of absolute correlation values is made, in order to obtain a *sorted edges list* upon which the networks' growth is based and nodes are created.

At this point, a network is observed in terms of *number of nodes*, *number of connected components*, *size of the biggest connected component* while adding an edge for each iterations.

In this way, the strongest connections in terms of correlations between features are the first to generate nodes in the network for each iterations. The goal is to establish if Alzheimer's diagnosed subjects network of correlations grows at a slower rate than normal control subjects' networks.

Each network's growth is observed for a range of $3 \cdot 10^3$ edges.

Besides observing networks for the two diagnostical groups AD and NC, growth of *Erdos-Reniy* random networks and *Random Wishart sampled* networks is also reported for comaprison. Results are shown in fig.(4.13).

The Wishart random sampling networks are generated using the *average of all ADNI2's subjects matrices as the scale matrix.*

By looking at fig.(4.13), we see that no significative separation is shown for the two classes.

On the other side, we can see that the growth trend of Wishart-generated random networks is comparable with that of real AD and NC observations, thus validating the model used for the generation of the estimated distributions.

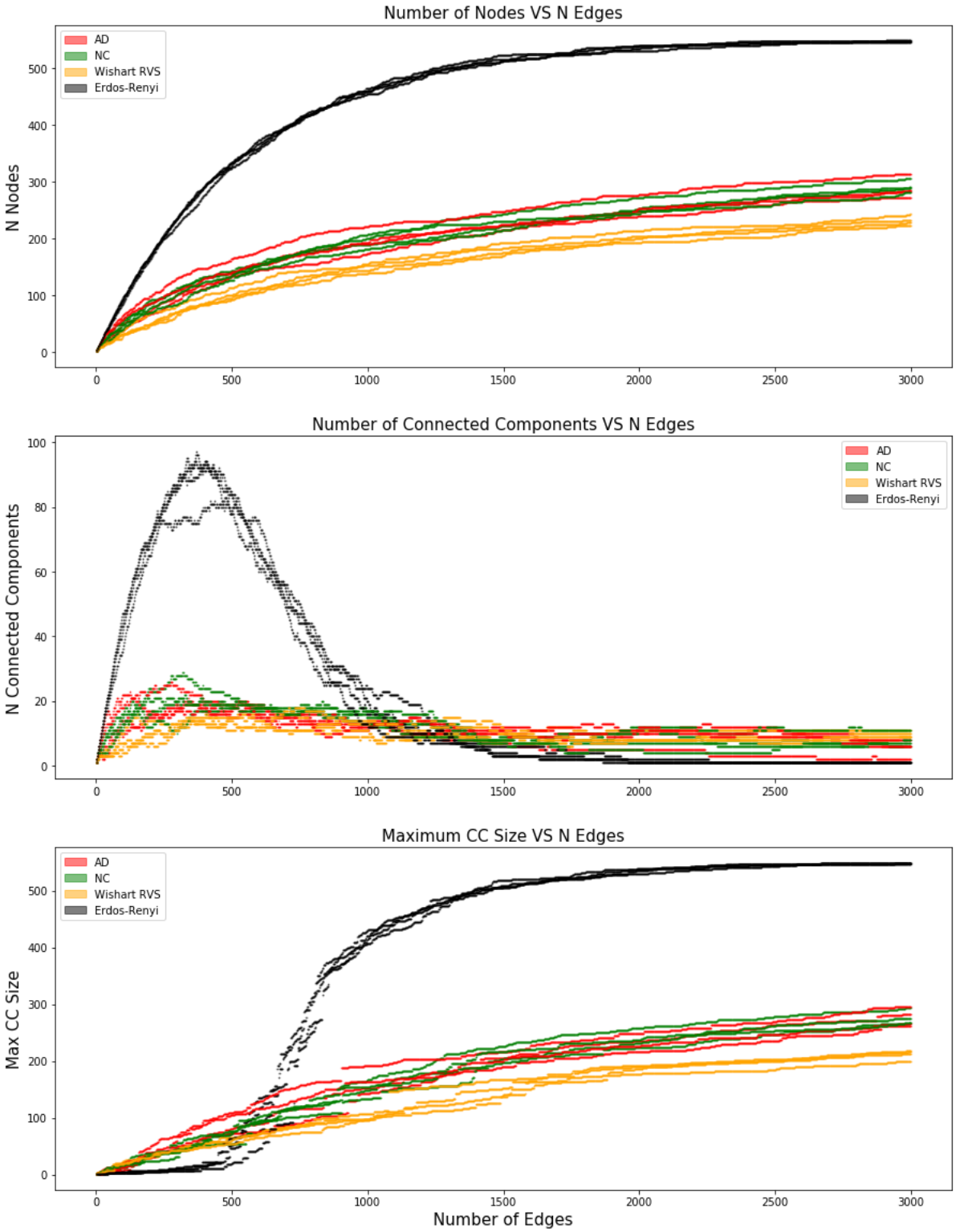


Figure 4.13: *Networks growth observed in terms of Number of Nodes, Number of Connected Components and Size of largest Connected components. 4 subjects are plotted for each category in order to give an idea of variability. While separation performances are abysmal, we can see that the Wishart null hypothesis is well suited, as its trend is comparable with AD and NC subjects' trend.*

To get a more accurate idea of inter-class growth's average behaviour, a *Lowess Regression* [23] as been computed on *50* subjects for each of the diagnostic group AD and NC.

As shown in fig.(4.14), there is no significative separation between the two classes.

An issue with fig.(4.14) is that the tool used to compute the lowess did not allow the visualization of a confidence interval; we can see however that regression's lines are overlapping even without a confidence inerval.

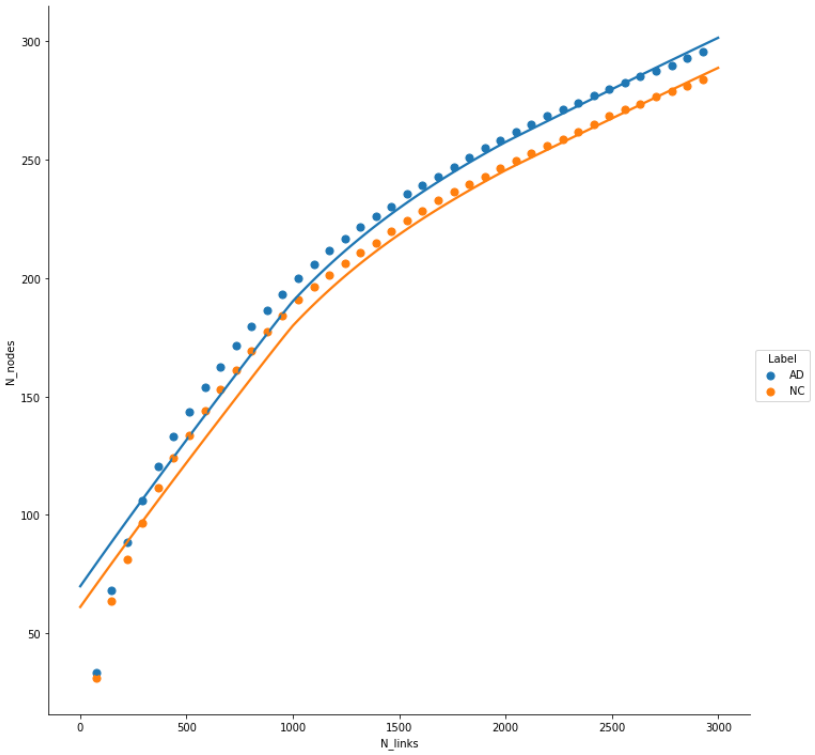


Figure 4.14: *Lowess regression on 50 subjects-per-class growth. No significant separation is yielded, given that for the tool used, a confidence interval for the plot could not be visualized. The x axis has been binned into 40 values to enhance visualization.*

# Conclusions

# Bibliography

[1] Randy L. Buckner, Jessica R. Adrews-Hannaand Daniel L. Schacter

*The Brain's Default Network Anatomy, Function, and Relevance to Disease*

New York Academy of Sciences 2008

[2] Raichle M. E., MacLeod A. M., Snyder A. Z., Powers W. J., Gusnard D. A., and Shulman G. L.

*A Default Mode of Brain Function*

Proceedings of the National Academy of Sciences, 98 (2), pp. 676-682, 2001

[3] Fox M. D., Snyder A. Z., Vincent J. L., Corbetta M., Van Essen D. C., and Raichle M. E.

*The Human Brain is Intrinsically Organized into Dynamic, Anticorrelated Functional Networks*

Proceedings of the National Academy of Sciences, 102, pp. 9673-9678, 2005

[4] M. P. Van Den Heuvel and H. E. H. Pol.

*Exploring the Brain Network: a Review on Resting-State fMRI Functional Connectivity*

European Neuropsychopharmacology, 20(8):519–534, 2010

[5] E. Bullmore and O. Sporns

*Complex brain networks: graph theoretical analysis of structural and functional systems*

Nature Reviews Neuroscience, 10(3):186–198, 2009.

[6] Xiaohu Zhao , Yong Liu , Xiangbin Wang, Bing Liu, Qian Xi, Qihao Guo, Hong Jiang, Tianzi Jiang , Peijun Wang

*Disrupted Small-World Brain Networks in Moderate Alzheimer's Disease: A Resting-State fMRI Study*

March 23, 2012

DOi: https://doi.org/10.1371/journal.pone.0033540

[7] N. K. Logothetis

*What we Can Do and What we Cannot Do with fMRI*

Nature, 453(7197):869–878, 2008.

[8] A. C. Evans, D. L. Collins, S. Mills, E. Brown, R. Kelly, and T. M. Peters.

*3d Statistical Neuroanatomical Models from 305 MRI Volumes*

In Nuclear Science Symposium and Medical Imaging Conference, 1993., 1993 IEEE Conference Record.,

pages 1813–1817. IEEE, 1993.

[9] Friston

*Functional and Effective Connectivity: A Review*

Brain Connectivity, 1, 13-36, 2011

[10] Hardle, Wolfgang and Leopold Simar

*Applied Multivariate Statistical Analysis*

Heidelberg: Springer Berlin Heidelberg, 2012

[11] Anderson, T. W.

*An Introduction to Multivariate Statistical Analysis*

New York: John Wiley and Sons, 2003

[12] Han Liu and Larry Wasserman

*Statistical Machine Learning*

Pittsburgh: CMU University, 2014

[13] Murphy, Kevin P.

*Conjugate Bayesian Analysis of the Gaussian Distribution*

Vancouver: University of British Columbia, 2007

[14] W.B. Smith and R.R. Hocking

*Algorithm AS 53: Wishart Variate Generator*

Applied Statistics, vol. 21, pp. 341-345, 1972.

[15] Köster, Johannes and Rahmann, Sven

*Snakemake - A scalable bioinformatics workflow engine*

Bioinformatics, Volume 28, Issue 19, Pages 2520–2522,1 October 2012

DOI: https://doi.org/10.1093/bioinformatics/bts480

[16] ADNI Commitee

*Alzheimer's Disease Neuroimaging Initiative - Procedures Manual*

http://adni.loni.usc.edu/, January 2016

[17] Ke Liu, Kewei Chen, Li Yao and Xiaojuan Guo

*Prediction of Mild Cognitive Impairment Conversion Using a Combination of Independent Component Analysis and the Cox Model*

Frontiers in Human Neuroscience, 06 February 2017

DOI: https://doi.org/10.3389/fnhum.2017.00033

[18] Wei R., Li C., Fogelson N., Li L.

*Prediction of Conversion from Mild Cognitive Impairment to Alzheimer's Disease Using MRI and Structural Network Features*

Frontiers in Aging Neuroscience 19 April 2017

DOI: 10.3389/fnagi.2016.00076

[19] Ziv Bar-Joseph, David K. Gifford, Tommi S. Jaakkola,

*Fast Optimal Leaf Ordering for Hierarchical Clustering*

Bioinformatics, 1 June 2001

DOI: https://doi.org/10.1093/bioinformatics/17.suppl_1.S22

[20] Daniel Mullner

*Modern Hierarchical, Agglomerative Clustering Algorithms*

12 September 2011

DOI: https://arxiv.org/abs/1109.2378v1

[21] Trevor Hastie, Robert Tibshirani, Jerome Friedman

*The Elements of Statistical Learning*

Springer ISBN 978-0-387-84858-7

[22] Hand, David J. and Till, Robert J.

*A simple generalization of the area under the ROC curve for multiple class classification problems,*

Machine Learning, 45, 2001

[23] Cleveland, W.S.

*Robust Locally Weighted Regression and Smoothing Scatterplots*

Journal of the American Statistical Association 74 (368): 829-836, 1979