Manipulation, Preprocessing, and Statistical Analysis of MRI, fMRI, and GWAS Datasets

Abtin Mogharabin

Anyone familiar with neurological and genetic datasets understands that preprocessing them is far from straightforward. This document offers a comprehensive review of various techniques and methods for preprocessing and manipulating neurological MRI and fMRI, as well as genetic GAWS datasets. We will explore a range of methods for preprocessing and analyzing these types of data, with an emphasis on the intuition and logic behind each method while also touching on some algorithmic details. It's worth noting that many of the technical specifics in each section can be bypassed, thanks to numerous software tools that automate the process. Nonetheless, I aim to provide some detail on the algorithms, particularly in the statistical analysis section, for a comprehensive understanding of these methods.

Table of contents:

*Only the red-colored sections are available in this version

Note 1: I suggest you to install the FSL toolbox on your computer. This is one of the most used MRI/fMRI preprocessing tools (developed by the University of Oxford). It contains a very wide range of predefined options to automatically execute most data manipulation techniques that I am going to explain. Having this toolbox at hand and testing it on MRI/fMRI data as you read the following steps would help you understand these concepts.

Note 2: Also, note that there are also many tools that are specialized to deal with each of the following steps I will cover. For each of the steps we take during the project, we should check all available tools in each step and use the one with the most efficient algorithm (so, we will not use FSL in most of the steps). For example, SynthStrip is a tool solely designed to deal with the Brain Extraction stage. And, FastSurfer is a very fast and efficient deep learning-based segmentation tool.

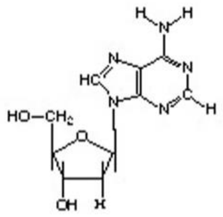
GWAS: Preprocessing and Statistical Analysis

1. Genome-Wide Association Studies (GWAS)

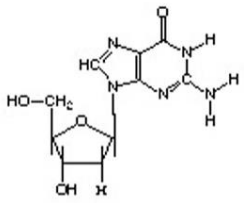
To summarize, Genome-wide association studies (GWAS) are a method of identifying genetic variants statistically associated with a biological trait of interest. Before we can understand what that means, we need to learn what is meant by ‘genetic variants’ and what is meant by ‘traits’:

1.1. Genetic Variation and Traits

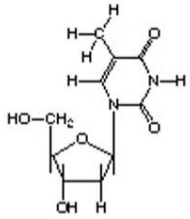
Human DNA, the molecule that carries genetic information for the development and functioning of humans, consists of only four nucleotides: adenine (A), cytosine (C), guanine (G), and thymine (T). These bases are formed by the structures shown below, and they pair in specific shapes: A pairs with T, and G pairs with C.



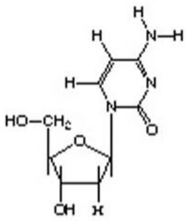
Adenine



Guanosine



Thymine



Cytosine

To have an intuitive example of gene, assume (four) English alphabets to be nucleotides, then we can take each (three letter long) word generated by these alphabets to be a word and every sentence constructed by these words would be a protein. Then, each paragraph constructed by these sentences would be a gene. These paragraphs would then get together to generate a book chapter (genes create chromosomes), and book chapters would come together and result in a book, that delivers the instructions on biological processes to the body of each human.

To keep it short, the 4 A, C, G, & T bases combine in groups of three, constructing three nucleotide long structures for their corresponding amino acids and codons. Then, these amino acids combine and form a protein. And, multiple proteins stick to each other to generate the structure of a gene.

A gene can be understood as a sequence of DNA instructions that direct cells to produce proteins, the building blocks of the body. Genes consist of exons, which encode amino acids, and introns, non-coding regions of DNA. These genetic instructions are organized into chromosomes, with genes resembling paragraphs within these chromosomes. Surrounding the genes are extensive stretches of non-coding DNA, containing sequences called transcription factors that regulate gene expression. Together, these genetic components form a complex system that governs various biological processes in the body.

* The human genome comprises 23 chromosomes and 3.3 billion letters.

1.2. Linkage Disequilibrium

To get to Linkage Disequilibrium, we first need to know what Genotyping is. To summarize, Genotyping is a process used in GWAS to identify specific nucleotides at various positions within an individual's genome. Unlike whole-genome sequencing, which deciphers every nucleotide in the genome, genotyping focuses on particular 'tag SNPs' to make the process more cost-effective. This approach is particularly useful in GWAS due to the importance of large sample sizes for detecting associations with complex traits. The rationale behind this method is that it provides significant information to impute, with high accuracy, the alleles at other genomic positions by using the principle of linkage disequilibrium.

Linkage disequilibrium (LD) is a fundamental concept in genomics that refers to the non-random association of alleles at two or more loci. It plays a crucial role in genotyping by allowing researchers to predict the presence of certain genetic variants based on the known variants within the same haplotype block. LD arises due to the manner in which chromosomes are inherited and recombined through processes like crossing over during meiosis. Dexplaiuring meiosis, chromosomes can exchange DNA segments, leading to new combinations of alleles. However, this recombination is not entirely random, and certain segments (haplotype blocks) are more likely to be inherited together, maintaining their association over generations.

We should note that although the predictive power of LD facilitates the identification of genomic regions associated with certain traits, it also obscures the clarity of pinpointing the exact causal genetic variants. The dilemma arises as GWAS often identifies multiple SNPs in tight LD, blurring the lines in distinguishing the variant directly influencing the trait. This complexity necessitates a deeper dive into the biological mechanisms at play, guiding researchers to hypothesize and investigate the role of genes within these identified regions, thereby unraveling the intricate web of genetic influence on complex traits.

One main reason for such problems is Crossing Over. Crossing over is a critical genetic process that occurs during meiosis, the type of cell division responsible for producing gametes (sperm and egg cells). During meiosis, homologous chromosomes—pairs of chromosomes, one inherited from each parent—align closely together. At this point, they can exchange equivalent segments of DNA in a process known as crossing over. This exchange results in chromosomes that are a mix of alleles (variations of a gene) from both parents, contributing significantly to genetic diversity in offspring. Unfortunately, although this process is vital for generating genetic diversity but also creates challenges in pinpointing the exact genetic variants responsible for specific traits. (Because it complicates the identification of the causal variants responsible for traits. Often, GWAS identifies multiple SNPs in LD with each other, making it difficult to determine which SNP is directly influencing the trait of interest.)

But despite the possible problems we just mentioned, genotyping and LD are still considered a critical aspect of genetics research. By identifying regions of the genome associated with specific traits, researchers can focus their efforts on understanding the biological processes involved. This might involve investigating the genes located within or near these regions and hypothesizing how variations in these genes might influence the traits, based on existing knowledge of gene functions.

Now, we cover some of the key statistical methods used in GWAS data processing and analysis:

1.3. P-values and Multiple Testing

P-values: P-values are a fundamental statistical measure used in GWAS to assess the strength of the evidence supporting an association between a genetic variant and a trait. A P-value quantifies the probability that an observed association (or one more extreme) would occur by chance if there were actually no true association between the variant and the trait. The closer the P-value is to 0, the stronger the evidence for a real association. Conversely, a P-value closer to 1 suggests weaker evidence. For instance, a P-value of 0.01 implies a 1% chance that the observed association could be a false positive, occurring purely by chance.

Multiple Testing: GWAS involves testing associations between potentially millions of genetic variants and traits, which introduces the multiple testing problem. With each additional test, the chance of encountering a false positive (a spurious association deemed significant purely by chance) increases. Historically, a P-value threshold of 0.05 was commonly accepted in statistical analyses, implying a 5%

chance of a false positive. However, given the massive number of tests conducted in GWAS, this threshold would result in an unacceptably high number of false positives. So we need to deal with this problem suitably. One common approach to address the multiple testing problem in GWAS is to employ a more stringent genome-wide significance threshold, typically set at $p < 5 \times 10\text{^}(-8)$. This threshold is determined by dividing the conventional significance level of 0.05 by the estimated number of independent common SNPs across the human genome, thus drastically reducing the likelihood of false positives.

However, it should be noted that although this adjustment increases the reliability of the results, it also has some disadvantages that we need to try dealing with them:

- This approach necessitates extremely large sample sizes to detect real, albeit small, genetic effects on complex traits. To summarize, correcting for multiple testing mitigates the risk of false positives but also affects the study's power to detect genuine associations, especially for complex traits where genetic effects are often subtle. Large sample sizes are crucial to overcome this limitation, enabling researchers to identify true genetic associations at the genome-wide significance level. A suitable data size usually contains thousands of patients.

1.4. Linear and Logistic Regression

Regression analysis in GWAS is a statistical method used to model the relationship between a dependent variable (phenotype or trait) and one or more independent variables (genetic variants, often SNPs). The goal is to quantify how much a change in the genetic variant is associated with a change in the trait. This involves constructing a statistical model that best explains the observed variability in the phenotype based on the genotype data.

Linear regression is applied to continuous traits, where the relationship between the genetic variant and the phenotype can be approximated by a straight line. Mathematically, the model can be expressed as:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Here, Y represents the trait value, X is the number of minor alleles (0, 1, or 2), β_0 is the intercept (the expected mean value of Y when X=0), β_1 is the slope (indicating the effect size of the genetic variant), and ϵ is the error term (capturing all other factors affecting Y besides X). The slope β_1 is crucial as it quantifies the change in the trait for each additional copy of the minor allele.

The coefficient of determination, R^2, measures the proportion of variance in the dependent variable that is predictable from the independent variable. In the context of GWAS, it quantifies how much of the variation in the trait can be explained by the genetic variant. R^2 is calculated as:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Where SS_{res} is the sum of squares of residuals (variation unexplained by the model), and SS_{tot} is the total sum of squares (total variation in the dataset). A higher R^2 value indicates a better fit of the model to the data, meaning the genetic variant has a more substantial role in determining the trait. Then, p-values are calculated for the slope coefficients to test the null hypothesis that the coefficient is equal to zero (no effect). The P-value is obtained from the t-statistic (or F-statistic for multiple regression), which compares the estimated coefficient to its standard error.

For binary traits, logistic regression models the probability that a trait is present as a function of the genetic variants. The logistic model is:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X$$

Here, p is the probability of the trait being present, log(1−p / p) is the log-odds of the trait, and X represents the genetic variant. The coefficients β_0 and β_1 are estimated using maximum likelihood estimation. The effect size β_1 indicates how the log-odds of the trait change with each additional copy of the minor allele. Here, the coefficient β_1 quantifies the change in the log-odds of the trait for a one-unit change in the predictor (the genetic variant). To interpret the effect size in terms of odds ratio (OR), we exponentiate β_1:

$$OR = e^{\beta_1}$$

An OR > 1 indicates an increased likelihood of the trait with each additional minor allele, while OR < 1 indicates a decreased likelihood. R^2 in logistic regression can be defined in several ways, with one common approach being the Nagelkerke R^2, which adjusts the Cox & Snell R^2 to vary between 0 and

1, making it analogous to the R^2 in linear regression. P-values in logistic regression are also derived from the likelihood ratio test, comparing the likelihood of the model with and without the predictor.

1.5. Polygenic Scores

Polygenic scores (PGS) are a quantitative representation of an individual's genetic predisposition to a certain trait or disease. They are derived from the cumulative effect of multiple genetic variants across the genome, each contributing a small effect to the overall trait or disease risk. PGS are calculated by summing the product of the number of effects alleles an individual has and the effect size (usually the beta coefficient from GWAS) of each SNP, effectively providing a score that predicts an individual's genetic liability for complex traits or diseases.

The calculation of a polygenic score involves summing up the effects of numerous genetic variants across the genome. For each variant, the number of effect alleles (0, 1, or 2) an individual has is multiplied by the estimated effect size of that allele (often obtained from GWAS results) on the trait. The sum of these products gives the individual's polygenic score:

$$\mathrm{PGS} = \sum_{i=1}^{n} (\beta_i \times \mathrm{Allele\ Count}_i)$$

Here, β_i is the effect size (beta coefficient) for the i-th SNP, and Allele Count_i is the number of effects alleles the individual has for the i-th SNP.

It should be noted that unlike monogenic traits (traits produced by the effect of a gene or an allele such as: the color of the animals, dwarfism, extreme muscularity, malformations, or severe health disturbances) that are influenced by a single gene, such as lactose tolerance or certain hereditary diseases, most complex traits like height, intelligence, or susceptibility to common diseases, are polygenic. Meaning they're influenced by many genetic variants each contributing a small effect. Polygenic scores aggregate these small effects across many SNPs to provide insights into an individual's genetic predisposition to these complex traits.

Here, the coefficient of determination (R^2) quantifies how well the polygenic score can predict the trait's variance in a population. It is essentially a measure of the predictive accuracy of the polygenic score. A higher R^2 indicates that the polygenic score explains a larger portion of the variance in the trait. For instance, an R^2 value of 0.11 in a polygenic score for IQ would imply that 11% of the variation in IQ across the population can be attributed to the genetic variants included in the score.

In general, PGS have various applications, including risk assessment for diseases, informing screening strategies, and understanding the biological basis of diseases with high comorbidity. They also play an important role in personalized medicine, helping to predict drug response and tailoring treatments to individual genetic profiles. In psychological research, PGS can help disentangle the effects of genetics (nature) from environmental factors (nurture) in influencing complex behaviors and traits. Overall, it should be noted that polygenic scores can be used to predict both continuous traits (like height or IQ) and dichotomous traits (like the presence or absence of a disease). The interpretation of the score differs based on the trait type; for continuous traits, the score might predict the level of the trait, while for dichotomous traits, it might predict the likelihood or risk of the trait occurring.

But I also mention that there are also some issues with PGS. One significant issue is the "missing heritability" problem, where the genetic variance explained by current GWAS and PGS falls short of the heritability estimates derived from family and twin studies. Additionally, the predictive power of polygenic scores can vary widely across different populations, and their utility may be limited by the genetic architecture of the trait and the sample size of the GWAS from which the score is derived.

Now that we have covered the key aspects of GWAS data analysis, I will provide some details on SNPs:

2. Single nucleotide polymorphism (SNP)

SNPs represent the most common type of genetic variation among individuals. A SNP occurs when a single nucleotide (A, T, C, or G) in the genome differs between members of a species or paired chromosomes in an individual. These variations can influence how individuals respond to diseases, bacteria, viruses, chemicals, drugs, and other substances.

SNPs are found in approximately every 300 nucleotides along the human genome, which means there are about 10 million SNPs in the human genome. Most SNPs have no effect on health or development, but some can act as biological markers, helping scientists locate genes associated with disease. When SNPs occur within a gene or in a regulatory region near a gene, they can play a more direct role in disease by affecting the gene’s function.

In general, SNP characteristics are 2:

- Bi-allelic Nature: Most SNPs are bi-allelic, meaning they have two possible nucleotide variations at a specific locus. This simplicity makes them highly amenable to genetic analysis and genotyping.
- Origin and Evolution: SNPs can arise due to errors in DNA replication or repair mechanisms, and once established in a population, they are subject to evolutionary forces such as mutation, drift, selection, and migration, which shape their distribution and frequency.

Due to their abundance and variability, SNPs serve as excellent markers for genetic fingerprinting, allowing researchers to differentiate between individuals or populations on a genetic level. This uniqueness in SNP patterns across individuals contributes to genetic diversity and can be used in various genetic studies, including population genetics, forensic analysis, and personal genomics.

2.1. SNP genotype data files

We cover two predominant data formats in genomics: the binary PLINK format (.bed, .bim, .fam files) and the classical plaintext format (.ped, .map files). Each format caters to different analytical needs, with the binary format being more efficient for computational analysis due to its compact nature.

PLINK Binary Format: This format consists of three files (.bed, .bim, .fam) that work together to store genotype data efficiently. It's designed for high-performance computational analyses, enabling rapid processing of large-scale genomic datasets.

- .bed File: Contains the binary representation of genotype data, indicating the presence of specific alleles at SNP positions across individuals. Its binary nature makes it compact and fast to process but not human-readable.
- .bim File: Acts as a detailed map for SNPs included in the .bed file. It lists each SNP's chromosome, identifier, genetic distance, and physical position, providing essential context for interpreting genotype data.
- .fam File: Provides metadata about samples in the dataset, including individual and family identifiers, parental information, sex, and phenotype. This file is crucial for understanding the demographics and traits of the sampled population.

Classical Plaintext Format: The .ped and .map files store similar information to the binary format but in a human-readable form that allows manual checking and editing.

- .ped File: Combines sample metadata with genotype data, listing individuals' familial relationships, phenotypes, and genotypes for each SNP. This format allows for a comprehensive view of the dataset but can be large and slow to process for extensive datasets.
- .map File: Corresponds to the .bim file in the binary format, detailing the genomic positions of SNPs. It provides the foundation for locating each SNP within the genome, crucial for linkage and association studies.

2.2. SNP data quality control and filtering

When dealing with genetics data, Quality control (QC) of SNP data is a critical preliminary step in genomic analysis to ensure the reliability and accuracy of your results. This process involves several key steps, each designed to identify and eliminate problematic data.

- **Missingness Per SNP and Per Individual:** This is done to remove SNPs and individuals with a high proportion of missing genotype data. High missingness can indicate poor DNA quality, issues during genotyping, or errors in data handling. In order to do this, we calculate the proportion of missing genotypes for each SNP and each individual. SNPs or individuals exceeding a predefined threshold (often 5-10%) are excluded from further analysis. This step helps maintain the integrity of statistical analyses by ensuring that conclusions are drawn from robust data.
- **Minor Allele Frequency (MAF):** This is done to exclude SNPs with very low MAFs, as they are less informative for association studies and more susceptible to genotyping errors. Rare alleles may also not be accurately represented in the sample, leading to biased estimates. In order to do this, we calculate the frequency of the minor allele for each SNP across all individuals. SNPs with MAF below a certain threshold (commonly 1-5%) are removed. Retaining only SNPs with higher MAF ensures that the remaining genetic variants are more likely to be of biological and statistical significance.
- **Hardy-Weinberg Equilibrium (HWE):** This is done to check for deviations from HWE, which can indicate inbreeding, population stratification, or genotyping errors. HWE states that

genetic variant frequencies should remain constant from one generation to the next in the absence of evolutionary influences. To do this, we perform a chi-squared test to compare observed genotype frequencies with those expected under HWE for each SNP in the control group or the entire sample for population-based studies. SNPs with significant deviations (e.g., p-value < 1e-6) are typically excluded, except for disease-associated SNPs in case-control studies, where deviations might be expected.

- **Sex Chromosome Anomalies:** This is done to identify discrepancies in sex chromosome data that may indicate sample contamination, mislabeling, or chromosomal abnormalities. In this step, we compare reported sex with sex inferred from sex chromosome genotypes. Discrepancies might lead to the exclusion of those samples or necessitate a review of the sample metadata.

- **Batch Effects:** This is done to detect and correct for variations in data that result from different sample handling, processing times, or genotyping platforms, rather than underlying genetic differences. In this step, principal component analysis or other clustering methods are used to visualize batch effects. Correction methods might include normalizing data across batches or including batch as a covariate in subsequent analyses.

- **Relatedness and Duplicates:** This is done to identify and remove closely related individuals or duplicate samples, as their inclusion can inflate the type I error rate in association studies. In this step, we estimate pairwise relatedness between individuals using identity-by-descent or identity-by-state metrics. Pairs exceeding a certain relatedness threshold may be pruned by excluding one individual from each related pair or duplicate set.

- **Population Stratification:** This is to control for spurious associations that arise when allele frequencies differ across subpopulations due to ancestry rather than a genuine association with the trait. Here, methods like principal component analysis are used to detect and adjust for population structure. Subsequent analyses can include these components as covariates to account for stratification.

2.3. Principal component analysis (PCA) based on SNP data

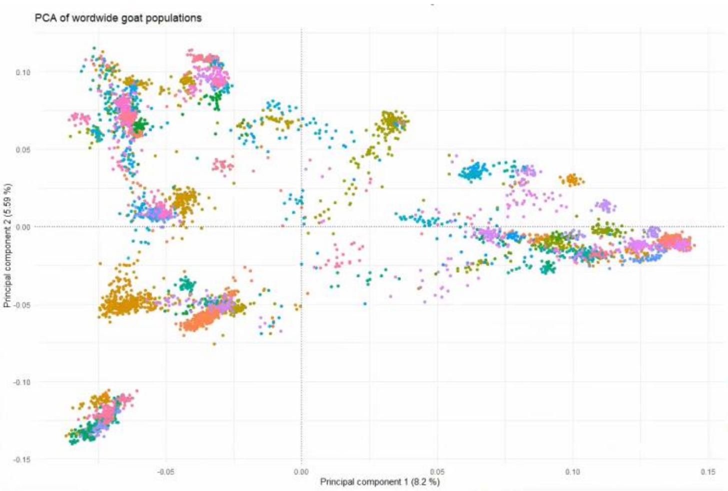
PCA is utilized in genomics to discern patterns of genetic diversity and structure within and across populations. It reduces the dimensionality of genetic data, transforming it into principal components

(PCs) that capture the majority of the variation in the data. PSA is commonly applied in genomic diversity studies and exploratory data analysis to visualize relationships and clustering among individuals or populations based on genetic similarity.

First note that as we described in the last section, before PCA, data must undergo rigorous quality control. This is to ensure accurate results. By filtering SNPs for missing data, minor allele frequency, and possibly Hardy-Weinberg equilibrium, among other metrics, we ensure that only high-quality and informative SNPs are retained. This reduces noise and potential biases in the PCA. Then, we continue with the following steps:

- **Computing Genetic Distances:** PCA requires a matrix of genetic distances or similarities between individuals, which reflects the genetic dissimilarity based on SNP genotypes. The genetic distance between two individuals can be calculated using various metrics, one common method being the pairwise identity by state (IBS) distance. IBS distance considers how often two individuals share the same alleles at a set of SNPs. For instance, if individuals share both alleles at an SNP, the IBS score might be 2 (completely identical), if they share one allele, the score might be 1 (half identical), and if they share no alleles, the score is 0. Then, we compile these pairwise distances into a distance matrix, where each cell (i, j) represents the genetic distance between individual i and individual j. Tools like PLINK calculate this matrix using SNP genotype data, considering each SNP's contribution to the overall genetic similarity or dissimilarity between every pair of individuals in the study.
- **Dimensionality Reduction:** PCA transforms the high-dimensional genetic data into a lower-dimensional space represented by the principal components. The first few PCs usually capture the most significant patterns of variation. This is done by mathematically decomposing the genotype matrix into principal components (PCs) through eigen decomposition of the covariance matrix of SNP data or singular value decomposition (SVD) of the data matrix itself. Each PC corresponds to an eigenvector of the covariance matrix, and the variance captured by each PC corresponds to its eigenvalue. In this process, the genotype data for each SNP is projected onto these PCs to transform the high-dimensional SNP data into a lower-dimensional space. The projection is a linear combination of the original SNP genotypes weighted by their loadings on the PCs, with the loadings indicating the correlation between each SNP and the PC.

- **Visualization and Interpretation:** In the PCA plot, each individual is represented as a point in the space defined by the first two or three PCs. The proximity between points reflects genetic similarity. This visualization can reveal clusters that often correspond to known population structures, genetic backgrounds, or even unexpected outliers that may warrant further investigation. In addition, the axes of a PCA plot correspond to the PCs, which are orthogonal to each other, ensuring that each PC captures unique variance in the data. The percentage of total variance explained by each PC is often annotated on the axes. This assists in analyzing the data's dimensionality and the informativeness of the PCs. There are some sample PCA plots shown below:



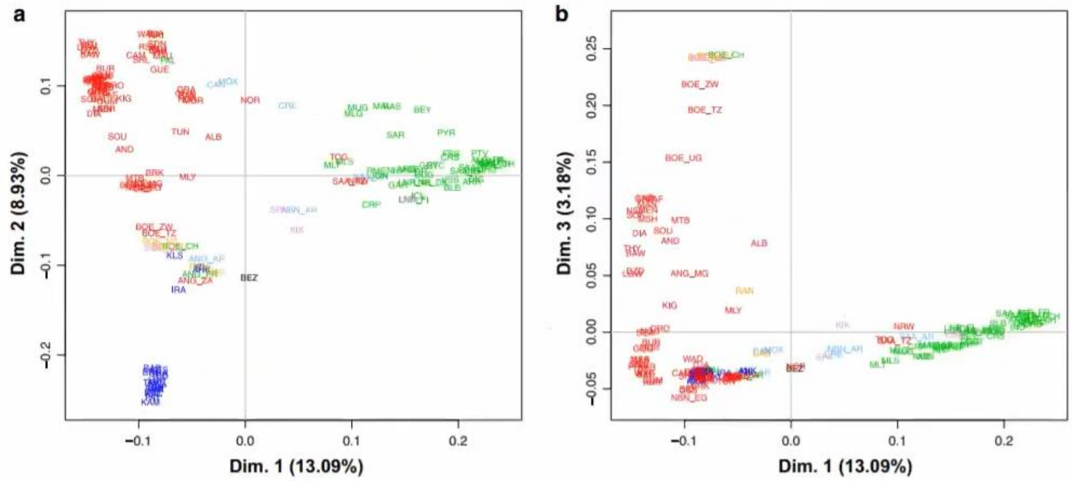


Fig. 4 Multi-dimensional scaling plot. Dimension 1 versus 2 (**a**) and dimension 1 versus 3 (**b**). The population labels are coloured according to the continent of origin as follows: red = Africa, green = Europe, blue = West Asia, pink = North America, light blue = South America, orange = Oceania, black = wild goats. To increase readability, the country codes are omitted from the population labels, with the exception of breeds sampled in multiple countries

Sources available here and here

- **Eigenvectors and Eigenvalues:** Eigenvectors determine the direction of the PCs in the multidimensional space of the original data. The loading of a SNP on a PC (its weight in the eigenvector) quantifies the SNP's contribution to that PC. SNPs with high loadings significantly influence the variance captured by that PC. In addition, eigenvalues quantify the amount of variance in the original data explained by each PC. The ratio of a PC's eigenvalue to the sum of all eigenvalues gives the proportion of total variance captured, guiding the selection of how many PCs to consider for a comprehensive yet concise representation of the data.

- **Advanced Visualization Techniques:** Also note that by incorporating metadata (e.g., population labels, geographical origins) as point annotations or colors in PCA plots, we can visually check and look for possible relations between genetic variation and external variables, potentially finding genetic patterns related to geography, phenotypes, or other factors.

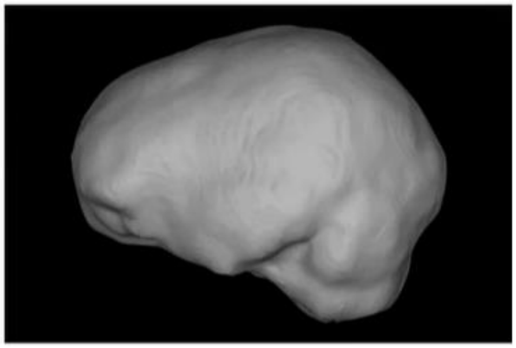
A very good series on data wrangling with PLINK could be found here.

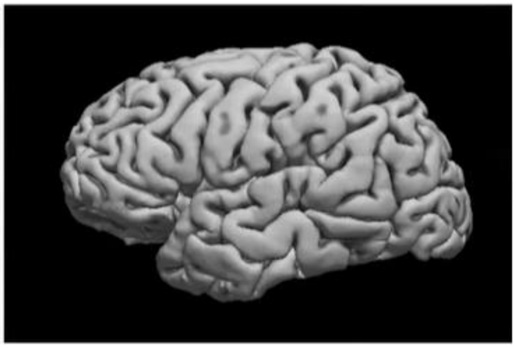
Skull Stripping and Brain Extraction: Brain/non-Brain Segmentation

In preparation for registration and segmentation, we need brain extraction. This process, often facilitated by tools like BET (Brain Extraction Tool), separates brain material from non-brain material. The goal is not absolute accuracy, but rather the removal of the bulk of non-brain material, ensuring subsequent algorithms are not influenced or biased by irrelevant structures.

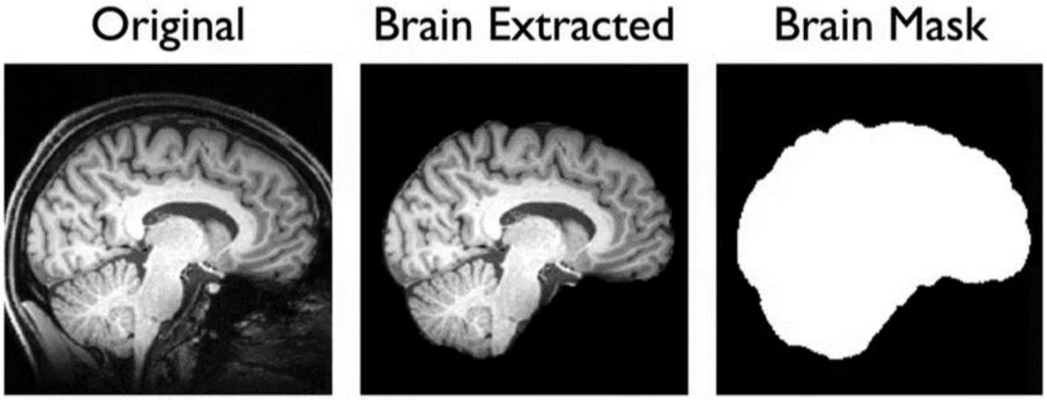
However, brain extraction is not without challenges. It is easy to encounter difficulties near tissues like marrow or membranes and we may have some blood vessels.

There are different ways to do this. For example, BET (used on MRI sequences) is a relatively popular algorithm, available in FSL, that focuses on achieving a simple separation that removes the bulk of non-brain materials outside the brain surface (e.g. scalp, marrow, etc.) and it is robust to bias fields (this is done by using local intensity changes). But BET does not delve into the intricate folds of the brain. For such purposes, we will need to leverage other tools. In the illustration, the image at the top belongs to the input image, and the image at the bottom shows the extracted output brain.





The underlying mechanism of BET involves starting with a sphere and expanding it, particularly when the sphere extends beyond the brain's boundaries. This expansion aims to find the brain's edge and achieve a clean separation. The tool provides flexibility in its output, offering either a brain-extracted image or a brain mask.

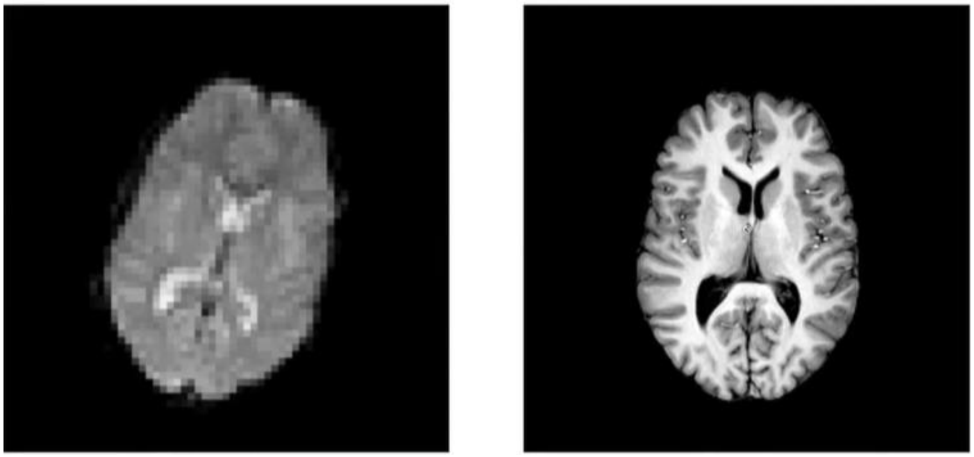


Despite its utility, brain extraction with BET does encounter challenges. The proximity of tissues like marrow, membranes, and blood vessels, often shifted during imaging, poses difficulties. However, for registration, the focus is on eliminating the majority of non-brain material rather than achieving absolute accuracy. However, in cases of quantifying specific brain components, a higher level of accuracy may be necessary, as in the context of segmentation.

*More useful tools on the Brain-extraction step will be covered later on in the “Suggested Available Automatic Tools” section.

Registration:

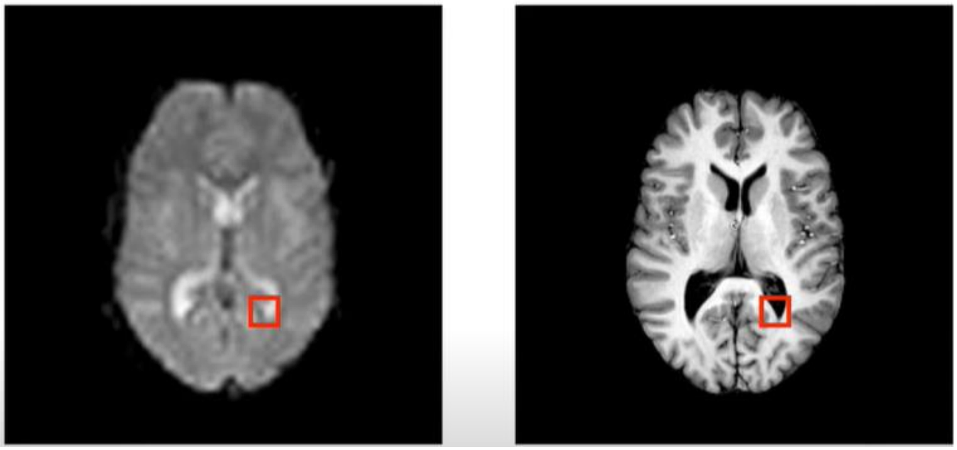
So, let's begin by understanding what registration entails. When we observe the following two images, it's evident that they are not aligned – one appears tilted concerning the other.



Ensuring alignment is crucial for picking a specific location within the image, represented by a particular row and column in 2D panels, signifying the same anatomy. Currently, these images don't share the same anatomy; this means that we cannot use the same row and column values to show a specific region.

It is helpful to make sure all our images have similar anatomy before we input them into a model. (However, we also need to use augmentation methods later on which changes our alignment again.)

Registration is the process through which images are brought into alignment. Post-registration, these images would be well-aligned, and a specific location within them represents the same anatomy.



Registration is especially helpful in cases where we are dealing with quantifying structural change, correction for motion changes in time), combining across individuals in group studies including fMRI and diffusion (changes between individuals), and functional analysis.

Now, I begin the details about different things that need to be done in registration. I'll be discussing four main concepts you need to grasp: image spaces, spatial transformations, cost functions, and interpolation. Understanding these concepts is crucial for selecting the best options and achieving optimal registration for our specific problem. At the end, I will give an introduction to the two registration methods: Pathological Image Registration (Cost Function Weighting), and Small Field of View registration (Small FOV).

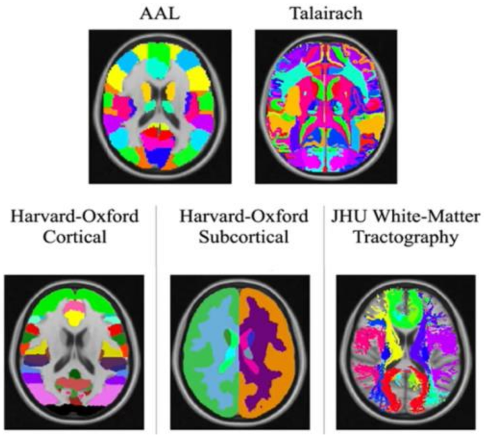
1. Image Spaces

We say two images are inside the same space if we can use the same process to map them to the anatomy image space. Note that a difference in image resolution isn’t the same as a difference in space.

1.1. standard space

Image spaces have multiple categories. The most familiar image space is the standard space, with examples like Talairach space and MNI space: Talairach space was initially introduced by Tara Contorno, who conducted detailed dissections on a postmortem brain, recording coordinates to create a standardized representation. However, the more common choice nowadays is the MNI 152 (or ICBM 152), an average of MRI images from 152 young healthy individuals. This template allows us to establish a common way of describing anatomical locations within the brain using coordinates. (More details about other MNI types can be found here.)

Standard space atlases are also mostly available in MNI 152. Because, even if they were not generated for the same 152 individuals used to create the MNI 152 template, they have been registered and aligned with the standard space templates. (Atlases are templates or reference maps that serve as standardized representations of brain anatomy.) Some atlases can be seen below:



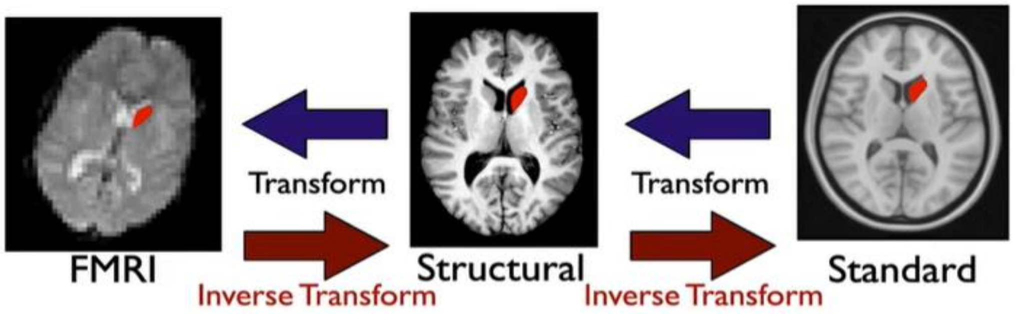
1.2. Other Spaces

Other than the standard Space, we also have other spaces such as functional space (from fMRI), structural space (like in MRI), etc. We take each structural and functional image to have a different space. Because there might be a change in the orientation of the head between the acquisitions (the person might have moved). Or it might be that the structural images are taken in a different session than functional images from the patients.

Because the mapping of the coordinates between these images to the anatomy image space is different, we say they have different spaces.

Note that our images might have different resolutions. So, even if we map them all into the same space, the resolutions would be different, Thus, we usually define different resolutions for each space that we map our images into (2mm, 1mm, and 0.5 mm resolutions are common). So, a different resolution doesn’t mean a different space.

That’s why we need registration. To make sure we can map all our images to any space we need in the same manner (registration links different spaces together and aligns all our images). Once registration is done, we can transform or inverse-transform the information:



Note that the outputs of the tools/software online, that preprocess the data might give you the output of different images in different spaces and you need to make sure all the outputs are in a suitable space before using them in your model.

Be careful when dealing with MRI/fMRI image spaces. Unlike what you might expect, the tools designed for MRI/fMRI preprocessing usually use multiple types of coordinate systems. For example, when dealing with NIFTI data, they might use a voxel coordinate system, standard space coordinate systems, or some other coordinate system. An explanation from FSLeyes documentation can be found here. Another example is FreeSurfer. It maps the 3D brain data into 2D rectangles (for segmentation).

*We already mentioned that there are many types of standard spaces. standard space coordinate systems also come from those. Such as MNI152 coordinates which could be used.

To move our images to image spaces (/image space atlases) of our choice, we need to select and use the suitable registration type (by selecting the necessary Spatial Transformations allowed). This ensures that, later on, we can move our images from one space to another space easily.

2. Spatial Transformations

Whenever we align our images, we need to decide what transformations we want to allow the registration to do. Basically, what things are possible within the alignment? Is it allowed for us to change the brain's shape or size? Spatial Transformations specify these things. They are specified by their name (a partial description of the transformation), and the degrees of freedom (DOF). Here, we will discuss 3 of such transformations:

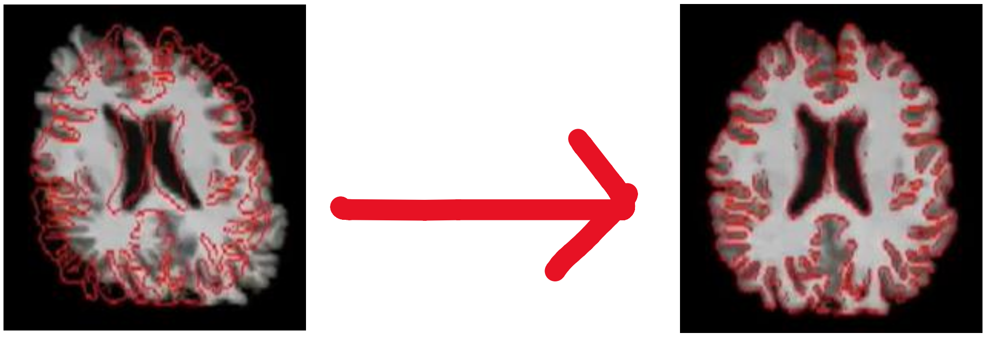
2.1. Rigid body transformations (rigid registration): 6 DOF in 3D

Rigid body transformation gives us the ability to use the types of transformations that are usually usable on a normal physical object. That is, it gives us the ability to use 3 rotations (rotation with respect to 3 axes) and 3 translations (translation with respect to 3 axes) (this is why we have 6 DOF. Because 3+3=6). But we cannot change the shape or size. The rigid body transformation is usually used in the case of within-subject registrations.

This is done because we take the movement of a patient's head into account. Because, when patients move their heads, they are only changing 'where' their brain is inside the space (the orientation, the rotations, and the position on the translations). They are not changing the size or the shape of their brains.

2.2. Affine transformation (affine registration): 12 DOF

In affine transformations, we still have 3 rotations and 3 translations which we had in rigid body transformation. But we add 6 extra DOF: 3 scaling (scaling with respect to 3 axes) and 3 skews/shears (skew with respect to 3 axes). Skew is similar to a combination of both rotating and scaling an image. Like below:

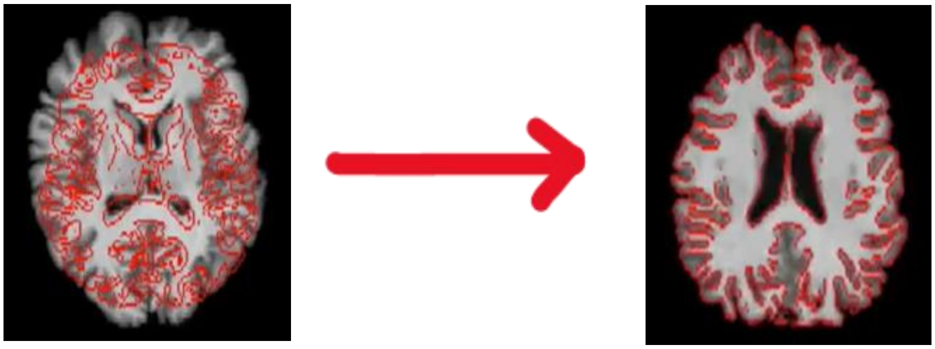


This transformation is either done to correct the distortions, which are linear (also called eddy current correction), or it is used to initialize non-linear registration.

Now, we cover the non-linear registration:

2.3. Non-linear transformation (non-linear registration): millions of DOF. Usually, 12 M

Here, our initial image could have millions of different weird shapes, like the following image. And we allow the registration algorithm to transform all those images using a non-linear transformation.

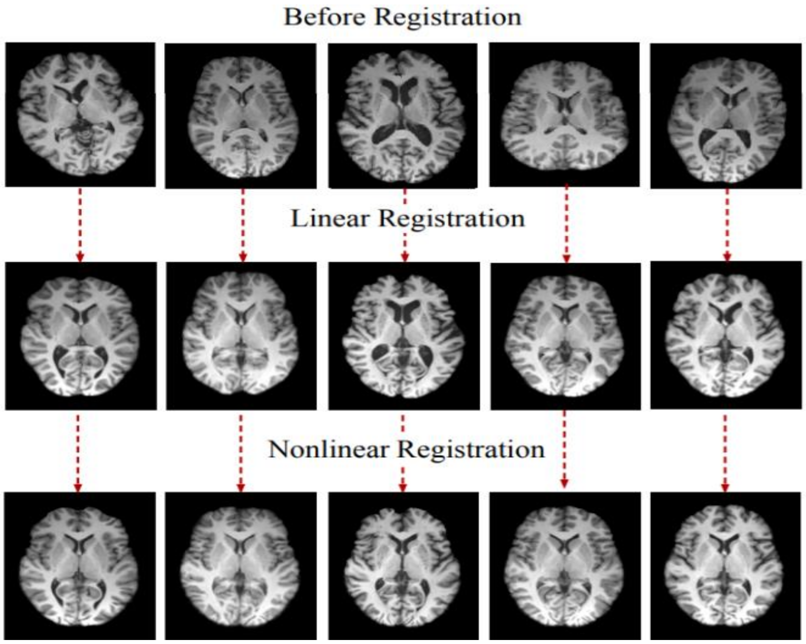


Of course, we still need to put some constraints on the transformation that happens here. We don't want to break up the pieces of the brain's anatomy during the transformation. Neither do we want to fragment the brain and put different things in different places. To summarize, we want to make sure everything sticks together in the correct order during the registration. While not also removing any region by accident.

This is why we need some basis functions (such as B-splines) over some regularization, topology-preservation, diffeomorphisms, etc. (these are mathematical methods)

This transformation is used for good-quality between-subjects registrations. Because for a good alignment between individuals, we need such a flexible algorithm. Note that the brain of a human being is similar to the fingerprint: completely different and unique for each human.

The following image provides a comparison between linear and non-linear registrations:



*To ensure non-linear registration doesn't have too much freedom in its transformations, it is usually recommended to use a warp resolution of 10mm for MNI152 as the regularization.

2.4. Global Scaling transformation: 7 DOF

A transformation used for good within-subject registrations (like rigid body). But we also have an extra global scaling (where the scaling is equal in all three axes). This transformation corrects for scanner scaling drift in longitudinal studies of Cost Functions. But this is not common to use.

*Note that, unlike what it may seem like, a higher DOF does **NOT** mean you will get a better model later on.

*In general, Spacing of points = warp resolution = regularization = DOF

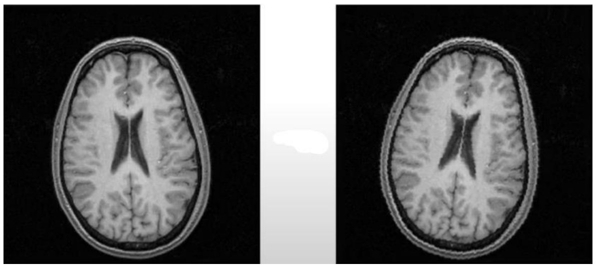
3. Cost Functions

Cost functions are the functions we use to measure how good the alignment between our images is before we run the registration algorithm.

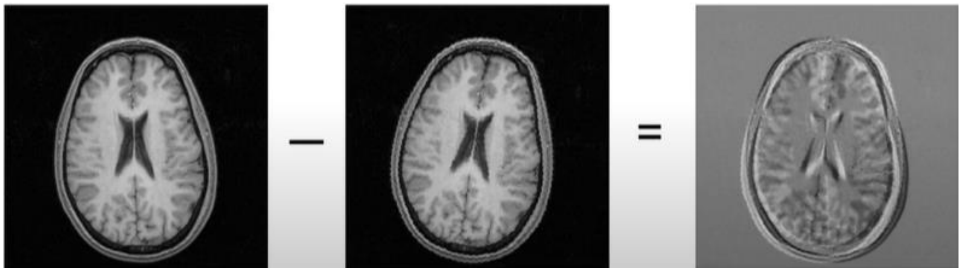
To make it short, these functions are necessary because the registration algorithm needs to have a way to identify whether or not our images are aligned. Then, if they are not aligned, the algorithm needs to activate and align them.

After all, unlike humans, the registration algorithm has no intelligence to understand this just by looking at the images. So, we need to find a way (cost functions) to make it understand when to activate the alignment algorithm.

For example, we can understand that the following 2 images are not aligned by looking at them:



But our computer couldn't do that. To make the registration understand that these 2 are not aligned, one basic method could be to subtract these images and ask it to look at the output. If the images are aligned, we expect it to be almost zero everywhere (an almost 'flat' image). However, if they are not aligned, the algorithm should be able to identify a pattern in the output.



The output has a clear pattern (it’s not ‘flat’ at all). So, our algorithm understands: these 2 images are not aligned and it needs to align them.

This way, when we have a group of images and we want to check how aligned are these, we can take their pairwise subtraction, take the square root of the subtraction outputs, and sum them up. This is called SSE (sum squared error/ sum squared differences) in statistics. And here, this value shows us how aligned are our images: the smaller the SSE, the more aligned our images are. The registration algorithm should then align the images, such that we can minimize SSE as much as possible without changing the structure of our images too much. After all, we don’t want to make all our images identical to each other.

*Some algorithms use the exact opposite of this: Similarity Functions have a similar process to the above method, but instead, they activate when the alignment is judged to be good. And try to find the maximum SSE value.

Now, I will discuss some famous cost functions and how they work. I will explain some mathematical background for these functions to help your understanding. But you could skip the details if you want. Similar to the other steps of MRI/fMRI preprocessing, these cost functions are predefined in most of the preprocessing tools you might be using. Also, note that these are just some examples. Depending on your problem, and the preprocessing software you are using, you might have better options available. *Note that not all below cost functions are suitable for non-linear registration.

3.1. Least Squares

This is the process explained above, where we minimize the sum squared differences between corresponding intensities in the reference and target images. It assumes that the relationship between the images is linear and seeks to find the transformation parameters that minimize the sum of squared intensity differences. This cost function could be used when you **only** have images of the same type and modality (e.g. when you only have T1 MRI images).

3.2. Normalized Correlation

Normalized correlation measures the similarity between two images by computing the Pearson correlation coefficient between corresponding intensity values after normalization. It assesses the linear relationship and similarity of intensity patterns between images.

Similar to the least square cost function, normalized correlation could also be used **only** if you have images of the same modality (e.g. when you only have T1 images).

This cost function allows for variations in overall brightness and contrast between the images.

3.3. Correlation Ratio

The correlation ratio evaluates the similarity between images based on the ratio of variances in intensities before and after registration. It assesses how well the intensity patterns match between images while accounting for variations in global intensity distributions.

This cost function is suitable for aligning images of **any MR modalities (i.e. fMRI, MRI, rs-fMRI, diffusion MRI, etc.)**, providing robustness to variations in overall intensity levels.

3.4. Mutual Information

Mutual information quantifies the amount of information shared between two images by measuring the statistical dependence between their intensity distributions. It does not assume linearity and is capable of capturing non-linear relationships between images.

This cost function is versatile for aligning images of **any modalities (i.e. MR, CT, PRT, etc.)** and is the most flexible in terms of modality. It works by assessing the shared information content regardless of intensity variations.

3.5. Normalized Mutual Information

Normalized mutual information is a variant of mutual information that normalizes the mutual information measured by the entropy of the images. It quantifies the amount of shared information between images while accounting for differences in their individual information content.

This cost function, similar to mutual information, is suitable for aligning images of **any modalities (i.e. MR, CT, PRT, etc.)** by assessing the shared information content while accommodating variations in individual image entropy.

3.6. Boundary-Based Registration (BBR)

BBR focuses on aligning image boundaries, particularly useful for EPI distortion correction. It utilizes intensity gradients along tissue boundaries to drive registration, emphasizing local alignment in regions with distinct tissue boundaries.

This cost function is specialized for aligning functional and diffusion MRI images with EPI distortion correction, optimizing registration in areas with high-contrast edges.

3.7. Non-linear registration

As mentioned before, many MRI/fMRI tools don't usually accept any of the above cost functions (there might be some specialized tools that do that, though. We need to carefully check all available tools and make sure).

For example, FNIRT which is one of the most used algorithms for non-linear registration, is primarily constrained by its cost function, which is exclusively the least squares cost function. This constraint limits FNIRT to registering images of the same modality, typically two T1-weighted images (although, this is usually not that big of a problem. Because most researchers are interested only in T1 images).

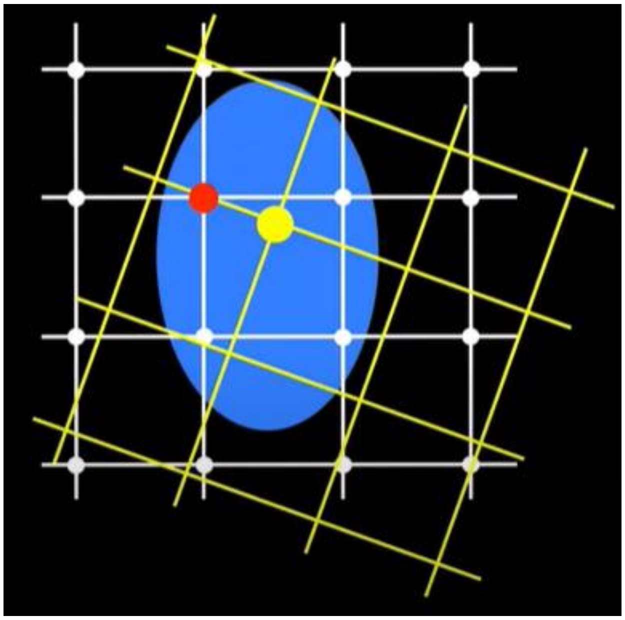
A feature that many algorithms (such as FNIRT) have is the usage of a bias-field model when doing the registration. This addresses the variations that exist in the image intensity of different regions Bias fields often manifest as darker or lighter areas within the image due to magnetic field inhomogeneity (this will be explained in detail later)

Such non-linear registration methods can deal with the bias-field problem by squeezing/expanding the white matter (and expanding/squeezing the dark matter at the same time) in the necessary regions (this increases/decreases the average density of the problematic regions in a suitable manner.

4. Interpolation and Masks (an old paper regarding this) (also check: 1, 2, 3)

MR data usually has a big limitation: the resolution of images. In general, all our images have poor resolution. This is why a post-processing technique that allows the best possible resolution to be obtained is needed. This is why we need interpolation. Assume we put a grid over our images. Now, interpolation is a method that allows us to find the intensity values of unknown points inside each grid block. This way, by adding new information about each point to each grid block (we 'fill in' each grid block), we slowly update the intensity of the image and try to reach the best possible resolution.

For example, in the below image assume that we initially put the white-colored grid on our image. Where, the grid lines intersect only at the image pixels (for example, here our image is very simple and has only 4 pixels. This grid would be much larger for higher resolution images). Now, we rotate the grid and get the yellow grid. We are interested in finding the value of these newly generated points that don't align with the white group. For example, assume we want to find the intensity of the yellow point (the rotated result of our image's red pixel).



Unfortunately, our image doesn't have any information about this yellow point. Because this point is not within our original image pixels. We need interpolation functions to figure such problems out. There are different approaches for interpolation and we'll have a short review on some of them here.

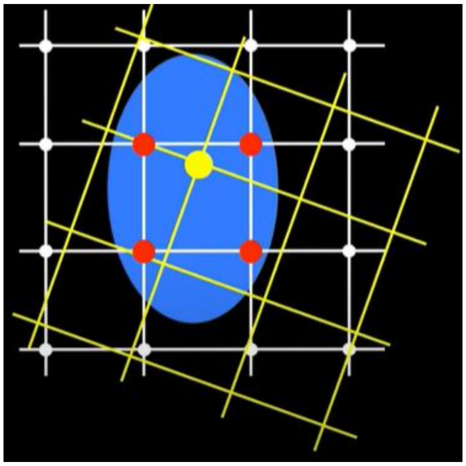
4.1. Nearest Neighbor

Nearest Neighbor is the most straightforward: it checks the distance of the yellow point from each of the 4 grid corners that surround it and finds the point that has the shortest distance (the red point in this example). Then, it copies the value of the nearest point to the new point. The same process would then continue for all other points on the yellow grid.

The advantage of NN is that it is fast. But it usually results in blocky edges and fails to preserve the details well.

4.2. Trilinear

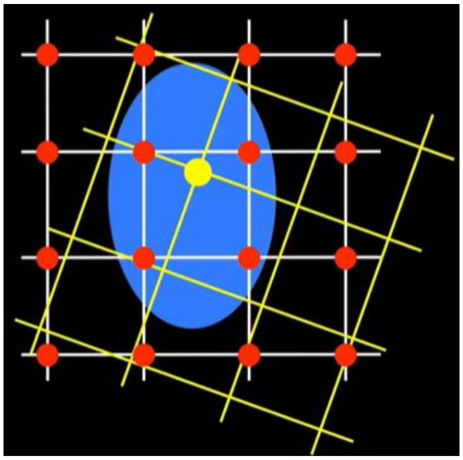
Trilinear (or bilinear in 2D) interpolation considers the immediate neighbors that surround the target point (in this case, the 4 red points) and computes a weighted average based on their distances. Such that, the shorter the distance from the yellow point, the more effect on the yellow point's intensity. The weighting is done using a linear function here.



This method is fast, similar to NN. And it leads to smoother results. But it usually blurs internal details.

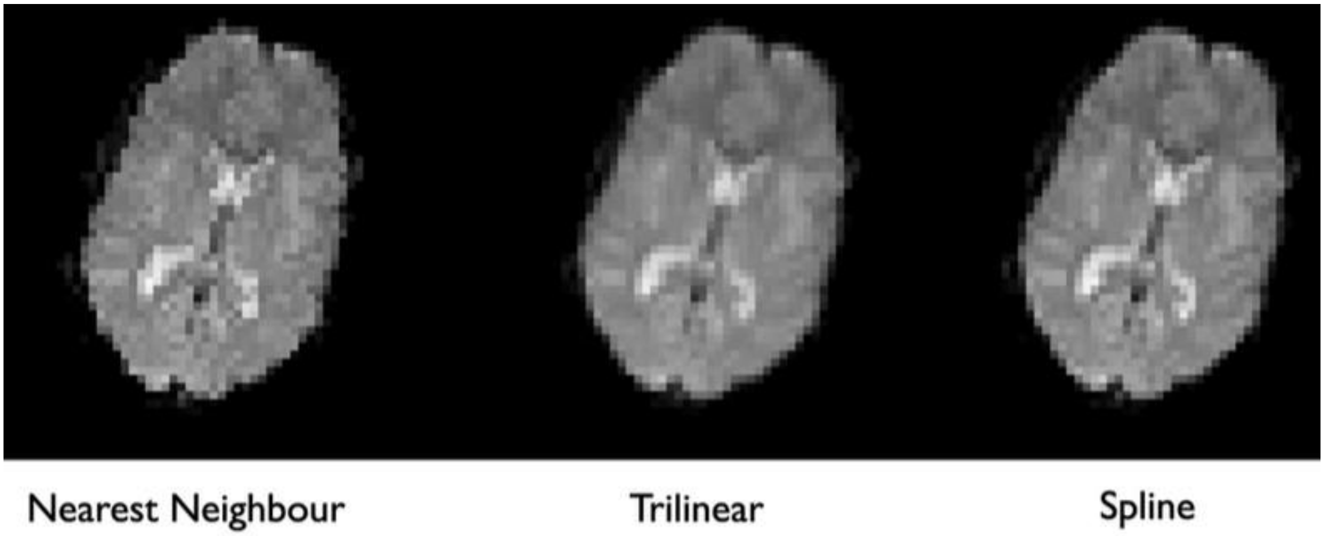
4.3. Spline

This method is very similar to the previous approach. We again use a weighted average to calculate the intensity of the yellow point. But here, instead of just taking the 4 surrounding points, we take the average between all intersection lines on the white grid. Basically, we take a weighted average over all the red points in the following image (weighted based on their distance to the yellow point).



This method provides sharper boundaries and better preserves the details. However, it is slower than NN (albeit still fairly fast) and it can lead to problems like ringing. Because we don’t constrain the output range. This could result in output values exceeding the original data range. (e.g. if you want your intensity to be always smaller than 2, taking the average over a large number of points might result in a value larger than 2).

Below, you can compare the outputs resulting from the previous 3 methods:



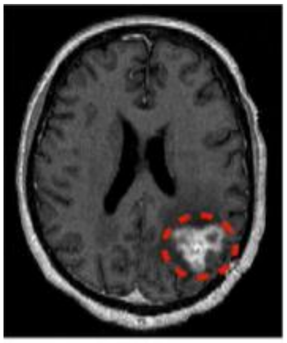
Two other famous approaches are Sinc interpolation (it's very mathematical and uses the Sinc function. It's a bit time-consuming as I understood. But the results are better compared to the previous 3 methods. Some details here) and k-space interpolation methods (such as this) that might be used. I will not add their details now (I might do so later).

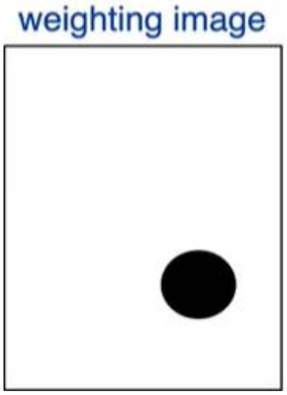
5. Pathological Image Registration (Cost Function Weighting)

This type of registration is a method we use as the solution to a special problem: when we have pathology or artefacts inside our image. To summarize, we need this method as a way of telling the registration that there is an area of the image that is not normal (it is not consistent with what we see in the other images that we're trying to register). Because we need to make sure that there's a good match between the images after they've been transformed (but the existence of pathologies causes a problem here).

So, we need to tell the registration to ignore the pathology (or artefact) areas where we know that there's just never going to be a good match. So that those areas don't bias the rest of the registration (if the registration algorithm doesn't ignore those areas, it would instead try to change the structure of other non-problematic images, and transform them into the structure of the problematic image).

Cost function weighting itself is quite straightforward: all We need to do is to set up another image which is a weighting image, and that image needs to be zero in the area that we want the registration to ignore and one everywhere else. One such example can be seen below:





*Cost function weighting can be done whether we're doing linear registration or nonlinear registration.

*Be careful not to put zeros in the background because it's essential for the registration to understand where the brain stops. If you have told it to ignore everything outside of the brain, it never knows that there is non-brain there, and that's a problem. So, we put one everywhere, including the background, except the areas around the pathology or the artifact that we want to ignore.

6. The Overall Overview of Registration

Now that we have a broader understanding of the registration process, let's go through a summary of the overall registration process:

- Step 1: First, we need to find a suitable spatial transformation method. Then, the registration algorithm knows an estimate of what transformation needs to be done. (In this step, we’re still not generating any new images. We’re just estimating what transformation ‘needs to be done’. So, no resampling is done yet). In this step, we select the previous things we talked about like the type of registration, cost function, interpolation, etc.

- Step 2: Now, we apply the transformation we have decided on. Meaning that, by resampling, we create a new, modified image.
- Note that in each transformation step, there is usually a **degradation in the image quality**. This is why we need to delay resampling as much as possible to avoid reducing the image quality too much. **(We don't want to apply resampling multiple times,** especially not the interpolation step**)**
- Other terms you might hear:
    - Co-Registration is doing registration between 2 images
    - Spatial Normalization is doing registration. When we want to map our images into a standard space such as MNI 152.

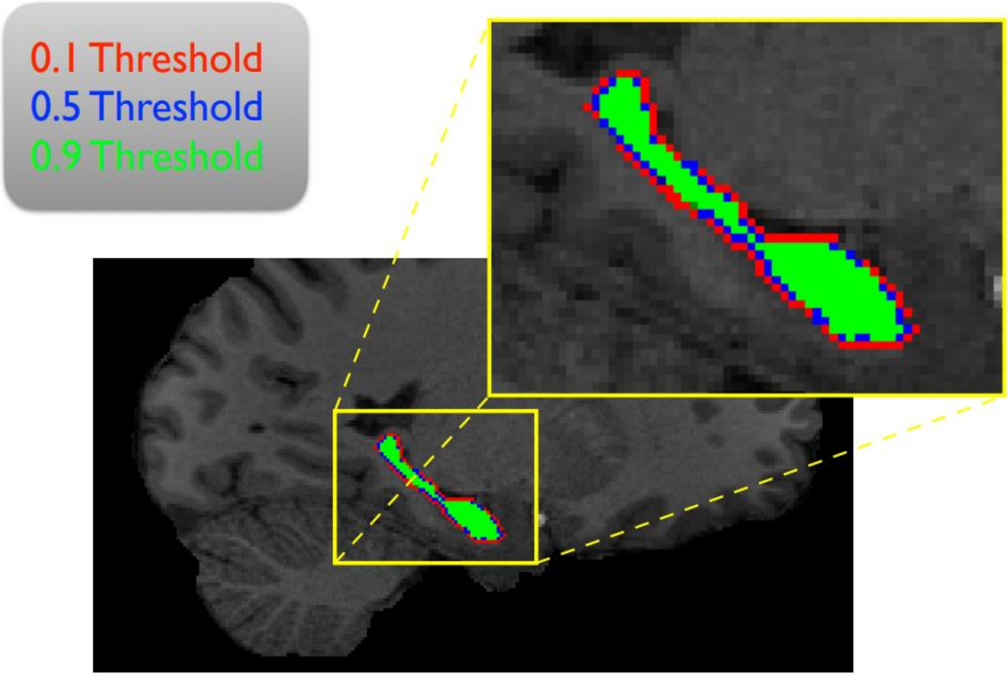
7. Transforming and Thresholding Masks

Binary masks are essential tools in image processing, often used to represent regions of interest or specific structures. When transforming these masks from one space to another, interpolation becomes necessary to ensure accurate representation.

A binary mask consists of black and white regions, where the white areas denote the region of interest, while the black areas represent everything else (we've already seen multiple examples in previous sections). Binary masks are commonly utilized to isolate and analyze specific structures within images, aiding in various analytical tasks.

Usually, Trilinear interpolation is a widely used method for transforming binary masks between different spaces. This is because, unlike simple binary masks, trilinear interpolation assigns values between zero and one to voxels along the edges of the mask. This interpolation takes the degree of overlap between the new voxel and the existing ones into account, offering a more robust representation. After this, thresholding is used to convert interpolated values back into binary form. By adjusting the threshold value, we can control the inclusiveness of the mask.

Different threshold values result in varying degrees of inclusiveness or conservatism. High Threshold (e.g., 0.9) leads to a smaller area of inclusion, preserving only voxels with high overlap with the original mask. Medium Threshold (e.g., 0.5) includes voxels with moderate overlap, expanding the area of inclusion beyond the high threshold. Low Threshold (e.g., 0.1) results in a high degree of inclusion, covering voxels with minimal overlap, including the core as well as surrounding areas.

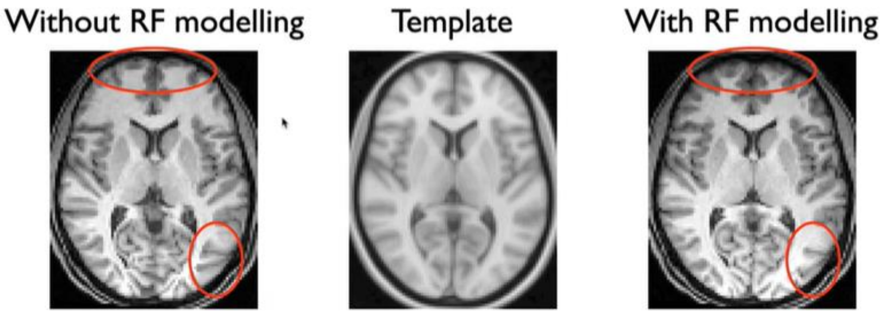

0.1 Threshold
0.5 Threshold
0.9 Threshold

Bias-Field Correction

A feature that many algorithms (such as FNIRT) have is the usage of a bias-field model when doing the registration. This addresses the variations that exist in the image intensity of different regions Bias fields often manifest as darker or lighter areas within the image due to magnetic field inhomogeneity (basically, some parts of the brain might be lighter than they should be, while some parts might be darker). So, it is helpful to understand and compensate for these variations during registration. Although, this feature might not always be beneficial and in some cases, you may want to disable such options from your registration. For example, when dealing with quantitative images, such as T1 relaxometry scans, you may want to disable bias field modeling to preserve quantitative values. However, for standard neuroimaging applications, having bias field modeling enabled improves the quality of registration.

To be more technical, bias-field (or illumination artifact) happens because of a lack of radio-frequency (RF) homogeneity. Although this is not significantly noticeable on visual examination, it can seriously degrade the volumetric quantification of MR volume upon applying the automatic segmentation algorithms that use intensity levels.

Various approaches exist to deal with bias-field effects, both during and after image acquisition. Some of These methods that aim to enhance the acquisition process itself are: phantom-based calibration, multi-coil imaging, and specialized sequences. On the other hand, retrospective techniques and algorithms such as filtering, masking, intensity adjustments, and gradient-based corrections could be applied post-acquisition to address bias-field artifacts.

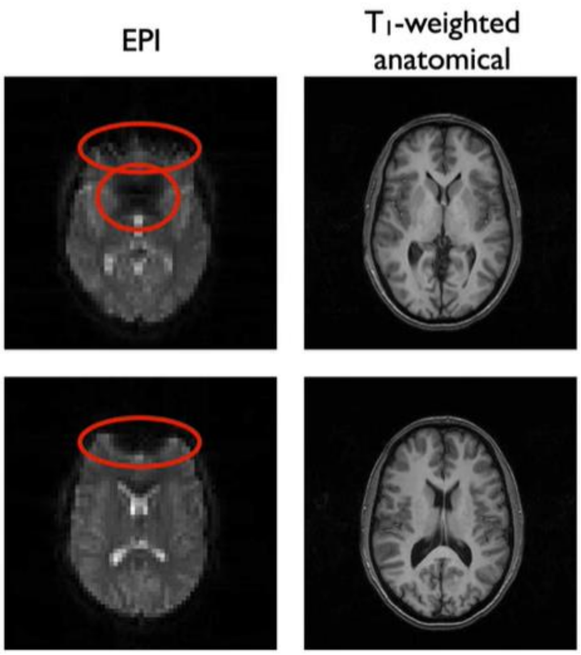


***If necessary, I will add the details of the mentioned methods to this file later

fMRI: Some relevant registration

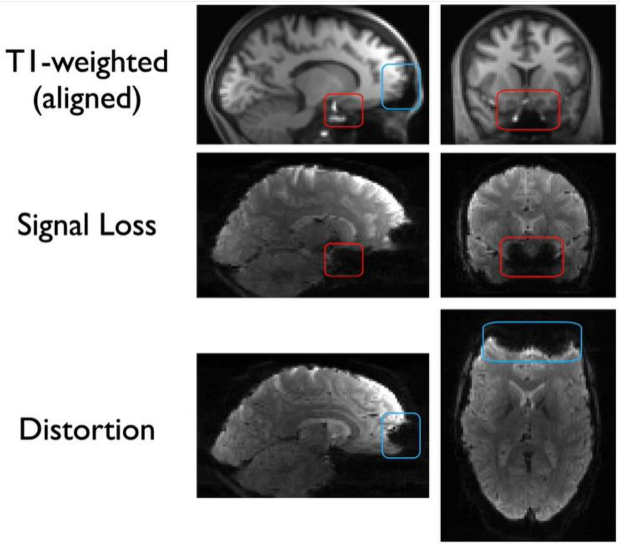
1. EPI Distortion Correction

When dealing with fMRI and diffusion MRI, a rather important type of preprocessing is the correction of Echo Planar Imaging (EPI) distortions. These distortions happen because the magnetic fields used in MRI aren't uniform everywhere. This is particularly troublesome near the front and bottom parts of the brain, where the signal can get displaced or lost entirely. These distortions show as geometric aberrations, where the signal is not located where it should be, and as signal loss, characterized by areas of the image lacking signal altogether. To deal with these issues, the concept of field maps is introduced. Field maps are special MRI acquisitions that measure the magnetic field's variations across the brain, providing invaluable data on field deviations. This information is particularly crucial in areas prone to signal loss, such as the inferior frontal and temporal lobes.



The correction process involves using these field maps to predict and correct spatial distortions, thereby enhancing the registration quality. It's important to note, however, while geometric distortions can be corrected, signal restoration is not possible once lost. Understanding the patterns of signal loss, though, can aid in making more informed corrections and improving the overall quality of the registration.

One of the primary challenges in EPI image registration to structural scans is the complex nature of the distortions present. Effective correction strategies, notably the utilization of field maps, can significantly improve the alignment accuracy, making the registration process more reliable. Among the techniques employed for registration is Boundary-Based Registration (BBR), which focuses on the more reliable boundary between white matter and gray matter. This approach provides robustness even in the presence of EPI distortions, making it particularly effective for structural to EPI registrations.



Evaluating the efficacy of distortion correction involves comparing corrected and uncorrected images, with properly corrected EPI images showing better alignment with structural scans, especially in regions prone to distortions. The success of the correction process is based on the quality of the field maps and setting accurate parameters like effective echo spacing and the phase encoding direction.

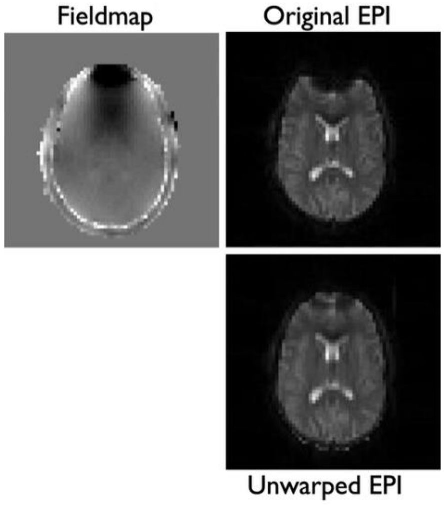
The use of EPI distortion correction is critical in fMRI imaging. This method is great for capturing images quickly. However, EPI has a problem: it can produce images with distortions and areas where the signal is lost, making it tricky to match these images accurately with more detailed structural MRI scans.

2. Unwarping

After correcting for distortions in EPI images, the next step is to align these corrected images with high-resolution structural scans. This process is a type of registration, and it's essential for combining fMRI or diffusion MRI data from EPI with detailed anatomical information.

The registration process involves matching the corrected EPI images to a structural scan, which serves as a reference. This matching is done by adjusting the position, orientation, and sometimes the scale of the EPI images so that significant features in both sets of images line up correctly. This alignment allows researchers or doctors to see where functional or diffusion activities are happening in relation to the detailed anatomy of the brain.

Field maps play a key role in this process. Before registration, the field maps are used to unwarp EPI images. Unwarping is the process of correcting the EPI images based on the distortions identified by the field maps. These maps show us where the magnetic field was not uniform and by how much, allowing us to reverse the distortions in the EPI images. This step is crucial because it ensures that the geometric distortions and signal displacements in the EPI images are corrected as much as possible before attempting to align them with the structural scans.



The unwarping process uses the information from the field maps to calculate how much each part of the EPI image needs to be adjusted. This involves shifting the positions of pixels in the EPI images so that they more accurately represent where they should be in reality. The amount and direction of these shifts vary across the image, depending on the variations in the magnetic field shown by the field map.

Once the EPI images are unwarped, the registration process can begin. During registration, various algorithms and techniques are used to find the best fit between the EPI and structural images. One common technique used in this context is Boundary-Based Registration (BBR), which focuses on the clear boundary between gray and white matter as a reliable feature for alignment. BBR is particularly effective because it relies on features that are less likely to be distorted in EPI images, making it a robust choice for aligning these images with structural scans. (Some details)

In practice, ensuring the accuracy of registration involves carefully checking the alignment in areas known to be prone to distortions, such as the frontal and temporal regions of the brain. It also means being mindful of areas with signal loss, as these won't align well due to the lack of data. Researchers and clinicians must critically assess the registration quality, especially in these challenging areas, to ensure the functional or diffusion data is accurately mapped onto the brain's anatomy.

This step in the imaging process is vital for a wide range of applications, from research studies that explore brain function to clinical diagnoses that depend on accurately locating areas of interest within the brain's complex anatomy.

3. Co-Registration

A combination of these registration steps in fMRI is called co-registration (the precise alignment of functional and anatomical images within the same individual.) In This section, I will give a more general summary of some more reasoning and some other things that could be used in this step.

Co-Registration ensures that the areas of brain activity detected through fMRI can be accurately localized within the structural context provided by anatomical MRI or other imaging modalities like PET or CT. The anatomical reference is often derived from a high-resolution 3D MR sequence such as MP-RAGE, which provides detailed structural information with isotropic voxels, allowing for versatile manipulation of the data including rotation and re-slicing.

The co-registration process employs various image correction and alignment techniques, which often start with image resampling through interpolation to adjust for any differences in voxel size or resolution between the functional and anatomical images. Following this, a series of rigid body transformations are applied, which include translations (shifts in the image position), rotations (turning the image around the x, y, and z axes), zooms (scaling the image size), and shears (distorting the image shape). These transformations are iteratively refined using an optimization protocol that aims to minimize a cost function, which quantifies the disparity between the functional and anatomical datasets. The optimization ensures that the functional data are accurately overlaid on the anatomical images, providing a coherent and precise representation of where brain activity occurs relative to the brain's anatomy.

Normalization, while related to co-registration, extends the process by aligning individual data to a common anatomical template, allowing group analyses. This involves more complex transformations to account for the anatomical variability across subjects, including affine transforms and nonlinear methods. Regularization techniques may be employed to constrain these transformations, preventing unrealistic stretching or shrinking of brain structures. Tissue segmentation can further refine the process by differentiating between gray and white matter, improving the alignment accuracy.

The choice of the anatomical template is significant in normalization. Historically, the Talairach Atlas, based on a single post-mortem brain, was widely used. However, templates from the Montreal Neurological Institute (MNI), derived from MRI scans of several hundred healthy adults, are now more common. These templates, like the widely used ICBM152, offer a more representative anatomical framework for group analyses in fMRI studies.

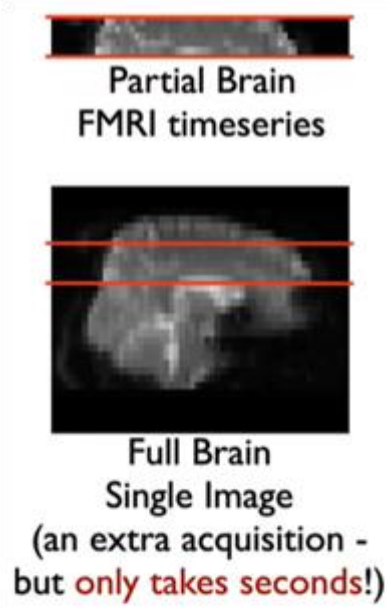
Co-registration and normalization are foundational elements in the preprocessing of neuroimaging data, ensuring that functional imaging findings can be accurately interpreted within an anatomical context. The precision of co-registration directly impacts the reliability of localizing functional activity, making it a critical step in the analysis pipeline of fMRI studies. By aligning individual subject data to high-resolution anatomical images and subsequently to standard templates, researchers can conduct detailed single-subject analyses and robust group-level studies, enhancing our understanding of the functional architecture of the brain.

4. Small Field of View registration

Small Field of View (FOV) Registration is a technique used in MRI imaging, particularly in fMRI, where the focus is on a specific area of interest within the brain. This approach allows for enhanced spatial or temporal resolution by limiting the scope of the scan to a smaller region. However, this reduction in FOV introduces challenges in the registration process, which is the alignment of these images with larger, whole-brain structural scans.

In the context of Small FOV Registration, the primary challenge is the limited information available due to the reduced FOV. This scarcity of data can hinder the effectiveness of standard registration techniques, which rely on a comprehensive view of the brain to align images accurately. To address this, a multi-stage registration approach is used, introducing an intermediary step that involves the use of a whole-brain EPI image as a bridge between the small FOV functional images and the structural scans.

The process begins with the acquisition of a whole-brain EPI image, which shares the same functional contrast and acquisition parameters as the small FOV images but covers the entire brain. This whole-brain image serves as an important link and provides a broader context that allows the alignment of the small FOV images with the structural scans.

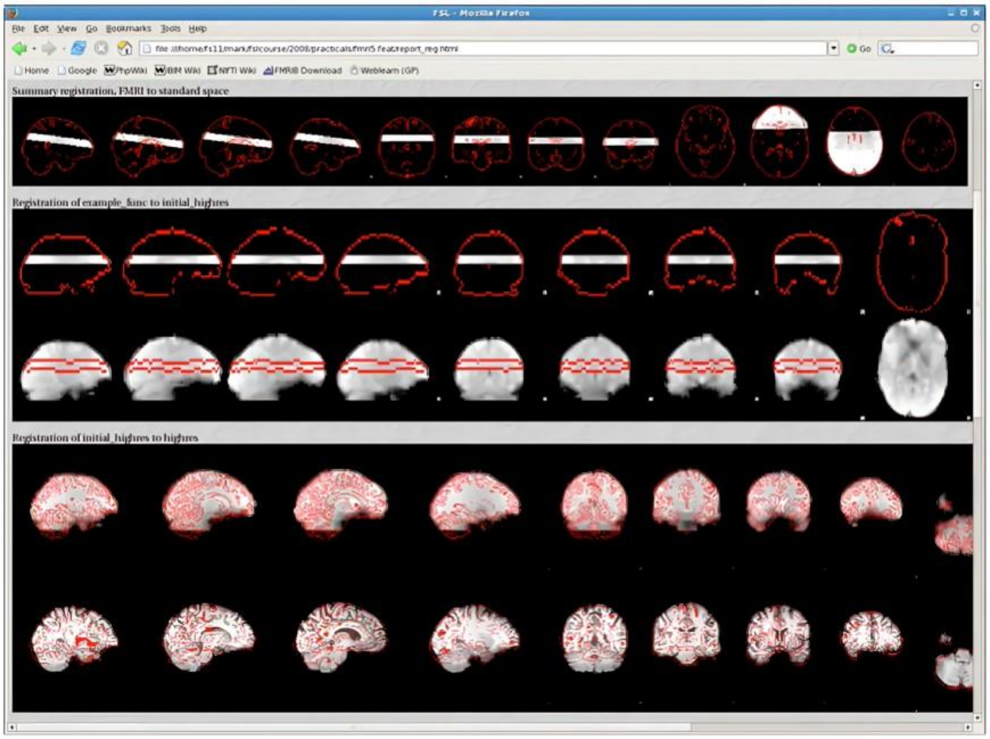


The registration process is then done in three stages:

- Small FOV to Whole-Brain EPI: In the first stage we align the small FOV images with the whole-brain EPI image. This step typically employs a translation-only registration, as the slice prescription remains consistent between the small FOV and whole-brain images. The goal here is to accurately position the small FOV within the broader context provided by the whole-brain EPI.
- Whole-Brain EPI to Structural Image: Once the small FOV images are correctly positioned within the whole-brain EPI, the next step is to register the whole-brain EPI to the structural scan. This stage uses standard registration techniques (unwarping), as it involves aligning two comprehensive views of the brain.

- Structural to Standard Space: The final stage aligns the structural image with a standard brain template, ensuring that the small FOV functional data can be accurately mapped onto a common reference space for analysis and comparison.

The following image shows the use of Small FOV Registration in FSL:


FSL - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
file:///home/fs11/mark/fslcourse/2008/practicals/fmri5 feat/report_reg.html
Go
Home Google PhpWiki BIM Wiki NIfTI Wiki FMRIB Download Webleam (GP)
Summary registration, FMRI to standard space
Registration of example_func to initial_highres
Registration of initial_highres to highres

This multi-stage approach significantly enhances the registration quality for small FOV images by providing a suitable pathway from the localized functional data to the broader anatomical context. It's important to plan and acquire the whole-brain EPI image during the imaging protocol to ensure the success of this registration strategy.

Dealing with fMRI Time-Series Problems (more resources on fMRI time series here and here)

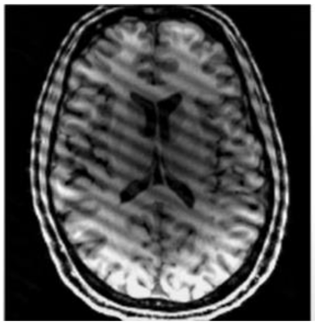
Unlike the 3D MRI data, functional MRI also has a 4th time series dimension. This difference might not seem significant at first, but we'll need multiple extra steps of fMRI preprocessing because of this. You will see why this time-series dimension of fMRI could result in huge problems, if not preprocessed properly. We also need to get rid of lots of structured noise and artefacts.

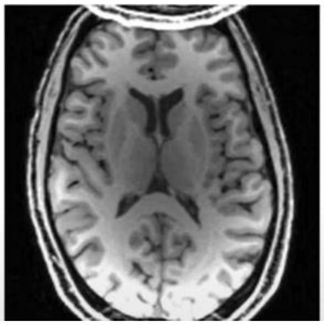
To get a taste of why time-series preprocessing is complex, remember that as time passes, the human body doesn't stay 100% constant. The patient's head would move inside the scanner. So, the fMRI images at different times are not aligned. The patient would breathe inside the scanner. This causes the brain size to change in different time-series slices. The eyes of the patient move inside the device. This would again affect the brain. Etc. etc. etc. All these problems need to be dealt with accordingly.

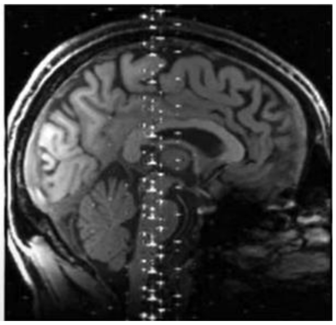
*Not all of the following steps need to be done on your data. Depending on your study, decide on the suitable combination and apply it during the preprocessing stage.

1. Reconstruction from k-Space Data

The step of image reconstruction is usually done by the scanner device. To summarize, scanners automatically reconstruct images from their native format, known as k-space, into the familiar image format using sophisticated algorithms. While these algorithms generally perform well, they might occasionally produce artifacts. Artefacts are typically easy to spot and require different approaches for resolution. The following images show some artefacts that might exist:







The left image shows **RF spiking** which manifests as irregularities in the image. This is caused by sparking or electrical faults.

The image in the middle shows **wraparound** artifacts that appear as distortions at the image edges. This is caused by insufficient field of view provided by the operator.

The right image shows **RF Interference** that occurs when external signals infiltrate the scanner room due to improper sealing or closure of doors.

RF spiking is easily fixed. There are methods that easily deal with it. On the other hand, defects in the reconstruction algorithm and wraparound artifacts are not fixable. For those, you would need a rescan. And, RF interference problem needs to be addressed depending on its location and severity. It might be fixable and doesn't cause much problem or it might be extremely troublesome.

The step of checking the data could be done automatically. Using exploratory analysis methods such as independent component analysis (ICA). But it's suggested to just spend some time manually checking the data. It doesn't take more than a minute to check each image.

*If you're working on famous publicly available datasets (e.g. ADNI), there's a good chance this step has already been done by experts before the data was made available publicly. But you might still want to make sure of this by checking the data directly.

2. Motion Artefact Correction

Even small movements from patients during scanning can have a significant impact on image quality. Motion correction is necessary because subjects inevitably move to some extent. And, despite precautions such as padding around the head, you should **always** expect some movement in every fMRI image (especially for the elderly and patients with severe diseases).

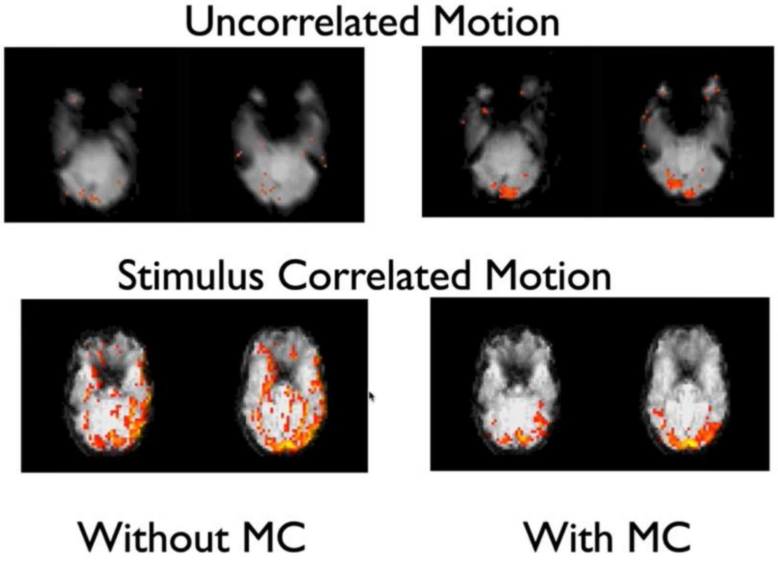
Although this movement is usually slight and not easily visible to the naked eye, it needs to be dealt with properly or it could cause serious problems in your subsequent models and analysis.

This is because, when a subject moves, even by a fraction of a voxel (the smallest unit of volume in imaging), it can cause significant changes in signal intensity at the border between brain and non-brain tissue. For example, if a subject moves half a voxel, it can induce a 50% change in signal intensity at that border. This is because the signal from the brain tissue is mixed with a signal from non-brain tissue due to the limited resolution of the imaging. (To make the explanation even simpler: Imagine looking at a point near the edge of the brain in an fMRI scan. If a patient moves slightly, this point might go from being inside the brain to being outside, or the reverse. This would cause the intensity of that point to significantly change, maybe even doubling or halving.)

Note that a voxel is 2 mm in size. So, a fraction of a voxel is, of course not visible to the naked eye.

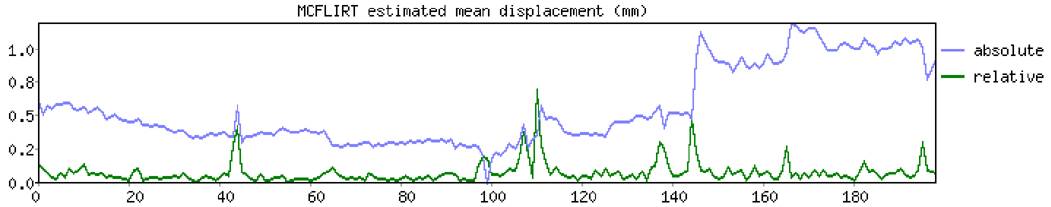
The way to do motion correction is very easy. All we need to do is to apply a bunch of different registrations. One registration is done per fMRI image. First, we select a target reference from each fMRI image. Then, we apply rigid-body registration within each fMRI image and align each fMRI image to the target reference selected (the target reference image is a 3D slice of the brain taken from a specific time of the time-series dimension). We use rigid-body registration because we only want to fix the problem of patient movement (the brain doesn't change in size or shape. It only moves – it rotates or translates). Selecting the target reference doesn't have any rules. We can take it from the beginning, the end, or from the middle of the time-series dimension. By default, many preprocessing tools select the middle 3D brain slice as the target.

The following shows the effect of Motion Correction used vs not used. The visualizations show the false positives received regarding functional brain activation.



We can observe that there is not much difference in using motion correction when we have uncorrelated motion (movement that is not synchronized with the timing of the experimental stimuli or events. Without motion correction, uncorrelated motion appears as noise, leading to minimal activation observed during analysis. Motion correction reduces noise, resulting in higher t statistics and increased activation detection). But for stimulus-correlated motion, there is a huge decrease in the false positives when motion correction is used (Stimulus-correlated motion occurs when the participant's movement coincides with the timing of the experimental stimuli or events. This means that the motion occurs concurrently with the presentation of the tasks or stimuli being investigated).

* After performing motion correction, some tools might generate a summary of motion parameters for you. For example, the MCFLRT algorithm (from FSL) provides relative and absolute motion graphs. If you want to analyze those graphs, know that relative motion graphs show motion to change from one time point to the next (this graph only shows you quick and sharp changes, not slow ones. Note that sharp changes are more problematic compared to slow ones). While absolute motion graphs show motion change from each time point to your selected target reference time point (this graph shows both sharp changes and slow drifts that happen through time). Following is an example output graph for one fMRI image:



3. Slice Timing Correction

Now, we into account the take the fact that we don't acquire all of the brain at one time. To summarize the process, in fMRI, each 3D image of the brain called a volume, is not captured all at once. Instead, it's acquired through a series of individual 2D images, or slices, taken one after the other over a period of time. These slices are then stacked together to form a 3D volume. Then, we continue this process over and over again through time. In this way, we construct a 4D fMRI image (although, some recent devices are able to take multiple 2D slices at once. But there are still no devices that could take all of the slices at the same time).

This is where the problem comes in: there exists a timing difference between when each slice is acquired within the volume. Without proper adjustment, differences in slice acquisition times can distort the data. For example, a stimulus may appear to peak earlier or later in different slices due to timing discrepancies (but in real-time, that stimulus has the same effect shape on all slices). This means we need to properly adjust the timing of each 2D slice in each 3D volume of our fMRI images to account for the delay in acquiring each slice of the brain. The process we use to deal with this problem is called Slice Timing Correction. There are multiple ways to do this step: Shifting the data using Interpolation, or Utilizing Temporal Derivatives to Shift the Model itself (this works on more simple statistical models. Not more complex ML models). I will now explain these methods.

Shifting the data using Interpolation

One way to approach slice timing correction is to first shift the data to achieve consistency between the model and the acquired data. Then, use interpolation to adjust the timing of acquired slices, this ensures alignment with the model.

Shifting the data involves adjusting the timing of each acquired slice to align them temporally with a reference point. This reference point is often chosen as the midpoint of the acquisition time series. This shift is calculated based on the known timing differences between the acquired slices. For example, if one slice is acquired 1 second after another, the data for the later slice can be shifted forward by 1 second to match the timing of the earlier slice. This way, we can ensure that the neural activity captured in each slice is aligned correctly with the experimental design.

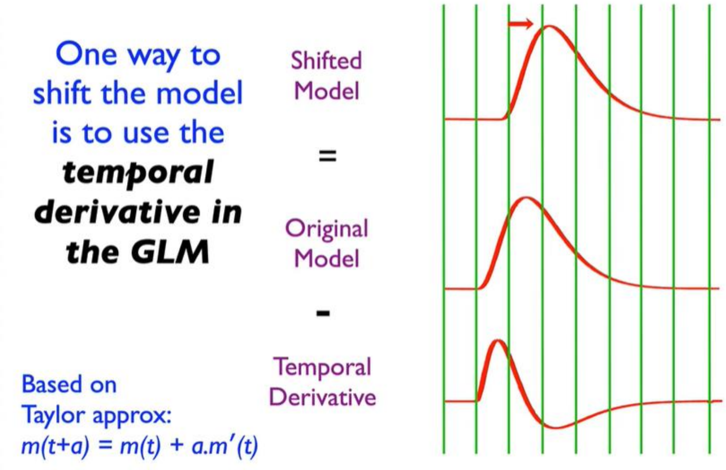
Then we use interpolation. This is the same technique we described in registration. But here, instead of doing the interpolation based on the space, we do it based on time. To interpolate the data, the signal values at specific time points within each slice are estimated based on neighboring data points. This estimation allows for the creation of a temporally adjusted signal that accounts for the timing differences between slices. We already discussed some famous interpolation techniques before. Each of which vary in complexity and accuracy, with sinc interpolation being particularly effective but computationally intensive.

However, as we mentioned in the registration part, doing interpolation usually alters the shape and quality of the signal, impacting data accuracy (this was the reason we mentioned we need to avoid resampling as much as possible).

Thus, another method is shifting the model to match the expected timing changes in the acquired slices. In this way, we can maintain the integrity of the model while compensating for timing variations.

Utilizing Temporal Derivatives to Shift the Model Itself

This one gets really mathematical so I won’t go into the details. To summarize, temporal derivatives refer to the rate of change of a signal over time. They capture how the intensity of the signal varies with time, indicating the direction and magnitude of changes. One way to use temporal derivatives is to add them into our model (usually GLMs. But there are some ways to incorporate temporal derivatives inside ML/DL approaches as well) framework, which is a common statistical approach used in fMRI data analysis. In the GLMs, by including temporal derivatives as additional regressors, we can effectively adjust the timing of the model to match the timing of the acquired data. These regressors capture the temporal dynamics of the experimental design, allowing the model to flexibly adjust to timing variations in the acquired data (the process is automatically done by the GLM model as it’s trained).


One way to shift the model is to use the **temporal derivative in the GLM**
Based on Taylor approx: $m(t+a) = m(t) + a.m'(t)$
Shifted Model
=
Original Model
-
Temporal Derivative

The above image shows a brief summary of the process. Basically, the temporal derivative shows us the slope derivative) function of the signal. If we add this to the original GLM model (based on the Taylor approximation formula) and train the new GLM model, we observe that the resulting GLM would be automatically ‘shifted’ in time for each 2D slice in a suitable manner. And the slice timing issue would be dealt with.

4. Motion non-rigid-body artefact problems

Now, we go back to dealing with problems of motion again. Remember that in the motion artefact correction step we dealt with the problems arising from the rigid body motion between different 3D brain images as we move through the time-series dimension. But, what about the effects of the patient’s movement between the 2D slices within each 3D brain image?

We already explained that each of the 3D brain images of an fMRI image is taken at different times. This means that when some 2D slice is taken, the brain would have some motion until the next 2D slice would be taken. This means the 2D slices that construct a 3D brain image are not actually parallel as they should be! Also, there are other motion artifacts that occur when a subject moves in the scanner such as spin history (the phenomenon where a spin (a property of atomic nuclei) transitions from one slice to another during the acquisition process) and susceptibility effects (Susceptibility effects arise from variations in the magnetic susceptibility of tissues within the brain. Different tissues have different magnetic properties, which can cause distortions in the magnetic field during fMRI scanning).

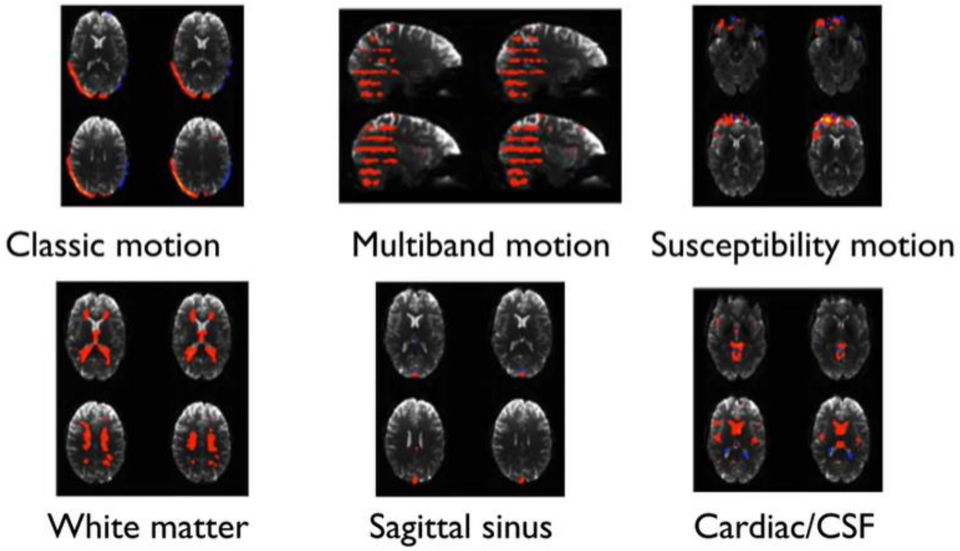
To summarize, just one step of motion artefact correction is not sometimes enough. Because motion artefact correction only takes the general rigid-body movements into account. But the non-rigid-body effects of motion on the data. Such artefacts severely degrade functional results. This is usually worse for stimulus-correlated motion.

Some potential analysis remedies exist for cases where you feel motion artefacts cause a problem:

- Motion Parameter Regressors in GLM: This method involves adding motion parameters (mainly the 6 rotation and translation parameters that are calculated by algorithms such as MCFLIRT. Non-linear parameters could also be included) as regressors in the model during statistical analysis. By including these parameters, the model can account for motion-related variance, effectively regressing the effects of motion from the data. This is similar to the temporal derivative approach we discussed previously. This method has the disadvantage that we assume our parameters have a linear effect on the data.

- Removing Artefacts with ICA Denoising: Independent Component Analysis (ICA) is a data-driven technique used to separate signal from noise in fMRI data. What ICA does is decompose the data into a set of independent components that explain the interesting structure of the data (they could be about the neural signals or the noise). Each component would have a spatial map and a time series associated with it. Then, by looking at these two elements of each component, ICA helps us identify and remove components related to motion artifacts, leaving behind the neural signal of interest. This method is especially useful on resting-state fMRI. But the advantage is that in addition to motion noise, it also helps us identify other types of noises such as physiological-related noise, and other MRI-related artifacts. The disadvantage of this method is that it is much more time-consuming compared to other methods.: first ICA should be applied, and then the noise components should be identified, either manually or by specific software tools. Then, those components should be removed and the remaining components should be added together again. Allowing you to preprocess your images one by one and get the new dataset.

The following image shows how some ICA components might look like:



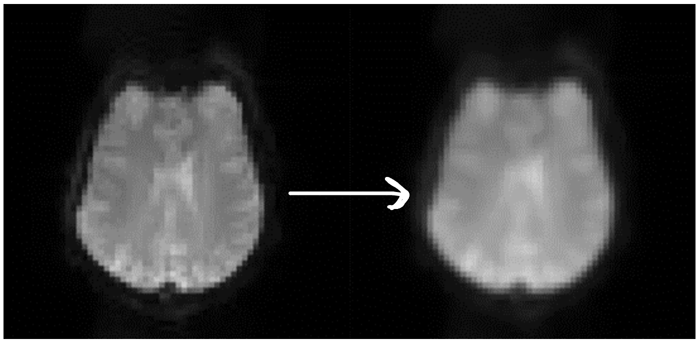
- Outlier Timepoint Detection and Exclusion: Outlier detection involves identifying time points in the fMRI time series that deviate significantly from the expected pattern of neural activity. These outliers may result from motion-related artifacts or other sources of noise. By employing outlier detection techniques within your GLM models, such as robust regression or Studentized residuals, you can identify and exclude problematic time points from the analysis. This step is especially useful if we have sudden motions (big jumps). Such cases are almost guaranteed to be detected by this outlier approach. (in this method, we select a threshold and if our change at some time point gets larger from the selected threshold, we consider it to be an outlier. And the effect would be removed from the data.)

- Rejection of Subjects Displaying "Excessive" Motion: In cases where motion artifacts significantly compromise the quality of the data, you may just exclude the image from analysis.

*These methods are not always needed. We only need them in cases where non-rigid-body effects are large. Because our goal is not to remove all motion artefacts from our data. Our goal is to remove the problematic and large motion artefacts (e.g. our goal is not removing small motions of 0.01 mm size. Our goal is removing larger artefacts of 1~2 mm)

5. Spatial Filtering

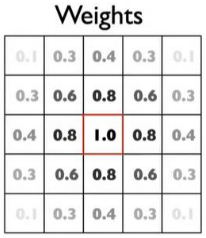
Spatial filtering, also known as spatial smoothing or spatial blurring, is a common preprocessing step we use in fMRI data preprocessing. This step is pretty straightforward compared to previous steps. All we do in this step is to blur our images. This step probably sounds weird when you first hear it. After all, the spatial resolution of fMRI images is very low by itself. Why would we want to make it worse?



There are two main advantages to this:

- Increasing the Signal-to-Noise Ratio (SNR): One of our main reasons for spatial filtering is to enhance the SNR of the fMRI data. The SNR represents the ratio of the strength of the signal (i.e., brain activation) to the level of background noise. By applying spatial filtering, we will reduce the existing noise in the data through a process of averaging neighboring voxels (spatial filtering is simply taking a weighted average of neighboring voxels. First, we define a Gaussian kernel that holds the weights. Gaussian smoothing assigns these weights to neighboring voxels based on their distance from the central voxel. Then, we take the average using those weights). This reduction in noise relative to the signal enhances the overall detectability of brain activations. (check here and here)

A simple Gaussian Kernel:



Weights

0.1	0.3	0.4	0.3	0.1
0.3	0.6	0.8	0.6	0.3
0.4	0.8	1.0	0.8	0.4
0.3	0.6	0.8	0.6	0.3
0.1	0.3	0.4	0.3	0.1

- Minimum Smoothness Requirement for Gaussian Random Field Theory (GRFT): Another reason to use spatial filtering is to ensure a minimum level of "smoothness" in the fMRI data, which is necessary for certain statistical analyses later on, particularly those based on Gaussian random field theory (GRFT). GRFT is a statistical framework commonly used for thresholding fMRI data to identify significant brain activations. “Smoothness” of our images refers to the spatial continuity of the fMRI signal, which helps in estimating the underlying noise distribution accurately (Basically, When the fMRI signal is smooth, it means that nearby voxels tend to have similar values, this means we expect a gradual transition in neural activity patterns). Without sufficient smoothness, the assumptions of GRFT may not hold, giving us unreliable statistical inference results.

However, you should remember that although Spatial Filtering is very important, it still blurs the image and affects the image resolution. Especially, blurring the boundaries between activated and non-activated regions causes smaller activation areas to become less distinct or even entirely obscured. There are two ways to approach this:

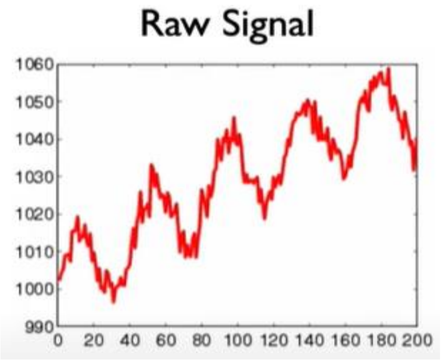
- You can choose to skip the spatial filtering step entirely: if you plan to use a model that is robust to the noises of the data, you might decide to skip this step.
- Or, to balance the benefits of noise reduction with the preservation of spatial detail, the recommended approach is to apply a modest amount of spatial filtering. This involves using a Gaussian kernel with a relatively small FWHM (Full Width at Half Maximum) parameter. Usually, an FWHM around 3~6 should be safe enough.

6. Temporal Filtering

Temporal filtering works by looking at the time-series within each voxel. There is a large amount of information contained within the time series of each voxel. Some of which are not related to the pure brain activation that is of interest. So, we first need to discern various components contributing to the signal. Then, we need to remove the components that are not related to the signal of the brain itself directly. This process is known as Temporal Filtering.

When we check the time-series of each voxel, in addition to the main brain signal that we need, they also contain scanner-related signals and physiological signals. All three of these could have both high and low-frequency components. For example, the image below shows us the time-series of a voxel’s signal. We can see a general increasing trend but with a high frequency of up and down movements at the same time. The fact that there is a general increasing trend might be because of the scanner. On the other hand, we see a constant up and down movement from the signal. This one might be because of physiological elements. For example, it might be because of the patient’s breathing. When the patient breaths, the brain gets slightly affected. This would result in the brain signal going through a temporary increase. Or this up and down movement might also be affected by the heart’s cardiac cycle. Etc. etc.

etc. What matters here is to identify, and separate these signals without removing the main signal of interest.



To be more detailed, the **low-frequency** movement in time-series (shown by a slow, gradual change in intensity over time) mostly stems from scanner drift, caused by factors like component heating. While physiological factors such as respiratory and cardiac cycles can contribute to slow changes in signal intensity. For instance, subtle variations in the breathing cycle or cardiac rhythm can induce gradual fluctuations in the recorded signal, mimicking a slow trend.

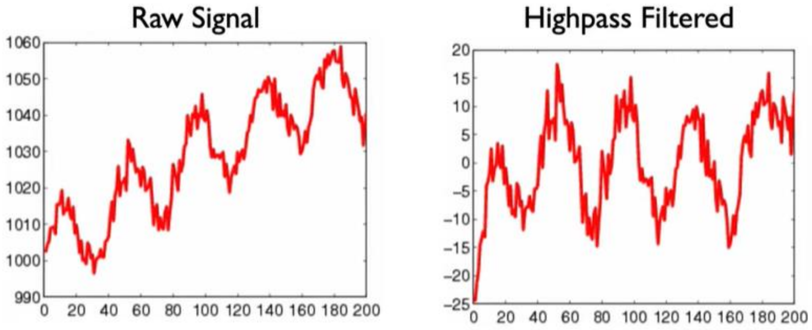
On the other hand, another component of the time series exhibits fast, **high-frequency** signal movements. These variations are more challenging to characterize because of the limited sampling rate used in fMRI data acquisition. While such movements usually arise from physiological processes like breathing and cardiac activity, they can also be the result of scanner-related artifacts.

Note that our signals of interest are contained within these high and low-frequency signals. So, we can't 'just remove' all of them. In that case, we would lose a great amount of information.

There are generally two ways to approach this (depending on our study): High Pass and Low Pass. Now, I will discuss these 2 filtering methods:

Temporal Filtering: High Pass

In high pass filtering, we only remove low-frequency signals and we pass from the high frequencies. In the following graphs, we can see how high pass filtering works. The filtered signal now no longer shows any signs of the increasing, linear-shaped, trend. Instead, all we have left is the up and down high-frequency trend. Which was most likely caused by the breathing and cardiac activity.



When we do this type of filtering, it is important to specify a cutoff frequency that is appropriate for the type of design that we need. The cutoff frequency is the parameter that determines the boundary between retained and attenuated frequency components. If we use a cutoff frequency that is smaller than the frequencies of our signals of interest, the algorithm would by mistake remove the neuronal activity signals we were interested in.

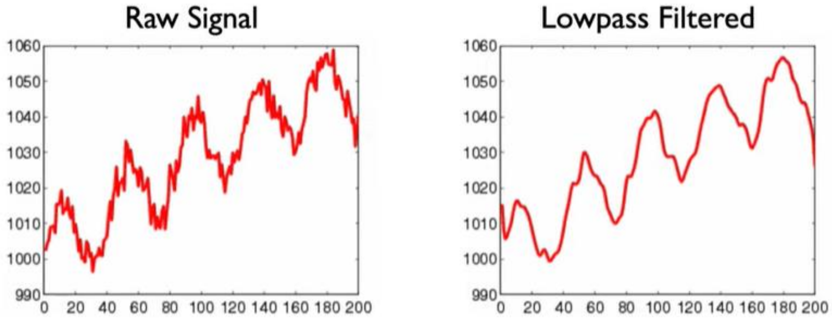
In fMRI, the cutoff frequency is usually derived from the experimental design and the expected frequency of neuronal responses to stimuli. You might also calculate a suitable cutoff frequency by analyzing the predicted responses obtained by convolving stimulus timing with the hemodynamic response function (HRF).

Although, the preprocessing software you use should be able to calculate and suggest a suitable cutoff. That cutoff usually works well enough if you don't have any other cutoff value in mind.

Temporal Filtering: Low Pass

This step is similar to the previous step, but the opposite. We keep removing the high-frequency noise, while we keep the low-frequency signals.

Similar to the previous filtering method, we need to be careful about our cutoff frequency.



This method is not usually recommended. low pass filtering could only be used if we know that our predicted model does not contain low frequencies.

*There is a third type of temporal filtering approach called bandpass that combines the above methods and removes both low and high-frequency noise. But this is not recommended because we might lose a large number of signals. Many of which might have been from the neural activities of our interest. (This is especially true for the more uniform-looking signals. The example above is complex and easy to classify. But not all signals are like this. When our signals are more uniform, using low pass or band pass causes many useful signals to be removed by mistake.)

*Another problem with using low pass or band pass is that removing too much noise would cause problems in the validation of your statistical analysis. Because calculating the correct autocorrelation would be hard in such a situation.

So, it is suggested to only use high pass. Just make sure:

- Ensure the cutoff frequency is higher than model frequencies
- Also put a lower limit on the cutoff frequency for good autocorrelation estimation (e.g. for TR=3s, cutoff period > 90s)

7. Smoothing

It is common to smooth the functional data or replace the signal at each voxel with a weighted average of that voxel's neighbors. This may seem weird when you first hear it. After all, why would we want to make the images blurrier than they already are?

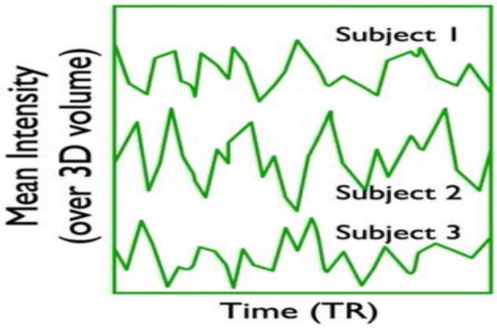
Smoothing indeed decreases the spatial resolution of your fMRI, which is a disadvantage. But there are benefits to smoothing that sometimes outweigh its drawbacks. For example, we know that fMRI data contains a lot of noise and that the noise is frequently greater than the signal. By averaging over nearby voxels, we can cancel out the noise and enhance the signal.

Another advantage of smoothing is when we are dealing with statistical analysis. To summarize, during group analyses in which all of the subjects' images have been normalized to a template, a decrease in noise could be beneficial. Because, although each subject's fMRI image will be transformed to match the general shape and large anatomical features of the template, there will be variations in how smaller anatomical regions align among the normalized functional images. If the images are smoothed, there will be more overlap between clusters of signals, and therefore greater likelihood of detecting a significant effect.

*Remember from the registration section that normalization is just doing registration when we want to map our images into a standard space such as MNI 152.

8. Global Intensity Normalization

One last problem of fMRI data that needs to be dealt with it's the fact that fMRI data lacks quantitative consistency, resulting in varying mean levels across subjects. The mean signal level per subject holds no essential relevance to the stimulus we are working on. Global intensity normalization aims to ensure consistency across subjects while preserving data structure. The following image shows an example:

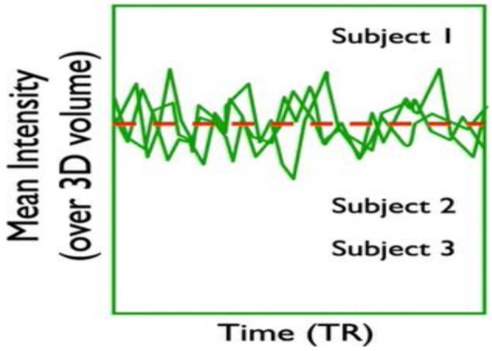


To summarize, if two patients show identical neural signals from their brains at some time point, the fMRI might not assign both of these signals the exact same value. This problem needs to be solved and we need to create a quantitative consistency between different subjects. This is why we use Global Intensity Normalization as our last preprocessing step (Note that we can **NOT** use this step earlier than other fMRI preprocessing steps are done. Because to generate a quantitative consistency between different images, we need to compare the pure neural signals of interest. Thus, all other artefacts should already be removed. And our images better all be in the same standard space).

The way this step gets done is pretty straightforward:

- First, we determine (calculate) a mean level across all voxels and time points for each subject.
- Then, we use the calculated mean levels to calculate the mean and variance.
- At the end, we scale each fMRI image based on those values. Ensuring that the mean level across all spatial and temporal dimensions has become uniform.

This process ensures that the mean value for each subject's dataset is consistent across the study group. Like the below example:



9. Summary

A short summary:

- Reconstruction: Create an image and remove gross artefacts
- Motion Artefact Correction: Get consistent anatomical coordinates (always do this step)
- Slice Timing: Get consistent acquisition timing (use temporal derivative if possible)
- Spatial Filtering: Improve SNR & validate GRF
- Temporal Filtering: Highpass: Remove slow drifts | Lowpass: Avoid it if possible
- Global Intensity Normalization: use it on all 4 dimensions. It keeps overall signal mean constant across different images

Preprocessing: Resting-State fMRI

Until now, we covered some of the main techniques used in the preprocessing of MRI and general fMRI data. But from here on, we take the fact into account that fMRI data itself is categorized into 2 main groups:

1. Task-Based fMRI

Task-based fMRI is one of the most common approaches in functional neuroimaging. In this method, participants are asked to perform specific cognitive or motor tasks while their brain activity is being measured. The tasks are designed to engage particular brain regions or networks associated with the function being studied. For example, a language task might involve reading or listening to words to activate regions involved in language processing.

The core principle of task-based fMRI is the cerebral blood flow and blood oxygenation level-dependent (BOLD) response to neural activity. When a specific brain region is more active, it consumes more oxygen. The body responds by increasing blood flow to that region, which changes the magnetic properties of the blood. fMRI detects these changes, allowing researchers to infer which parts of the brain are involved in the tasks.

This approach is powerful for understanding the brain's functional organization, identifying which areas are involved in different cognitive processes, and how these areas interact during task performance. It's particularly useful in clinical research for understanding how diseases or injuries affect brain function and in cognitive neuroscience for mapping brain activity related to specific tasks.

2. Resting-State fMRI

Resting-state fMRI, on the other hand, does not involve any explicit task. Instead, participants are usually asked to lie still, often with their eyes closed or fixed on a point, and let their minds wander without focusing on anything in particular. This method investigates the brain's activity when it is at rest, not engaged in any focused mental activity.

Despite the lack of a task, the brain is never truly "inactive." Even in a restful state, various brain regions continue to communicate and coordinate, reflected in spontaneous fluctuations in the BOLD signal. By analyzing these fluctuations, researchers can identify networks of regions that consistently show synchronized activity, known as resting-state networks. These networks are thought to reflect the brain's intrinsic functional architecture, providing insight into how different regions of the brain are connected and cooperate, even in the absence of external tasks.

One of the key advantages of resting-state fMRI is that it can be used with individuals who might not be able to perform specific tasks, such as infants, young children, or patients with severe neurological disorders. It also avoids the complexities related to task design and allows for the examination of the brain's default network, which is active during rest and implicated in self-referential thought and mind-wandering.

Now, we will cover some of the more rs-fMRI-specific data processing methods. Note that, similar to other sections, you don’t need to use all these methods. Just selecting some of them (preferably over 3) and using them would work well enough.

1. **Single subject ICA**

   Single-subject Independent Component Analysis (ICA) is a computational method applied to fMRI data to uncover underlying hidden factors or components that are statistically independent from each other. At its core, ICA is predicated on the assumption that the observed fMRI signals are linear mixtures of various independent sources, and the goal is to segregate these sources to extract meaningful patterns of brain activity and artifacts.

   In the context of single-subject fMRI data, ICA is performed on the preprocessed time-series data obtained from the MRI scans of an individual. The high-dimensional fMRI dataset, typically consisting of voxel-wise time-series, is complex and multifaceted, with signals arising from both neural and non-neural origins. ICA approaches this dataset without any prior knowledge or assumptions regarding the source or number of signals, aiming to decompose it into a set of spatially independent patterns (spatial maps) and their associated time courses.

   Technically, ICA starts by creating a matrix where each row represents a voxel’s time series and each column corresponds to a time point in the scan. The number of independent components to be estimated is a crucial parameter and can be determined using heuristic algorithms that consider the data's complexity or may be set a priori based on the researcher's discretion.

   ICA then seeks to factorize the original signal matrix into two matrices: one representing the independent components (the spatial maps), and the other the weighting (time courses) of these components across the time series. This factorization is not trivial due to the inherent ambiguity in the solution—there are potentially infinite ways to decompose the signals into independent components. ICA circumvents this issue by relying on the statistical independence of the time courses of the components and often uses higher-order statistics (beyond variance and covariance) to enforce this independence.

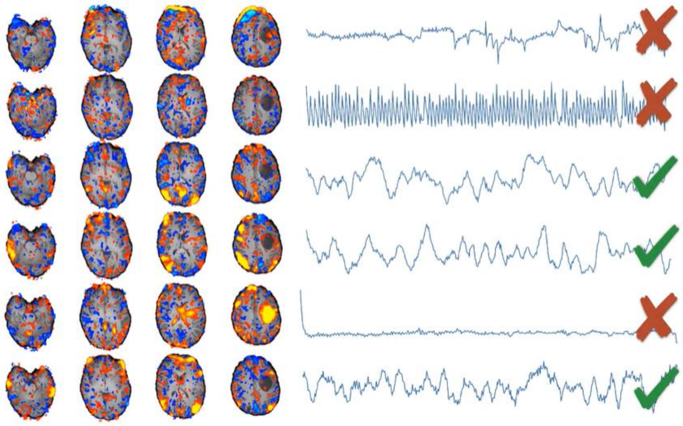
The algorithms used to perform ICA, such as FastICA or Infomax, typically involve iterative processes that maximize the statistical independence of the output components. They employ measures such as negentropy or mutual information and adjust the weights iteratively until the outputs—the independent components—achieve maximal statistical independence from each other. This independence is often measured in terms of non-Gaussianity, as per the central limit theorem, the sum of independent non-Gaussian variables tends to be more Gaussian than the original variables themselves.

Once ICA converges, the resulting components each contain a spatial map, indicating the voxel-wise contribution to that component, and a time course, representing the temporal characteristics of the component's contribution to the overall signal. The spatial maps can reveal distributed networks within the brain, corresponding to regions that share functional connectivity. The time courses reflect the activation patterns of these networks over the course of the scan.

The components identified can represent meaningful neural activity, such as those corresponding to known brain networks like the default mode or motor networks, but can also represent noise. Noise components might be due to head motion, cardiac and respiratory cycles, scanner artifacts, or other physiological processes not related to the neuronal activity of interest.

The ability of single-subject ICA to disentangle these sources is particularly powerful for resting-state fMRI data, where the lack of an experimental design makes it challenging to separate the brain's intrinsic activity from noise. The end goal of ICA in this setting is to enhance the interpretability of the fMRI data by isolating and subsequently removing the components attributable to artifacts, thereby refining the analysis of neural connectivity and function.

Here we have a sample result of ICA diagnosing:



This visualization is organized into columns representing different components extracted from the data, with each component followed by its spatial map (the colorful brains!) and time series, and by the red cross or green check in the end we can see whether the component is considered noise (cross) or signal (check).

Spatial Maps: These maps show the spatial distribution of brain areas that are involved in each component. The color scale typically signifies the intensity or weight of the component at each voxel, with warmer colors often indicating stronger involvement in the component. For example, components that are spatially confined to gray matter regions are likely to represent neural activity, while those that show artifacts around the edge of the brain or ventricles are often considered noise.

Time Series: To the right of each spatial map we can see the time series plot for the respective component. These plots show us the fluctuation of the component's signal over the time course of the fMRI scan. A time series characterized by low-frequency and smooth oscillations show actual neuronal activity. In contrast, erratic, spike-laden, or high-frequency patterns are suggestive of noise, such as motion or physiological artifacts.

In the end, we can see the classification of each component as noise or signal. This classification is essential for denoising the data, as noise components are to be regressed out or otherwise removed from the dataset to enhance the clarity and interpretability of the fMRI data for subsequent analysis.

2. **Seed-based correlation analysis**

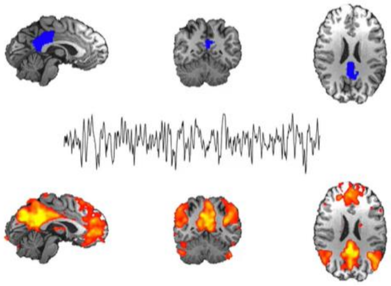
Seed-based correlation analysis is a commonly used method in rsfMRI studies to evaluate functional connectivity. This technique involves selecting a region of interest within the brain, often referred to as the "seed," and then assessing the degree of synchrony or correlation between the seed's activity and the activity throughout the rest of the brain. The main outline of this method is as follows:

Selection of the Seed Region: First, we choose a specific brain area based on our research question or hypothesis. The choice of this region is critical, as it will affect the connectivity pattern that emerges from the analysis. This seed could be a region known to be involved in a particular function or a hub within a known network.

Extraction of Time Series Data: Then, the average fMRI signal within the seed region is extracted, compiling a time series that represents the neural activity in that area over the course of the scan.

The extracted time series is then correlated with the time series from every other voxel in the brain. This is done to establish a correlation map that represents the functional connectivity of the seed region to all other regions.

The resulting map displays a value for each voxel indicating the strength of its correlation with the seed. Positive values indicate a positive correlation, meaning that as the seed region's activity increases or decreases, so does the activity in that voxel. Negative values, conversely, indicate an inverse relationship.



It should be noted that one of the challenges of seed-based analysis is the potential for bias based on the seed location. Small variations in seed placement can lead to significantly different connectivity patterns. In addition, although seed-based correlation analysis is straightforward and intuitive, it simplifies the brain's complex organization and does not account for the influence of other networks, potentially conflating different sources of connectivity into a single map.

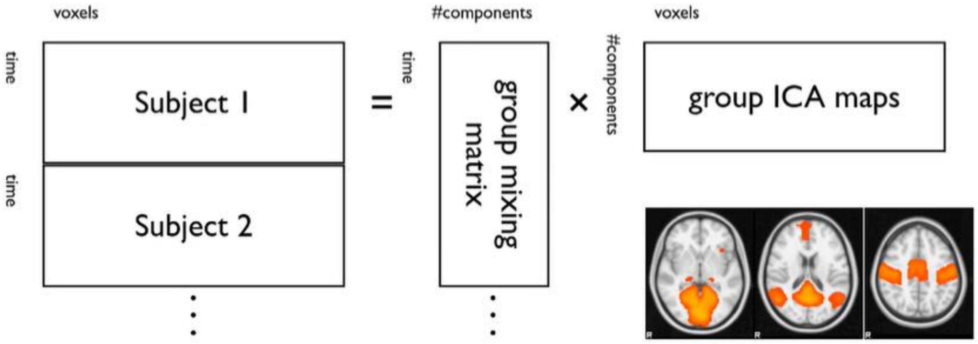
Because of this, although this method is suitable for hypotheses that are centered on the function or pathology related to specific brain regions but may not be the best fit for exploratory analyses where the global network organization is the focus.

3. Group-ICA networks

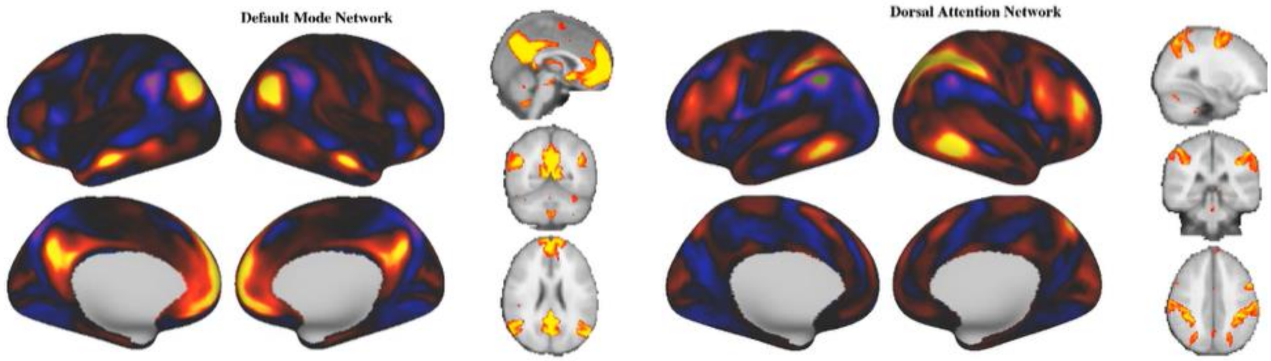
Group-level Independent Component Analysis (Group-ICA) is an advanced neuroimaging method employed to explore shared neural patterns across multiple subjects within fMRI datasets. When applied to resting-state fMRI, Group-ICA is particularly potent, as it uncovers key brain networks that are consistent across a population, providing insights into the common functional architecture of the human brain.

The central premise of Group-ICA is that while each individual's brain activity is unique, there are patterns of activity or networks that are universally present. These could be networks involved in basic functions such as sensory processing or higher-order functions like attention or memory. Group-ICA tries to identify these underlying common networks by analyzing data across a group rather than on an individual basis.

To get a little technical, Group-ICA involves concatenating preprocessed fMRI data from all subjects into a single large dataset, where the temporal dimension now includes the combined scans of all subjects. This merged dataset is then subjected to ICA, which assumes that the recorded signals – now a combination of signals from different brains – are mixtures of statistically independent sources. The ICA algorithm works to reverse this mixing, aiming to return the separate source signals. The end result is a set of spatial maps and associated time courses that represent the group-derived components. These components reflect the consistent patterns of brain activity or networks shared across the study population.



A key aspect of Group-ICA is the reliance on the statistical independence of the derived components, which means that each component's time series should not be able to predict another's. These independent components often reveal resting-state networks (RSNs), which are sets of brain regions that exhibit synchronized activity when the brain is not engaged in any directed, externally focused task. RSNs discovered via Group-ICA, such as the Default Mode Network, are considered fundamental to understanding the brain's functional organization.



Note that we an important step following Group-ICA is the interpretation and validation of the identified components. Not all components correspond to neural networks; some may represent noise or artifact. Thus, we must manually examine the components to discern which are signal and which are noise. This classification can be conducted through visual inspection, relying on the neurobiological plausibility of the spatial maps, the consistency of the time courses, and the component’s frequency spectrum.

After identifying RSNs through Group-ICA, researchers typically employ dual regression or similar techniques to map these group-level networks back to individual subjects. This step allows for the examination of inter-individual variability within the identified networks and the investigation of relationships between network expression and behavioral or genetic variables.

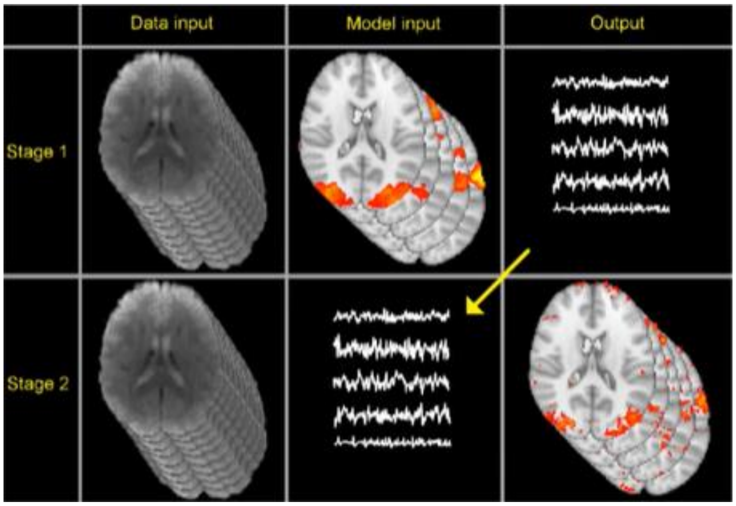
In the end, note that Group-ICA’s power lies in its ability to leverage the multidimensional nature of fMRI data across subjects to explain the shared neural dynamics. However, the method requires careful consideration of numerous factors, including the selection of the number of components, the handling of signal vs. noise classification, and the implications of the findings for understanding brain function and its variations across different conditions or populations.

4. Dual regression group analysis

Dual regression is a two-stage regression process used in fMRI to identify subject-specific spatial maps of resting-state networks (RSNs) that are consistent across a group of individuals. The technique is generally applied following group-level Independent Component Analysis (Group-ICA), which identifies common patterns of brain activity across a group of subjects. Dual regression allows for the examination of these group-level patterns within individual subjects, allowing comparisons and statistical analyses across individuals or groups.

The first stage of dual regression involves spatial regression, which takes the group-level spatial maps obtained from Group-ICA and uses them as spatial regressors in a general linear model (GLM) applied to each subject's individual 4D fMRI data. This step essentially asks where in each individual's brain data can we find patterns of activity that look like the group-level patterns. The outcome of this spatial regression is a set of subject-specific time series, one for each group-level spatial map, reflecting how the activity in each network evolves over time for that individual.

Once the time courses have been extracted for each individual, the second stage of dual regression comes into play. Here, temporal regression is applied: the subject-specific time series obtained from the first stage are used as temporal regressors in another GLM, now applied to the same individual's 4D fMRI data. This step essentially maps out where each subject-specific time course is expressed across the brain. This results in a set of subject-specific spatial maps corresponding to each of the group-level RSNs. These maps show where in the brain the fluctuations of each time course are found, allowing researchers to quantify the expression or connectivity of each network within individuals.



One of the key strengths of dual regression is its ability to retain the individual-specific details of brain activity while also allowing for group-level inferences. It can provide insights into individual differences in brain network architecture and how these relate to behavior, genetics, or disease state. For example, dual regression has been widely used in clinical research to understand how brain network connectivity differs between patients with neurological or psychiatric conditions and healthy controls.

Despite its utility, dual regression does have limitations. The accuracy of the individual spatial maps generated in the second stage of dual regression is contingent on the quality of the group-level components identified by Group-ICA. Moreover, the method assumes that the Group-ICA components are largely similar across individuals, which may not always be the case. There can be subtle but important differences in how RSNs are represented across different people, and dual regression may not capture this variability fully. Care must be taken when interpreting the results, and the technique is often complemented with other analyses to confirm findings.

5. **Node parcellations (this method has much more detail than I -badly- summarized here. For a more comprehensive review, check out this tutorial)**

Network Modeling Analysis in the context of brain imaging is a computational framework designed to understand the complex interconnections within the brain, often referred to as the brain's "connectome." This framework encompasses several stages, with Node Definition and Edge Calculation being important in the construction and analysis of brain networks.

Node Definition is the process of determining the regions of interest within the brain that will serve as the primary units or nodes in the network model. These nodes are typically chosen to represent specific brain regions or clusters of voxels with a functional or anatomical basis. The granularity of these nodes can vary greatly depending on the scale of the analysis and the resolution of the data. Nodes could represent individual brain regions defined by anatomical atlases, or they could be functionally defined regions that are identified based on their activity patterns during the task or resting states.

The selection and definition of nodes are critical because they set the stage for all subsequent analyses. Nodes must be defined so that they are meaningful in the context of the research question and consistent with the spatial resolution of the imaging modality used. There are several methods for defining nodes, ranging from using predefined anatomical or functional regions to employing data-driven approaches such as clustering or Independent Component Analysis (ICA) that define nodes based on the patterns found in the data itself.

Edge Calculation is the next step and refers to the quantification of the relationships or connections between pairs of nodes, which are termed "edges" in network analysis. The primary method for calculating edges in functional neuroimaging is to compute the statistical association between the time series of brain activity from two nodes. Pearson correlation coefficients are commonly used to measure the linear relationship between two nodes' activity, where the strength of correlation is taken as a proxy for the strength of connectivity.

Edges can represent various types of connections, including anatomical connections (physical white matter tracts measured through diffusion MRI), functional connections (synchronous activity measured through fMRI), and effective connections (the influence one region exerts over another). While the Pearson correlation can provide a measure of functional connectivity, it does not infer directionality or causality. For understanding causal relationships or influence, methods such as Granger causality or dynamic causal modeling might be employed, although these approaches have limitations and are subject to certain assumptions that may not always hold true for fMRI data.

Partial correlation and network-based statistics can be used to remove indirect connections and emphasize direct connections between nodes, but these methods require careful application. They involve controlling for the activity of all other nodes in the network when calculating the edge strength between two nodes, which can sometimes lead to false negatives or false positives if not implemented with consideration for the network's complexity and the quality of the data.

The result of Node Definition and Edge Calculation is a graph or a matrix that represents the brain's network, with nodes as the graph's vertices and edges as the lines that connect them. The characteristics of this graph, such as its density, modularity, and the centrality of its nodes, can provide helpful information to us about the brain's functional architecture. Network Modeling Analysis can reveal how brain regions coordinate activity, how this coordination may be disrupted in disease, and how it might be modulated by various cognitive or behavioral states.

To summarize, Network Modeling Analysis is a comprehensive approach that translates complex brain imaging data into a structured representation of brain connectivity. This representation can then be used to check and understand the functional organization of the brain, both in health and disease.

6. **Subject and Group Network Modelling Analysis (if you're interested, more details could be found here)**

Network modeling analysis in the context of neuroimaging is an approach used to understand and quantify the complex web of interactions between different regions of the brain. In the application of this approach, especially in rsfMRI, the primary goal is to map and analyze the functional connectivity networks that underpin various cognitive processes and brain states. In the following, I will describe some of the key steps in this approach:

In network modeling analysis, especially when dealing with fMRI data, the process depends heavily on how we define nodes and calculate edges within a brain network. Nodes are essentially points in the network that represent specific brain regions or groups of neurons, and edges are the connections between these nodes, indicating some form of relationship or interaction.

Defining nodes can be done in various ways, but it often involves segmenting the brain into distinct areas based on either anatomical landmarks or functional characteristics. For instance, we might use an anatomical atlas to divide the brain into predefined regions like the prefrontal cortex or hippocampus, assigning each region as a node. Alternatively, functional criteria might group voxels showing similar activity patterns during certain tasks or resting states, using clustering algorithms or Independent Component Analysis (ICA) to identify functionally coherent regions. The choice of node definition method can significantly impact the network's subsequent analysis and interpretation, making it a critical step in the process.

Once nodes are established, the next step is to determine the edges, which quantify the relationship or interaction between each pair of nodes. In functional connectivity studies, this often involves calculating the statistical correlation between the time series of brain activity in each pair of nodes. A high correlation suggests that the two regions are functionally connected, possibly part of the same network. However, simply using correlation can be misleading because it doesn't distinguish direct from indirect connections. For a more nuanced view, techniques like partial correlation or graphical lasso might be used to eliminate spurious connections and highlight more meaningful relationships.

A significant challenge in group analysis is ensuring consistency in node definitions across individuals, whose brain anatomy and functional organization can vary. One common approach is to normalize individual brain data to a standard template before analysis, but this can sometimes warp individual-specific features. Another challenge is dealing with the inherently indirect and noisy nature of fMRI data, where non-neural factors like participant movement or physiological processes can contaminate the signal. Preprocessing steps, including motion correction and spatial smoothing, are crucial for minimizing these effects, but they must be carefully calibrated to avoid introducing new artifacts or losing relevant information.

All in all, note that network modeling analysis in neuroimaging is a technically demanding process that requires careful attention to how nodes are defined and how connections between them are quantified. The choices made at each step can significantly impact the results, making it essential to understand the underlying principles and potential pitfalls of different methods.

7. Nuisance Regression

Nuisance regression is a crucial preprocessing step in resting-state fMRI data analysis, aimed at removing unwanted variance from the data that is not related to neural activity of interest. This process is essential because resting-state fMRI does not rely on a task-based model to explain expected brain activity. Instead, it examines the natural similarities in the time courses of different brain regions' signals. Such analyses are particularly sensitive to various sources of noise, such as head motion, physiological processes (like cardiac and respiratory cycles), and scanner artifacts.

In resting-state fMRI, the absence of a task means that any movement or physiological process can easily introduce similarities between the time courses of different brain regions, potentially confounding true neural connectivity signals. For instance, head motion can cause widespread, synchronous signal changes across the brain, mimicking functional connectivity. This is especially concerning in studies comparing different populations, such as patients and controls, where motion-related differences might be mistaken for genuine neural differences.

Nuisance regression involves identifying and removing signals in the data that can be attributed to non-neural sources. For this purpose, similar to ant regression model, we first need to decide on suitable regressors. Some such regressors include:

- **Head Motion Parameters**: Movement-related artifacts are a major source of spurious variance in fMRI data. Six parameters representing rigid-body head motion (three translations and three rotations) are typically included as regressors in the nuisance regression model.

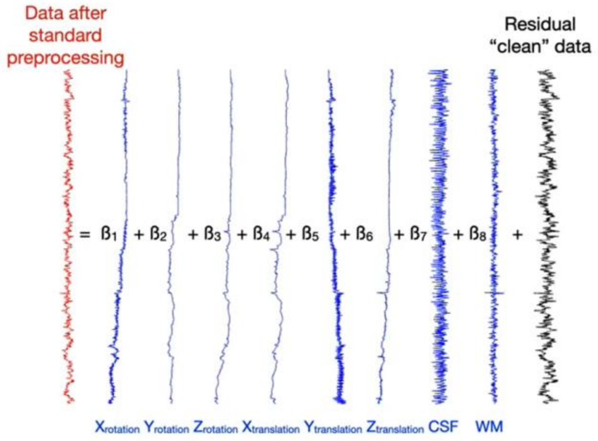
- **Physiological Signals**: Fluctuations related to cardiac and respiratory cycles can also introduce noise into the fMRI signal. Signals representing these physiological processes, often derived from external measurements, can be included as nuisance regressors.

- **Signals from Non-Neural Tissues**: Averages of the fMRI signal from white matter and cerebrospinal fluid (CSF) are often used as nuisance regressors, under the assumption that these areas are less likely to be involved in task-related neural activity and more likely to contain noise signals

Nuisance regression is used by leveraging the Generalized Linear Models (GLMs), a statistical method widely used in fMRI data analysis. The GLM is applied to each voxel's time series data to model the nuisance signals and separate them from the neural signals of interest. In this context, the model includes:

- **Data Time Series**: The observed fMRI signal for a given voxel, represented in the model by the time series data (in red in the below plot).

- **Nuisance Regressors**: Time series data for each identified source of nuisance variance (e.g., head motion parameters, physiological signals), included in the model as regressors (in blue in the figure).



- Also, the black line in the right side is the residual data after the contribution of the nuisance variables has been accounted for. Ideally, this should represent 'cleaner' data from which one can derive more accurate interpretations of brain activity, with the influences of head motion and non-neuronal physiological changes minimized.
- The GLM estimates coefficients (beta values) for each nuisance regressor that best fits the observed data, effectively modeling how much of the observed signal can be explained by these nuisance sources. The residuals of this model, representing the portion of the data not explained by the nuisance regressors, are then considered the cleaned data, purged of the identified nuisance variance.

It could be said that Nuisance regression is a critical step in cleaning resting-state fMRI data, ensuring that subsequent analyses, such as functional connectivity studies, are less likely to be confounded by non-neural sources of variance. By carefully removing these sources of noise, researchers can more confidently attribute observed connectivity patterns to genuine neural processes, enhancing the validity of resting-state fMRI studies.

8. Volume Censoring

Volume centering, also referred to as scrubbing, spike regression, or despiking, is a method used in the preprocessing of fMRI data, particularly effective for addressing the issue of head motion. This approach is based on the principle of identifying and excluding data volumes (or time points) that exhibit excessive motion, beyond a predefined threshold, to mitigate the impact of large motion artifacts on the analysis.

- **Motion Estimation**: The first step involves estimating the head motion for each volume (or time point) within the fMRI data set. This is typically done using motion parameters obtained from the motion correction step, which provides six parameters per volume representing three translations and three rotations of the head.
- **Framewise Displacement**: A common metric used in this context is framewise displacement, which provides a scalar value representing the amount of motion between consecutive volumes. It combines the translational and rotational parameters into a single measure, allowing for a straightforward assessment of motion at each time point.

- **Thresholding**: Once the motion for each volume is quantified, a threshold is set to determine what constitutes "excessive" motion. Volumes with framewise displacement values exceeding this threshold are flagged for exclusion. The choice of threshold is somewhat arbitrary and may vary between studies, but it is typically set based on empirical observations or the specific requirements of the analysis.
- Exclusion of High-Motion Volumes: The flagged volumes are then either removed from the data set or otherwise accounted for in a way that minimizes their impact on the analysis. This might involve excluding these time points from the analysis altogether or using interpolation to estimate the signal based on neighboring volumes.
- **Effectiveness**: Volume centering is particularly effective in reducing the impact of large head motions, which can introduce spurious correlations in functional connectivity analyses. By removing or mitigating these artifacts, the method helps in preserving the integrity of the neural signal of interest.
- **Side Effects**: The exclusion of volumes leads to variability in the number of time points available across different subjects or sessions. This variability poses challenges for group-level analyses and necessitates careful consideration in the study design and data interpretation.

Volume centering is a critical tool in the presence of significant head motion. Its application can significantly enhance the quality of the data, although researchers must be mindful of its limitations and the potential impact on subsequent analyses. (the amount of data that is removed in volume censoring is often relatively high (between 20% and 60% of all volumes.) Also, there is a large reduction in the temporal degrees of freedom of the data. The implication of this reduction in degrees of freedom is that the statistical power is reduced. This means that the estimate of functional connectivity becomes more noisy when it is calculated using fewer volumes

9. ICA-Based Clean-up

ICA-based cleanup in resting-state fMRI data preprocessing is a technique aimed at identifying and removing components of the data that represent noise rather than neural activity. ICA, or Independent Component Analysis, is a computational method used to separate a multivariate signal into additive, independent non-Gaussian signals or components. In the context of fMRI data, these components can represent various sources of signal, including both neural activity and various forms of noise such as head motion, physiological fluctuations, and scanner artifacts.

- **Component Decomposition**: The first step in ICA-based cleanup is to decompose the fMRI data for each subject into a set of spatial maps and associated time courses using ICA. This decomposition essentially attempts to break down the complex, intertwined signals in the fMRI data into simpler, independent components that can be more easily categorized as signal or noise.
- **Component Classification**: Once the ICA decomposition is complete, the resulting components need to be classified as either representing neural activity (signal) or various sources of noise. This classification can be done through several approaches:
- **Manual Classification**: In this step, we visually inspect the spatial and temporal features of each component, along with its frequency spectrum, to determine whether it more likely represents noise or neural signal. This process can be guided by known characteristics of noise components (e.g., spatially focal artifacts, high-frequency content) versus neural components (e.g., distributed networks, low-frequency fluctuations).
- **Automated or Semi-Automated Methods**: Techniques like FIX (FMRIB's ICA-based X-noiseifier) or ICA-AROMA (ICA-based Automatic Removal Of Motion Artifacts) use predefined criteria and machine learning algorithms to classify components automatically, reducing the need for manual intervention and subjective judgment.
- **Removal of Noise Components**: After classification, components identified as noise are removed from the data. This is typically done using a regression approach, where the time courses of the noise components are treated as nuisance regressors in a GLM model, and their influence is regressed out from the original data. The residuals from this model represent the cleaned data, with the effects of the identified noise components removed.

After these steps, ICA-based cleanup can effectively identify and remove a wide range of noise sources, including those that might be overlooked or inadequately addressed by other methods. By isolating and removing only the components classified as noise, ICA-based cleanup preserves the integrity of the neural signal in the data, which is crucial for accurate functional connectivity analysis. However, it should be noted that the manual classification of components requires expertise and can introduce subjectivity into the preprocessing pipeline. Automated methods help with this but might not be perfect for all datasets. In addition, ICA decomposition can be computationally intensive, especially for high-resolution or long-duration fMRI datasets.

10. Physiological Noise Regression

Physiological Noise Regression (PNR) in fMRI is a preprocessing step that aims to isolate and remove variations in the fMRI signal that arise from physiological processes, primarily cardiac (heartbeats) and respiratory (breathing) cycles. These physiological processes introduce fluctuations in the fMRI data that can be mistaken for neuronal activity, thereby confounding the interpretation of the results. The goal of PNR is to enhance the accuracy of fMRI analyses by ensuring that the observed signal changes are more representative of brain activity rather than artifacts of bodily functions.

The process begins with the concurrent collection of physiological data alongside the fMRI scans. Specialized equipment such as a pulse oximeter for monitoring the heart rate and a respiratory belt for tracking the breathing patterns is used. This additional data allows researchers to track the timing and intensity of each heartbeat and breath during the scanning session.

To deal with the effects of these physiological factors, the recorded cardiac and respiratory data are converted into regressors. Regressors are time series that model the expected influence of heartbeats and breathing on the fMRI signal. For cardiac effects, the timing of each heartbeat is used to generate regressors that reflect the pulsatile changes in blood flow and oxygenation with each cardiac cycle. Similarly, for respiratory effects, the cycle of inhalation and exhalation is modeled to account for the changes in thoracic pressure, blood CO2 levels, and subsequently, cerebral blood flow that affect the fMRI signal.

The complexity of the physiological influences necessitates sophisticated modeling to accurately capture their effects. One common approach is to use a Fourier series to decompose the physiological signals into a sum of sine and cosine waves at various frequencies, corresponding to the heart and respiratory rates. This helps in modeling the periodic nature of these artifacts more comprehensively.

Tools like RETROICOR (RETROspective Image CORrection) employ these Fourier-transformed physiological regressors to predict and correct for the cyclic variations in the fMRI signal attributed to cardiac and respiratory cycles. The method assumes that these physiological noises can be represented as a combination of several harmonics of the fundamental frequencies of heart and respiratory rates.

The actual correction is implemented through the General Linear Model (GLM), where the fMRI data is regressed against the physiological regressors. The GLM fitting process estimates the contribution of each physiological regressor to the observed fMRI signal at each voxel, and these contributions are then subtracted from the original data. The residuals from this regression, which represent the cleaned fMRI signal, are presumed to be purged of the physiological noise and thus more reflective of true neuronal activity.

Despite its efficacy, PNR is not without challenges. It should be noted that the variability in physiological processes across individuals means that the regression models need to be tailored to each dataset. Furthermore, there's a risk of over-correcting and inadvertently removing neural signals that might overlap in frequency with physiological noises. Therefore, careful implementation and validation of PNR are paramount to ensure the reliability of fMRI-based inferences on brain function.

11. Global Signal Regression

Global Signal Regression (GSR) is a technique used in the preprocessing of fMRI data to mitigate global sources of noise and artifacts. The method involves the computation of the global signal, which is the average time series of the fMRI signal across the whole brain, or a large, representative portion of it. This global signal is presumed to contain components of systemic physiological noise, such as fluctuations related to respiration and cardiac cycles, as well as other global artifacts.

Once computed, the global signal is then regressed out from the time series of each voxel in the brain. This process involves using the global signal as a regressor in a linear model and removing its influence from the data, ideally leaving behind a cleaner signal that is more representative of localized neuronal activity.

- **Computation of Global Signal**: The global signal is calculated as the mean time series across all voxels within the brain or a predefined brain mask. This calculation requires careful consideration to ensure that the mask or the selected voxels represent a broad and unbiased sampling of brain areas.

- **Regression Process**: The global signal is included as a nuisance regressor in a General Linear Model (GLM) applied to the time series of each voxel. The model estimates the contribution of the global signal to each voxel's time series, and this contribution is subtracted out, yielding residuals that serve as the cleaned data.

- **Residual Analysis**: After GSR, the resultant voxel time series (the residuals from the GLM) are used in subsequent analyses. These residuals are presumed to be less contaminated by global physiological and other artifacts, potentially enhancing the specificity of functional connectivity analyses.

- **Alteration of Functional Connectivity Patterns**: One of the main criticisms of GSR is its potential to alter intrinsic functional connectivity patterns. By removing the global signal, GSR can introduce negative correlations into the data and shift the overall distribution of correlation values, which could lead to misinterpretations of functional connectivity networks.

- **Physiological Noise vs. Neural Signals**: The global signal is thought to contain a mix of physiological noise and neural signals that are globally synchronized. Removing this signal might also remove neural information of interest, particularly signals related to large-scale brain networks like the Default Mode Network.

Note that the impact of GSR may vary across individuals and experimental conditions, making its effects difficult to predict and standardize across studies.

Suggested Available Automatic Tools: Brain Extraction

https://www.sciencedirect.com/science/article/pii/S1053811904001557

Suggested Available Automatic Tools: Segmentation

https://journals.sagepub.com/doi/10.1177/0972753121990175

*For now, these are the best resources I have. I might put some time and write down a few pages of analysis about this whole segmentation step later (after I finish the fMRI steps)

References

I used the FSL documentation for a great part of this file. This YouTube Channel also has great resources. The source for most visualizations I used, and the general outline, come from the FSL course. For the fMRI part, you could also check Andy’s Brain Book. Another very good resource that was helpful for the rsfMRI section was the book Introduction to Resting State fMRI Functional Connectivity. For the genetics part, I suggest taking a look at the great genetics boot camp tutorial from Gábor Mészáros.