



Pre-processing and Analysis of Functional MRI Data

M.Vink, M. Raemaekers, A. van der Schaaf, R. Mandl, N. Ramsey

Rudolf Magnus Institute of Neuroscience
University Medical Center Utrecht
Department of Psychiatry
Section Functional Neuroimaging

1 Overview of fMRI pre-processing and analysis

[figure 1]

2 Pre-processing

2.1 Introduction

Raw data obtained from an MRI scanner requires several manipulations to allow statistical analysis on these data. These manipulations include the following steps:

- STEP 1 → *Fourier transform* of the raw data obtained from the scanner
- STEP 2 → *realignment* of all functional images to the same orientation and position
- STEP 3 → *coregistration* of the T1 weighed image (anatomy) with the functional images
- STEP 4 → *normalization* of all images to a standardized size, orientation, and position (MNI space)
- STEP 5 → *smoothing* of the functional images by convolution with a Gaussian kernel

(Normalization is only applied when a group analysis is performed. Smoothing can also be applied to the statistical images rather than the functional images).

2.2 Fourier transformation – under construction

See Arjen van der Schaaf

2.3 Realignment – under construction

The goal of realignment is to align all functional images to one specific image, so that all functional images are in the same orientation and position. Functional images from an fMRI time-series are not in the same orientation and position due to movement of the subject during the fMRI experiment. Therefore, the source of the signal in one voxel can differ between scans (over time), resulting in a decrease in signal-to-noise ratio (SNR), by an increase in noise (see figure 2). A successful realignment ensures that the source of the signal in one voxel originates from the same location within each scan. However, some movement artefacts still remain after realignment (see Movement related noise 4.2.4 for a more detailed discussion).

[figure 2]

Realignment is carried out in two steps. First, the parameters needed for the (linear) rigid-body transformation of the images to a user-selected fixed image are determined. This image can be any one of the functional images. (The transformation is a so-called rigid body transform, meaning that 'the size of the brain' is kept constant.) There are three rotations (over x, y, and z -axis), and three translations (left-right, up-down, and forward-backward), making a total of 6 parameters which fully describe the movement of the subject over time.

Second, the parameters are applied to the functional images. To obtain the new voxel values, resampling of the data is required.

2.3 EXPLORATION

The rigid-body parameters are parameterised by:

$$\begin{matrix} \text{Translation} & & \text{Pitch} & & \text{Yaw} & & \text{Roll} \\ \begin{pmatrix} 1 & 0 & 0 & X_{translation} \\ 0 & 1 & 0 & Y_{translation} \\ 0 & 0 & 1 & Z_{translation} \\ 0 & 0 & 0 & 1 \end{pmatrix} & \times & \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\Phi) & \sin(\Phi) & 0 \\ 0 & -\sin(\Phi) & \cos(\Phi) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & \times & \begin{pmatrix} \cos(\Theta) & 0 & \sin(\Theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\Theta) & 0 & \cos(\Theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} & \times & \begin{pmatrix} \cos(\Omega) & \sin(\Omega) & 0 & 0 \\ -\sin(\Omega) & \cos(\Omega) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{matrix}$$

[figure 3]

2.4 Coregistration – under construction

The goal of coregistration is to obtain an overlap between functional images and the anatomical image, so that the activation areas are located at their correlation anatomical positions. There are two methods one can use: (a) segmentation of the images and subsequent matching of the separate segments, and (b) mutual information, which should be used if the images have a different modality.

[figure 4]

The mutual information method maximizes the mutual information in the 2D histogram in a limited number of iterations. In other words, the difference between the two images is minimized. Because there are a limited number of iterations, it is important in SPM99 that prior to coregistration, the images are in approximately the same location.

2.5 Normalization – under construction

The brain of every individual is different. For an individual analysis, in which one is only interested in the regions that are active due to a task in a particular subject, this is no problem. However, when performing a group analysis it is essential that all brains are of the same size and orientation. During normalization, the images are warped so that functionally homologous regions from different subjects are as close together as possible. However, there is no exact match between function and structure (cf. the use of Brodmann maps), and in addition the structure itself differs between subjects (for example, the cingulate cortex in some subjects consists of two but in other subjects of three gyri). To somehow correct for these factors, the image is blurred (see Smoothing). Here, we will discuss how normalization is performed.

First, a template brain is selected. This can be either a template as shown in figure 5 or any linear combination of templates. A template is typically in MNI space (Montreal Neurological Institute). This template is used worldwide, so once your images are in MNI space, you can compare your results per coordinate with results from all other institutes. The second step in normalization involves the minimalization of the sums of squared differences between the

template brain and the original brain, and also the squared number of standard deviations away from the expected parameter values. In contrast to realignment, which is a rigid-body transformation, normalization involves also changing the size of the brain using a linear 12 parameter affine registration to match size and position of the template (see figure X). By masking the original image non-brain voxels are deleted and hence cannot affect this affine registration. Such a brain mask can also be applied when normalizing lesioned brains. The third step involves a global non-linear warping of the original brain to match the template. This non-linear warping is based on a Bayesian framework to simultaneously maximize the smoothness of the warps.

[figure 5] & [figure 6]

2.6 Smoothing

Smoothing involves blurring *functional* MRI images using (typically) a Gaussian filter (i.e. data is convolved with a Gaussian kernel). Smoothing, in practice, is mainly applied when a group analysis is performed. By smoothing the image, the overlap of activation between subjects is increased.

When one smooths an image, each voxel effectively becomes the result of applying a weighted region of interest (ROI; the voxels under the kernel). The size of this kernel (see figure 7) is determined by the full width at half maximum (FWHM). The FWHM is an indication of the distribution of the kernel values, meaning that when the FWHM is 8 mm, the kernel is 8 mm wide at 50 percent of its peak value. The voxel falling within the range defined by the FWHM receive the highest weights, while the voxel falling outside this range receive lower weights. Typically, a FWHM kernelseize is chosen so that it matches the size of the expected activation. For example, if the activation is an area of 10 mm³, then the FWHM should be 10. If a FWHM of 20 is chosen, then the activation in that activation area is averaged with the activation in the surrounding voxels. If these are not significantly active, then the overall activation in that region (which is now 20mm³) is reduced compared to the ROI of 10mm³.

[figure 7] & [figure 8]

How is the data convolved with the Gaussian kernel? Figure 8 depicts a 3D representation of the kernel. This can be rewritten in a 2D form (figure 9). Now, let's calculate the new value of the middle voxel. On the left, the 2D Gaussian kernel is depicted. This kernel is applied to the data (e.g. from a t-map or functional scan). To calculate the new value for the voxel in the middle, multiply each voxel value with the corresponding value of the Gaussian kernel and divide by 273 (which is the total numeric value below the curve). In this way, the value of the middle voxel in the smoothed image effectively depends for $\frac{41}{273}$ part upon it's own old value, and for $\frac{273-41}{273}$ part upon the voxel values of it's neighbouring voxels. After the new value of the middle voxel is calculated, the kernel is moved one voxel to the left (or up), and the procedure is subsequently repeated for all voxels in the three dimensions.

[figure 9]

After smoothing the image, the number of voxels and the size of the voxels (i.e. the sampling rate) remains the same. However, the resolution of the image becomes less (see figure 7). Whereas in the not-smoothed image, the resolution was the same as the voxel size (for example 4 mm), the resolution of the blurred image is specified in terms of *resels* (i.e. *resolution elements*). A resel consists of a number of voxels that fall within the FWHM. For

example, if the FWHM is 8 mm (3D) and the voxels are $4 \times 4 \times 4$ mm, then a resel consists of $2 \times 2 \times 2 = 8$ voxels. The number of resels is then the number of voxel divided by 8.

2.6 EXPLORATION

Calculation of the smoothed value

Multiply the value of each voxel with the corresponding kernel value and divide this by the total value of the kernel (= 273). The sum of these values is the smoothed value of the middle voxel:

$$((1 \times 5) / 273) + ((4 \times 7) / 273) + ((7 \times 4) / 273) + \dots + ((1 \times 8) / 273) = 4.8242$$

There are three reasons for smoothing: (1) *Increase the signal-to-noise ratio (SNR)*. Because the signal in a voxel in a smoothed images originates not only from the voxel itself but also from it's neighbouring voxels, the effect of random (uncorrelated) noise is reduced. (2) *Increase intersubject overlap*. After normalization, the brains are in MNI space, but there may be slight differences between subjects in the relationship between function and structure. By averaging the signal over a larger area, the overlap between activation spots between subjects (possibly) increases. (3) *Increase validity of the analysis*. Neighbouring voxel values in an image are spatially correlated. Recall that the goal of pre-processing is to allow statistical analysis on the data. The analysis we will discuss in chapter 4 is voxel-based, which means that per voxel a test of significance is performed. To assess which voxels are significantly active during for example a task, a threshold needs to be set. In order to determine the height of that threshold, the number of independent tests that are performed is needed. This number is not the same as the number of voxels, since these voxels are correlated (the results of these voxel-based tests are also correlated). An estimation of the number of independent tests that are performed is the number of *resels* in an image. However, these resels are also not completely independent, but are a better estimation of the true number of independent observations than the number of voxels (see also Thresholding).

Convoluting the image with a Gaussian kernel poses a problem on the edges of the image (see figure 10, Maisog & Chmielowska, 1998). The value of the smoothed voxel in the middle still depends for $\frac{41}{273}$ part on its old value and $\frac{232}{273}$ part on it's neighbouring voxels. However, around the edges some of these neighbouring voxels are outside the brain and have value 0. A similar problem occurs on borders between white and grey matter. The values of the voxels outside the brain as well as of the voxels in white matter are not correlated with the value of the voxel in the middle, but these values are included in the calculation of the smoothed value for this middle voxel. This results in an underestimation of the smoothed voxel value. To correct for this border-effect, one can either (a) mirror the region, so that the values of the voxels outside the brain (or in white matter) are replaced by the values of the voxels in the grey matter, (b) apply some form of edge truncation so that the voxels which fall outside the brain (defined by a brain mask) do not affect the weighing of the smoothed voxel values, or (c) not smooth these voxels.

[figure 10]

3 Experimental design

3.1 Introduction

In this chapter, we will discuss two main design types used in fMRI research; blocked and event-related (ER) designs. A blocked design refers to a design in which the task is presented in so-called blocks of for example 30 seconds, alternated with periods of rest (rest blocks). The signal that is measured effectively arises from a combined brain response to a number of trials (events). In contrast, the idea behind an event-related design is to obtain the brain response to a single event. In discussing these designs, we will focus on power and efficiency. The power of a design refers to how well two conditions can be separated in terms of brain activation. Efficiency refers to the amount of data that can be obtained within a fixed time period. First, let us look at the characteristics of the signal (BOLD dependent) fMRI is based on and the function which is used to describe this signal.

3.2 BOLD and HRF characteristics

The signal that is commonly measured using functional MRI is called the BOLD (Blood Oxygenation-Level Dependent) signal (see figure 11). As suggested by the name, this signal depends upon the oxygen-level in the blood rather than on direct neuronal activation. There is, however, a relationship between neural activation and the BOLD signal. Neuronal firing will commence immediately in response to stimulus presentation. This process requires oxygen, which is supplied through the blood. Because immediately after stimulus onset no extra oxygen is supplied, a slight decrease in the signal can be observed. This initial dip quickly transforms into a positive signal when the oxygen supply starts to build up. After approximately six seconds post-stimulus, the signal that is detected with (BOLD dependent) fMRI reaches its maximum (peak). After 20 seconds, the signal from the original stimulus has returned to zero. This BOLD signal is modelled by a haemodynamic response function (hrf), which is a generalized approximation of the actual BOLD curve (figure 14, green line). However, the specific characteristics of the BOLD curve can differ between (a) *brain regions within one subject*, and (b) *subjects*. So, for example, in some regions (or subjects) the actual BOLD response may peak after 5 instead of 6 seconds, or the BOLD response is much wider in some brain regions (subjects). Due to these possible variations, the signal we attempt to describe using a fixed hrf may not be present at all in the data. Then the conclusion may be that there is no significant brain response to a specific stimulus, while in fact there is a response but it differs from the function we use to define a brain response. To prevent such 'false conclusions', a correction is needed to account for inter-region and inter-subject variability in BOLD responses. SPM99 allows a correction for both onset time and dispersion of the BOLD curve, by including a time and a dispersion derivative in the hrf (see figure 14, blue and red line). Input functions other than those strictly based on the BOLD curve can be used in SPM99, such as (a) a set of Fourier functions, (b) Gamma functions, and (c) a Finite Impulse function. However, the problem with using input functions which are not based on (at least an estimation of) the true physiological signal lies in the interpretation of the results. Namely, when activation in a specific region is highly correlated with a specific Gamma function, it is difficult to understand the source of this relation in terms of the known BOLD response. Using an input function that is based on the true BOLD signal will yield areas in which the signal behaves in a way described by this function. Since this input function is physiological valid, brain activation can be explained in terms of the mechanisms underlying the BOLD signal.

[figure 11] & [figure 14]

3.3 Blocked designs

A blocked design potentially has a lot of power, since the brain is either repeatedly doing the task, or doing 'nothing' during rest blocks. The signal associated with performing the task is maximized during task blocks and low during resting periods, thereby maximizing the difference in brain activation during these two conditions. A blocked design is also efficient in that a lot of brain activation is recorded within a fixed time period, because the separate task trials are spaced closely together.

One should consider (at least) two points when designing a blocked paradigm:

(a) Ideally, the number of scans and events should be equal in all conditions, so that the variance in all factors is the same. This means also that the power for all conditions is equal. Consider for example, a task with two conditions and rest periods. Every condition should then consist of 1/3 of the scans.

(b) The length of a block should be between 14 to 20 seconds, yielding a task frequency of 0.036 to 0.025 Hz (Zarahn et al. 1997). However, when there is no accurate description of the BOLD response, then the block should be longer (around 30 seconds; 0.016 Hz, Aguirre & D'Esposito, 1999). Using slightly longer blocks places less importance on how well the rise and subsequent fall at the beginning and the end of the block is described (figure 13 A and B, and see text below). When this description is very accurate, for instance because the actual BOLD response is known, then the blocks can be short. The longer the blocks are, the more chance there is for a correlation with low-frequency scanner-related artefacts (see also 4.2.2 LOW-FREQUENCY NOISE).

A limitation when using a blocked design is that randomization of trials from different conditions is not possible. The idea of a blocked design is to maximize the signal related to a specific task by presenting a number of trials from the same condition in close succession. So although there may be different blocks consisting of different conditions, each block typically consists of only one condition. This poses a problem for some cognitive tasks, for which it is not possible to divide the different conditions into separate blocks (e.g. a Go-NoGo task, in which blocks consisting of only NoGo trials are not possible). Furthermore, a drawback of using a blocked design is that the strength of the brain signal can decrease over time (neurons which fired strongly at the beginning of the block may decrease their firing rate over time due to neuronal mechanisms like habituation). The idea of a blocked design is that presenting multiple trials in close succession maximizes the signal. This BOLD response is modelled by a so-called box-car function (see text below). When activation drops (i.e. the BOLD response decreases), then this box-car function is not a good estimation of the neuronal response anymore. In short, the model that is used to describe the BOLD response is inadequate, leading to bad statistical values (see chapter 4 STATISTICS).

A commonly used function to describe the signal variance in a blocked design is a *box-car* (see figure 13, A), which encodes a '0' for no-task (i.e. rest or another task) and a '1' for task. However, brain regions do not suddenly activate and stop activating in the way modelled by the box-car function, but rather the BOLD response peaks after (approximately) 6 seconds of stimulus onset, and has an undershoot at the end of the block. By convolving the box-car input function with a function which best describes how the BOLD signal rises (e.g. a hrf or

haemodynamic response function), a better estimation of the true block-related BOLD signal is obtained (figure 13, B).

[figure 13]

3.4 Event-related designs

When two stimuli are presented in close succession (for example 1 second apart), the corresponding BOLD curves will overlap, presumably in a linear fashion (meaning that the response is an addition of both curves). If the form of the BOLD curve is known, then these two overlapping curves can be disentangled (alternatively, one can use a hrf to estimate the two BOLD curves). However, the way BOLD curves add up depends upon (a) the distance in time between two stimuli, and (b) the number of consecutive stimuli. When a large number of stimuli (e.g. 20) are presented in close succession (i.e. 1 second apart), the individual BOLD curves cannot be disentangled anymore. The response will reach a plateau with little variance, so that individual BOLD curves originating from individual events cannot be calculated anymore. In an event-related design, in which different trials are randomly intermixed, one would like to determine these event-related BOLD curves. What would be an optimal interstimulus interval (ISI) for an event-related design? On the one hand, one would like to present as many events as possible in a short time period (i.e. high task frequency). However, then the power of each task factor (representing the various task conditions) is low. To increase the power, stimuli should be spaced more widely over time (i.e. low task frequency). However, at lower frequencies, noise occurs (such as scanner-drifts, see 4.2.2 and 4.2.3), thereby disturbing the task related signal. Combining these factors (1 high task frequency → low power, 2 low frequency → scanner noise) suggests an optimum frequency for an event-related design with a fixed interstimulus interval, namely around 0.05 - 0.08 Hz (a/o. Bandettini & Cox, 2000). This effectively comes down to presenting an event every 12 to 20 seconds.

[figure 12]

However, there are other event-related designs that allow a shorter interstimulus interval. Basically there are four types of event-related designs:

1. fixed long interstimulus interval (ISI) (a/o. Bandettini & Cox, 2000)
2. variable ISI (a/o. Dale, 1999)
3. variable ISI with occasional small blocks of the same condition (van der Schaaf, 2002)
4. fixed short ISI (task 1) randomly interspersed with other trials (task 2) (Buckner, ???)

3.4.1 Fixed long ISI

3.4.2 Variable ISI

3.4.3 Blocked variable ISI

3.4.4 Fixed short ISI

4 STATISTICS

4.1 Introduction

The goal of performing an fMRI experiment is to determine which brain activation is significantly associated with a cognitive/physiological process/mechanism. Essential for localising the activation are the realignment and co-registration steps of pre-processing. To determine which activation is significant, some statistical analysis is required. We will begin this section by describing the strategy for such an analysis.

When looking at the signal in a particular voxel over time, one immediately sees that this is not a straight line, but rather goes up and down in a seemingly random fashion. In other words, the signal values vary over time. There are several factors, or rather *effects*, that are responsible for this signal variance. The goal of the statistical analysis is to describe (i.e. to model) these effects and subsequently assign weights to them.

For example, the subject has to perform a task, which is alternated with periods of rest. It is to be expected that in voxels which are located in brain regions involved in performing the task (e.g. motor cortex activates when pressing a button), the signal will increase during the task, while during rest the signal will return to baseline state. In this case, (part of) the variance in the fMRI signal is caused by, and can therefore be described by, a factor which specifies when the task is performed and when there is a rest period. The variance in the fMRI signal is also affected by various other effects. We can make a distinction between effects of interest (variance due to performing a task), effects of no interest (variance in the signal due to 'known' effects like scanner artefacts), and random noise.

$$\begin{aligned}
 [3.1] \quad \text{fMRI signal} &= \text{effects of no interest} &+& \text{effects of interest} &+& \text{random noise} \\
 &(\text{variance due to} && (\text{variance due to task}) \\
 &\text{- global effects} \\
 &\text{- low frequency noise} \\
 &\text{- high frequency noise} \\
 &\text{- movement related noise)
 \end{aligned}$$

The idea is to best describe the fMRI signal in terms of effects of interest and effects of no interest. The better you can describe the variance in the signal (i.e. explain the variance), the less random noise you have.

4.2 Effects of no interest

Effects of no interest can be generated by for example scanner drifts (i.e. drifts in the signal due to scanner instabilities), breathing, and heartbeat. The goal of including factors that model effects of no interest is to remove the variance these effects generate in the signal. It is very important to include factors that correct for effects of no interest, because if one omits these, the error variance will increase. This, in turn, will lead to lower significance of the effects of interest (fit of the model decreases). Basically, the effects of no interest are a source of error, but in contrast to *random* error, these effects can be described by some

factor(s). By removing these sources of error by correcting for them, only random error will remain.

4.2.1 Global effects

The global scaling factor corrects the differences in image intensity over time within an fMRI time-series. These differences in image intensity are mainly caused by scanner drifts (EXPLAIN). The global scaling correction is done by dividing intensity values for each scan by the mean value for all voxels for each scan. It is implicitly assumed that global intensity (i.e. global volume mean) is not correlated with the task. However, if this is the case, than task activation is also scaled down (see figure 15).

[figure 15]

In a typical fMRI volume, there are 20,000 voxels (which describe the brain). For activation to have a great effect on global volume mean a very large number of voxels needs to be very active during the task. In most cases, it is not likely that activation and global volume mean are highly correlated. Rather, other factors such as scanner drifts are more likely to be responsible for changes in global volume mean. Therefore, including a global scaling factor will generally be beneficial.

4.2.2 Low frequency noise

In the data (fMRI time series), some low frequency signals may be present, which are not due to the task. These signals may come from baseline drifts from the scanner. Removing these signals will decrease the error variance. To remove these signals, SPM99 offers the possibility to apply a high-pass filter (i.e. signals with a high frequency may pass) over the data. This high-pass filter consists of some low frequency cosine functions, ranging from a halve cosine which describes a trend in the fMRI time-series up to the cosine function with the highest frequency.

$$[3.3] \quad f_r(t) = \cos\left(r\pi \frac{t}{t_N - t_1}\right)$$

The function $f_r(t)$ is applied to an fMRI time series of N scans, acquired at times t_1, \dots, t_N , whereby r ranges from 1 (which will give a halve cosine function (i.e. basically a linear trend factor)) to a specified number reflecting the highest frequency (see figure 16). It is important that the cosine function with the highest frequency is not correlated with the task factor (see 5.2 MULTICOLLINEARITY). To ensure that the cosine function with the highest frequency does not correlate with the task factor, a simple formula can be used to calculate the highest frequency allowed for the high-pass filter, using a cut-off period for this cosine function (i.e. effectively determine the frequency). The cut-off period is given by:

$$[3.4] \quad \text{cut-off period} = (\text{rest} + \text{task}) \times 2$$

This number should be multiplied by the TR (time for one scan) for SPM99. The maximum value for r (and hence the highest frequency) is calculated by:

$$[3.4a] \quad \text{maximum } r = \frac{N}{(\text{rest} + \text{task})}$$

where N is the total number of scans and (rest + task) refers to one experimental cycle consisting of one rest period and one task period.

[figure 16]

4.2.3 High frequency noise

Because fMRI data is in fact data from a time-series, the value in a particular (brain) voxel at a given time t_n is correlated with the value in that same voxel at time t_{n-1} or t_{n+1} (where 'n' is in scans). This fact is important for the way the analysis is performed. Originally, you treat the data as if you have an observation of the brain signal each time you make a scan (i.e. independent observations). However, because temporally neighbouring scans are correlated with each other over time, these observations are not independent. In fact, the number of true independent observations is less than the number of scans. Therefore, when evaluating individual data a correction is needed. There are two possibilities: (a) Apply a low-pass filter on the data (e.g. a Gaussian curve). This way, the correlation between the neighbouring scans is forced in a specific pattern (in the same way as spatial smoothing, see 2.6 SMOOTHING). A disadvantage of this method is that the temporal specificity decreases. (b) The other option is to calculate the true temporal correlation between neighbouring scans, using an AR(1) model (i.e. first-order auto-regression model):

$$[3.5] \quad y_t = \alpha_1 y_{t-1} + e_t$$

with y_t is the residual error from scan obtained at time t . Basically, this model calculates for which α the residual error of scan $_{t-1}$ best describes the residual error in scan $_t$. This is calculated for each voxel separately. In the analysis, the number of observations will be adjusted for the factor α .

When you are only interested in group results, including a correction for temporal auto-correlation is not needed. In fact, including such a correction does not affect your group results (This is because the correction involves a proportional decrease in the number of degrees of freedom, which will hence lead to proportional lower t-value, but also to a proportional lower standard deviation (see chapter 6 GROUP ANALYSIS)).

4.2.4 Movement related noise

Subjects do move while performing an fMRI experiment. This movement is calculated during the realignment and the images are transformed accordingly (section 2.1.1). However, movement induces a disturbance in the signal for the entire scan. For example, the signal to noise ratio (SNR) changes in all voxels. Basically, signal variance is added due to movement. By including the movement parameters in the design matrix, this additional variance can be explained. A new problem presents itself here, namely if the movement is correlated with the task. In that case, either a task factor or the movement factor can explain the variance in the signal. If you do correct for movement (i.e. include the movement parameters as factors in your design matrix), this will negatively influence the amount of variance in the signal left for

the task factor to explain. However, the variance in the data you subsequently explain with the task factor is reliably related to that task factor, because movement related variance is deleted.

4.3 Effects of interest

The effects of interest refer to the variations in the signal which are due to the task. For example the signal may increase when the task is performed, and decrease during a rest period. At the beginning of this chapter, the term 'model fit' was introduced. This term refers to how well the model explains the variance in the signal. For fMRI, it is important that the model as a whole explains as much variance as possible, because the amount of unexplained variance (i.e. *residual error*) should be as small as possible. In addition, specifically in the case of fMRI, we also look at the variance explained by each factor separately, as to make an estimation of how strong this particular effect is present in the signal. If a particular factor describes task related activation, then we are very interested in how much variance is explained by that factor in particular, rather than what the entire model explains. Consider for example a factor describing a blocked design (see chapter 3), encoding a '0' for rest and a '1' for activation. The factor would then look like this:

#scan	1	2	3	4	5	6	7	8	9	...	30	31
event	rest	rest	rest	rest	rest	task	task	task	task	...	rest	rest
FACTOR	0	0	0	0	0	1	1	1	1	...	0	0

Figure 17b gives a graphical representation of the same factor:

[figure 17b]

So we expect the activation in some voxels to follow the pattern of the factor. If there is activation that behaves according to this pattern, then this would be interpreted as being activation (variance) due to this factor. We want to make an estimation of how strong this activation (variance) is present in the data. This is roughly measured in terms of the number of times the factor is present in the data. In figure 17b, the letter 'B' depicts this multiplication scalar. This letter B represents a matrix containing b-values (the multiplication scalars, see figure 17) for all factors separately. We expect there is a linear relation between the factor and the fMRI signal (hence the name of the analysis, General Linear Model, GLM). If the variance related to the factor is strongly present, then the b-value for that factor is high (and vice versa). Remember that this signal processing is done for each voxel separately. So one factor can result in high b-values in some voxels, but low b-values in others.

Taken together, the factors modelling the effects of no interest and effects of interest form the model that as a whole describes as much as possible of the variance in the fMRI signal. The remaining part of the variance in the fMRI signal is presumably random noise (randomness of the noise is one of the assumptions which have to be met before statistical analysis can continue). The next step after building the model is to 'fit the model to the data'. In other words, we are going to estimate how strong each factor is present in the fMRI signal.

4.4 Fitting the model: b-values

Assume we have the following situation:

[figure 17]

Y : represents the data, in this case the data from one voxel is displayed. The analysis is voxel-based, meaning that the entire analysis is performed for each separate (brain) voxel (voxels outside the brain are discarded using a mask).

X : represents the design matrix or model, which contains 9 factors:

1. factor for task 1
2. factor for task 2
3. linear trend (see effects of no interest)
- 4-8. movement related parameters (see effects of no interest)
9. *intercept*

The *intercept* actually is a vector (column in the matrix) consisting of elements with the value 1. This factor is included to model the basic activation level in the signal. During rest periods, the signal is not zero but rather at a different level compared to task periods. During task periods, the signal effectively is the sum of the basic activation level PLUS the increase (or decrease) due to the task.

B : represents the vector of multiplication scalars

E : represents the error (i.e. the variance in the signal that is not explained by the model)

In mathematical terms, this is stated by:

$$[3.6] \quad Y = XB + error$$

At this point, we want to find the multiplication scalars (i.e. b-values) for this particular model which best describe the data. This 'best description' is defined in terms of yielding the smallest residual error. The residual error refers to that part of the data that is not described in the model, and hence is given by:

$$[3.7] \quad error = Y - XB$$

[figure 19]

A more formal approach of obtaining b-values that yield the smallest residual error is called the 'least squares criterion'. In mathematical annotation, this becomes minimizing:

$$[3.8] \quad \sum (error^2) = (Y - XB)'(Y - XB) = e'e = SS_e = \text{sums of squares of the error}$$

The solution for the minimalization of the sums of squares of the error is given by:

$$[3.9] \quad B = (X'X)^{-1}X'Y$$

provided that $(X'X)^{-1}$ exists. The B matrix contains estimations of the true B values.

The residual error, meaning the part in the signal which is not explained by the model, is called the *residual mean square*, and is given by:

$$[3.11] \quad MS_e = \frac{e'e}{N-H} \quad (\text{in contrast, the error variance is given by: } \frac{e'e}{N-1})$$

where N is the number of scans (number of independent observations) and H is the number of factors of the model. Now we can make a b-map for every factor which represents the how strong this factor is present in the data, specified for each individual voxel.

4.4 EXPLORATION

Formula 3.9 is obtained using:

$$[3.10] \quad Y = X^\perp + XB$$

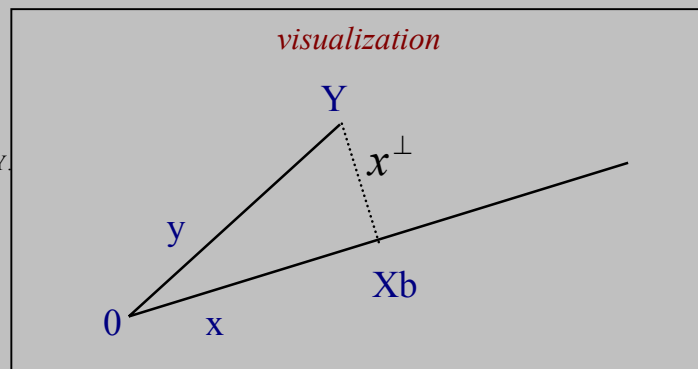
which states that the best estimation of Y is a vector X straight below Y plus a correction for the height of Y.

Multiplying [3.10] with X' gives:

$$[3.10] \quad X'Y = X'X^\perp + X'XB = 0 + X'XB = X'XB$$

Then multiply [3.10] with $(X'X)^{-1}$ gives:

$$([3.9]) \quad B = (X'X)^{-1}X'Y$$



4.4.1 An example

Assume that a specific subject has performed an fMRI experiment with two conditions. We will look at the activation time series of two voxels in a brain. Figure 18 depicts both the fMRI signal for two voxels as well as the model describing this signal (note that it is a limited model with only factors modelling effects of interest). Originally, the model encodes a '0' for rest and a '1' for activation. Similar to SPM, we first transform the model so that the factor means are 0.

[figure 18]

Now we can determine which linear combination of the model best describes the data for each voxel separately.

These values are found using formula 3.9. For **voxel 1**, this becomes:

$$([3.9]) \quad B = (X'X)^{-1}X'Y = \begin{pmatrix} 0.192 & 0.07 & -0.00 \\ 0.07 & 0.192 & -0.00 \\ -0.00 & -0.00 & 0.032 \end{pmatrix} \times \begin{pmatrix} 26.026 \\ 8.256 \\ 101.00 \end{pmatrix} = \begin{pmatrix} 5.54 \\ 3.32 \\ 3.26 \end{pmatrix}$$

For both voxels, the b-values become:

	Voxel 1	Voxel 2
	b-value	b-value
Task 1	5.5386	7.5386
Task 2	3.3173	9.8173
Intercept	3.2581	5.4516

Note that the b-values for both task factors are higher in voxel 2 than in voxel 1. As can be seen in figure 18, this is caused by the higher signal in voxel 2. Signal strength is represented by the variance of the signal (variance(voxel1) = 6.24, variance(voxel2) = 21.84).

4.5 Making a t-map for a single factor

In this section, we will describe the transformation from b-value to t-value in order to determine the significance of the b-value. So far, we have obtained a b-value for each voxel in the brain, resulting in a b-map. There is a b-map for each factor of the model. The next step is to determine per b-map in which voxels this factor is significantly present. This is not directly reflected by the size of the b-value, but rather in every voxel a so-called t-test is performed to assess if the factor adds to the variance which is explained by the entire model. After performing the t-test, a t-map can be made for each factor, reflecting in which voxels this factor adds to the explained variance of the entire model. In other words, whereas the b-value reflects the weight of the factor within the model, the t-value reflects the contribution of this factor, corrected for the other factors, compared to the amount of error of the entire model. Whether this contribution is significant depends upon whether the t-value exceeds a certain threshold (see 5.3 THRESHOLDING).

How can we obtain the t-value associated with the b-value? This transformation is actually the resultant of a test to determine if the b-value significantly differs from zero. In other words, a test is performed to determine whether the factor, which is multiplied by a certain b-value, adds something to the model or that this factor explains no variance and can better be deleted from the model. Factors that do not add variance to the model only ensure lower t-values for all factors of the model (because residual mean square increases). Basically, we must decide whether the increase in the amount of explained variance (so-called regression sums of squares) is sufficient to warrant using the additional factor in the model. The hypotheses for testing the significance of any individual b-value (regression coefficient) are:

$$H_0 : B_j = 0$$

$$H_1 : B_j \neq 0$$

If H_0 is not rejected, then this indicates that the factor j can be deleted from the model, because the factor does not uniquely explain some part of the variance. The test statistic for this hypothesis is:

$$[3.14] \quad t = \frac{c' B}{\sqrt{MS_e c'(X' X)^{-1} c}} = \frac{c' B}{se(B_j)} \quad \text{with } df = N - H - 1$$

This test is actually a test of the contribution of the factor j given the other factors in the model. This is easy to see because the standard error of the mean, $se(B_j)$, depends upon both the residual error of the model and the unique contribution of the factor given the other factors.

Another way of looking at the t-value is to say that the t-value reflects the reliability of the b-value. The more reliable the b-value is, the smaller the standard error of the mean becomes, and hence the higher the t-value.

Let's look at the different components of this formula:

c' refers to the contrast vector. This is a matrix used to select a specific factor. For example, if you want to work only with the first factor of a design matrix with 9 factors (from figure 17), the contrast matrix will become:

$$c' = (1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)$$

B is the matrix containing the b-values for each factor of the design matrix:

$$B = (b_1 \quad b_2 \quad b_3 \quad b_4 \quad b_5 \quad b_6 \quad b_7 \quad b_8 \quad b_9)$$

MS_e is the mean of the squared residuals (residual mean square) and is a weighted representation of the error variance $1 - R_{y,H}^2$ (see [3.11]). If a particular factor is included in the model, but explains no unique variance, then the residual sums of squares remains the same, but the residual mean square increases (see [3.11]), because of the increase in the number of factors H.

$c'(X'X)^{-1}c$ is a weighted representation of $\frac{1}{1 - R_{j,G}^2}$ and refers to that part of the variance uniquely explained by a particular factor j . The more unique variance this factor explains, the smaller this element becomes, and hence the higher the t-value becomes.

4.5 EXPLORATION

Model fit; the F-test

The residual sums of squares, the part of the data that is NOT explained, is:

$$SS_E = S_{yy} - SS_R$$

The regression sums of squares, the part of the data that IS explained, is:

$$SS_R = B'X'y - \frac{\left(\sum_{i=1}^n y_i\right)^2}{n}$$

and the total sums of squares, the total data that can be explained, is:

$$S_{yy} = y'y - \frac{\left(\sum_{i=1}^n y_i\right)^2}{n}$$

so that the residual sums of squares effectively becomes:

$$SS_E = y'y - B'X'y$$

The model fit can be tested using the F statistic:

$$F = \frac{SS_R / H}{SS_E / N - H} = \frac{MS_R}{MS_E} \text{ with } df = N, N - H$$

where H is the number of factors and N is the number of observations (scans)

4.5 EXPLORATION

Exploring the meaning of $(X'X)^{-1}$

$(X'X)^{-1}$ is the weighted inverse covariance matrix of X . A covariance matrix has diagonal elements which represent factor variance and non-diagonal elements which represent covariance. Factor variance is given by:

$$[3.15] \quad S_{factor}^2 = \frac{\sum (x - \bar{x})^2}{N - 1}$$

and covariance between factors by:

$$[3.16] \quad S_{12} = \bar{x}_1 \bar{x}_2 - \bar{x}_1 \times \bar{x}_2$$

In matrix annotation, the covariance matrix containing both variance and covariance of the factors of the model X is obtained by:

$$[3.17] \quad Cov(X) = X_{dev}' X_{dev}$$

where [3.18] $X_{dev} = X - \bar{X}$

where \bar{X} has columns containing factor averages. In SPM99, all factors are weighted so that each factor has mean 0. Therefore, in this particular case, we get:

$$([3.18]) \quad X_{dev} = X - \bar{X} = X - 0 = X$$

so that

$$([3.17]) \quad Cov(X) = X_{dev}' X_{dev} = X' X$$

Element (1,1) of $X'X$ represent the factor variance for factor 1, although not corrected for the number of observations (see formula 3.15). This element is also called the sums of squares for factor 1 (SS). The matrix $X'X$ is also referred to as sums of squares and cross products (SSCP), since the non-diagonal elements are cross products of pairs of factors.

The next step is to invert $X'X$:

$$[3.19] \quad (X'X)^{-1} = \frac{1}{\det} \begin{pmatrix} m(X'X)_{(1,1)} & \dots & m(X'X)_{(1,H)} \\ \vdots & \ddots & \vdots \\ m(X'X)_{(H,1)} & \dots & m(X'X)_{(H,H)} \end{pmatrix}$$

where H is the number of factors, and $m(X'X)_{(h,h)}$ refers to the *minor* of the h^{th} element $X'X$ which is defined by the determinant of the matrix formed by deleting the h^{th} row and h^{th} column from $X'X$. This element reflects the combined variance, corrected for the covariances, of the factors other than h . The next step is calculate the determinant of the entire matrix $X'X$. This determinant reflects the generalized variance of the entire model. Covariance between factors would decrease this variance, since in that case more than one factor explains a segment of the total variance.

The first element of the matrix $(X'X)^{-1}$ then becomes:

$$[3.20] \quad inv(X'X)_{(1,1)} = \frac{m(X'X)_{(1,1)}}{\det(X'X)}$$

and basically reflects the ratio between generalized variance of the other factors and the generalized variance of the entire model. In more practical terms, this element reflects the variance explained by the first factor (increasing the factor variance four times will result in four times decrease in value of this element).

4.5.1 An example - continued

In this example, we will calculate the reliability of the b-values by rewriting them as t-values for both task factors in both voxels. Recall that the formula for t-values is given by:

$$([3.14]) t = \frac{c' B}{\sqrt{MS_e c'(X'X)^{-1} c}} \text{ with } df = N - H - 1$$

Let's start with B. B refers to an entire matrix containing b-values for all factors for each voxel. These have been calculated in the previous part of the example:

$$B \text{ for voxel 1 : } B_{\text{voxel1}} = \begin{pmatrix} 5.54 \\ 3.32 \\ 3.26 \end{pmatrix} \text{ and for voxel 2: } B_{\text{voxel2}} = \begin{pmatrix} 7.54 \\ 9.81 \\ 5.45 \end{pmatrix}$$

So the scalar by which to multiply the factor, the b-value, is known for each factor. To calculate the t-value for the first factor in voxel 1, we need the first element of the B matrix for voxel 1. This is accomplished by multiplying the B matrix with a contrast vector. To select the first element, this contrast vector becomes:

$$c' = (1 \quad 0 \quad 0)$$

and to select the second element (b-value for the second factor), this vector becomes:

$$c' = (0 \quad 1 \quad 0)$$

So that $c' B$ for the first factor becomes:

$$(1 \quad 0 \quad 0) \times \begin{pmatrix} 5.54 \\ 3.32 \\ 3.26 \end{pmatrix} = 5.54$$

Next, we calculate the residual mean square:

$$([3.11]) MS_e = \frac{e'e}{N - H}$$

whereby $e(\text{error})$ is defined as:

$$([3.7]) \text{ error} = Y - XB$$

This is depicted in figure 19 for both voxels.

[figure 19]

Recall that Y (the data) for voxel 1 and for voxel 2 is a 31×1 matrix (31 scans, 1 column) and that X (the design matrix) is a 31×3 matrix (31 scans, 2 factors + intercept). So calculating the error for every data point (i.e. scan) results in a 31×1 matrix:

$$([3.7]) \quad e = Y - XB = (31 \times 1) \text{ matrix}$$

containing the difference per scan between Y (real data) and XB (estimation of that data).

Consequently, the residual sums of squares becomes one number, because of the matrix multiplication. The dimensions are given by:

$$[3.7b] \quad e'e = (1 \times 31) \times (31 \times 1) = (1 \times 1) \text{ matrix}$$

Now the residual mean square of the variance can be calculated using:

$$([3.8]) \quad MS_e = \frac{e'e}{N - H} = \frac{15.68}{31 - 3} = 0.56$$

Finally, we calculate $c'(X'X)^{-1}c$, which reflects a division by the variance of the first factor, corrected for the correlation with the other factors. Calculation of this element is done in three steps:

First, we calculate the sums of squares and cross products matrix:

$$[3.17] \quad (X'X) = \begin{pmatrix} 5.936 & -2.065 & 0 \\ -2.065 & 5.936 & 0 \\ 0 & 0 & 31 \end{pmatrix}$$

which has the sums of squares for each factor on the diagonal and the cross products between the various factors as non-diagonal elements. These are weighted variances and covariances (divide the elements by $N-1$ to obtain the variances and covariances).

Second, this square matrix is then inverted:

$$[3.19] \quad (X'X)^{-1} = \frac{1}{\det} \begin{pmatrix} 184 & 64 & -0 \\ 64 & 184 & -0 \\ -0 & -0 & 30.967 \end{pmatrix} = \begin{pmatrix} 0.192 & 0.067 & -0.000 \\ 0.067 & 0.192 & -0.000 \\ -0.000 & -0.000 & 0.032 \end{pmatrix}$$

4.5.1 EXPLORATION

To understand what the elements of this matrix mean, a little exploration:

Element(1,1) of the matrix $(X'X)^{-1}$ is called the minor of element (1,1) of the matrix $(X'X)$, and is the determinant of the matrix:

$$\begin{pmatrix} 5.936 & 0 \\ 0 & 31 \end{pmatrix}$$

which is obtained by deleting the first row and the first column of the matrix $(X'X)$. We get:

minor(1,1) = $(5.936 \times 31) - (0 \times 0) = 184$. This value reflects the generalized variance of the other factors of the model (factor 2 and 3). The determinant of the entire model X is 960. This determinant reflects the generalized variance of all the factors within the model.

The first element of the matrix $(X'X)^{-1} = 184/960 = 0.1917$. This element depicts the ratio between variance explained by the factors G and the generalized variance of the entire model (all factors H). Hence this value represents the amount of variance explained by the factor j.

Third, apply the contrast vector to this inverted matrix, so that the first element of this matrix is selected:

$$c'(X'X)^{-1}c = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0.192 & 0.067 & -0.000 \\ 0.067 & 0.192 & -0.000 \\ -0.000 & -0.000 & 0.032 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = 0.192$$

The t-value for the first factor in the voxel 1 is subsequently calculated by:

$$([3.14]) \quad t = \frac{c'B}{\sqrt{MS_e c'(X'X)^{-1}c}} = \frac{5.5386}{\sqrt{0.56 \times 0.1917}} = 16.91 \text{ with } df = N - H - 1$$

The other t-values are computed in the same way:

	Voxel 1		Voxel 2	
	b-value	t-value	b-value	t-value
Task 1	5.5386	16.91	7.5386	12.70
Task 2	3.3173	10.13	9.8173	16.54
Intercept	3.26	-	5.45	-

Note that the highest b-value (voxel 2, task 2) is NOT associated with the highest t-value. This is due to the higher error variance (residual mean square, MS_e) in voxel 2. Let's work out the numbers for the second factor for voxel 2:

$$([3.14]) t = \frac{c'B}{\sqrt{MS_e c'(X'X)^{-1}c}} = \frac{9.8173}{\sqrt{1.84 \times 0.1917}} = 16.54 \text{ with } df = N - H - 1$$

4.6 Making a t-map for a contrast between two factors

In the previous section, the b-value for one factor was rewritten as a t-value for that factor, thereby reflecting how strong the b-value deviates from zero. However, in many experiments one would like to answer the question if there are regions that are significantly more active during one condition compared to another. For example: Is activation in the motor cortex higher during the experimental condition than during the control condition? Answering this question involves comparing the b-values of both conditions (factors).

Recall the formula for t:

$$([3.14]) t = \frac{c'B}{\sqrt{MS_e c'(X'X)^{-1}c}} \text{ with } df = N - H - 1$$

The B matrix contains the b-values for each separate factor. We simply subtract the second b-value from the first, which is done by multiplying the B matrix with a contrast matrix similar to the one used to select only the first b-value.

Whereas for one factor (from a model with nine factors) the contrast vector was:

$$c' = (1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$$

for a two-factor t-test, for example factor 1 versus factor 2, the contrast matrix for the same model becomes:

$$c' = (1 \ -1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$$

Multiplying the latter contrast vector with the B matrix will result in subtracting the second b-value (for the second factor) from the first b-value (from the first factor).

Subsequently, the multiplication

$$c'(X'X)^{-1}c$$

will now yield a combination of the weighted variances of both the first and the second factor plus a correction for the covariance between these two factors.

Finally, we need to determine the size of MS_e . The residual mean square (error, MS_e) is a fixed number for each voxel and is dependent upon the entire model (recall, error = $Y - XB$). So for all contrasts that can be made between the factors of the model, this component does not change.

4.6 EXPLORATION

When only one factor was tested, $c'(X'X)^{-1}c$ gave a weighted representation of the unique variance of the first factor. In a contrast between two factors, this gives:

$$c'(X'X)^{-1}c = \begin{pmatrix} 1 & -1 & 0 \end{pmatrix} \times \begin{pmatrix} \text{var}(1) & \text{cov}(1,2) & \text{cov}(1,3) \\ \text{cov}(2,1) & \text{var}(2) & \text{cov}(2,3) \\ \text{cov}(3,1) & \text{cov}(3,2) & \text{var}(3) \end{pmatrix} \times \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix} = (\text{var}(1) - \text{cov}(2,1)) - (\text{cov}(1,2) - \text{var}(2)) = (\text{var}(1) + \text{var}(2)) - 2 \times \text{cov}(1,2)$$

4.6.1 An example - continued again

In this example, we will perform a t-test between the two task factors for the data in voxel 1, to determine if factor 1 explains significantly more variance than factor 2 in this voxel.

The contrast becomes: $c' = (1 \quad -1 \quad 0)$

$$\text{So } c'B = \begin{pmatrix} 1 & -1 & 0 \end{pmatrix} \times \begin{pmatrix} 5.54 \\ 3.32 \\ 3.26 \end{pmatrix} = 5.54 - 3.32 = 2.22$$

And

$$c'(X'X)^{-1}c = \begin{pmatrix} 1 & -1 & 0 \end{pmatrix} \times \begin{pmatrix} 0.192 & 0.067 & -0.000 \\ 0.067 & 0.192 & -0.000 \\ -0.000 & -0.000 & 0.032 \end{pmatrix} \times \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix} \approx 0.192 + 0.192 - 0.067 - 0.067 \approx 0.25$$

We can now determine the t-value of the difference between the two factors in voxel 1 by applying the general formula:

$$([3.11]) t = \frac{c'B}{\sqrt{MS_e c'(X'X)^{-1}c}} = \frac{2.22}{\sqrt{0.56 \times 0.25}} = 5.93 \text{ with } df = N - H - 2$$

5 Several statistical issues

5.1 Introduction

In this chapter, we will look at several important issues. In 5.2 we will discuss the implications of multicollinearity. Multicollinearity refers to the correlation between the factors in the design matrix. Before scanning subjects with a particular design, one should calculate the multicollinearity of that design, because high a correlation between factors can results in low t-values. In 5.3 we will discuss thresholding of statistical maps (t-maps) so one can determine 'which voxel is significant'.

5.2 Multicollinearity

Multicollinearity refers to the correlation between the factors in the design matrix. Multicollinearity is a form of *singularity* of the matrix. When a matrix is *singular*, then one factor can be rewritten as a linear combination of (some of) the other factors. In that particular case, adding this factor to the model does not add anything in terms of extra information concerning the signal you want to explain, since this information is already present in the other factors. There are degrees of multicollinearity, from none to complete singularity. Even small degrees of multicollinearity can pose a problem for the analysis, because of three reasons:

5.1 EXPLORATION

An example for calculating the multiple correlation between factor j and the other three factors in a design:

$$[3.XX] \quad R_{j,G}^2 = r_{j1}^2 + r_{j2.1(s)}^2 + r_{j3.12(s)}^2$$

where

$$[3.XX] \quad r_{j2.1(s)}^2 = \left(\frac{r_{j1} - r_{j1}r_{12}}{\sqrt{1 - r_{12}^2}} \right)^2$$

and

$$[3.XX] \quad r_{j3.12(s)}^2 = \left(\frac{r_{j3.2} - r_{j2.1(s)}r_{23.1}}{\sqrt{1 - r_{23.1}^2}} \right)^2$$

The easiest way to calculate the squared multiple correlation is by using SPSS (linear regression).

There are degrees of multicollinearity, from none to complete singularity. Even small degrees of multicollinearity can pose a problem for the analysis, because of three reasons:

1. *limits the amount of variance explained by the model.* Because a particular factor (or factors) can be partially rewritten in terms of other factors, these factors in fact all explain the same part of the variance in the signal, instead of explaining additional variance. As a consequence of this overlap in variance explanation, the model fit will be non-optimal and therefore the error variance will remain unnecessary large. (alteration of the model may fix this)

2. *makes determining the importance of a given factor difficult.* Because multiple factors explain the same part of the variance in the

fMRI signal, it is difficult to decide which of these factors is 'responsible' for this variance. For example, a task factor may be correlated with the linear trend factor. Then, task activation can be attributed to either the task factor or the linear trend factor. Maybe the task factor is not important at all because a linear trend is present in the data.

3. *increases the standard error of the b-values and hence decreases t-values.* To better comprehend this, inspect the formula for the standard error of the b-values:

$$[3.1XX] \quad se(B_j) = \sqrt{\frac{1 - R_{y,H}^2}{(1 - R_{j,G}^2)(N - H - 1)}}$$

The term $R_{j,G}^2$ refers to the squared multiple correlation between factor j and the other factors. In other words, this term reflects the collinearity for this factor in the model. The greater this term becomes, the greater the estimated standard error of b (for factor j) becomes. This, in turn, will result in a smaller t -value since the b -value is divided by this estimated standard error of b .

To get an indication of the variance which explained uniquely by a particular factor, one can correct the factor variance for the squared multiple correlation with the other factors using the variance inflation factor:

$$[3.16] \quad \frac{1}{1 - R_{j,G}^2}$$

where G refers to the factors in the design matrix minus factor j , and hence $R_{j,G}^2$ denotes the explained variance of factor j by the other factors.

To investigate the size of the multicollinearity in the design matrix, one can calculate the multiple correlation between a specific factor and the remaining factors in the design (see 4.1 EXPLORATION).

5.3 Thresholding

After the statistical analysis, we have statistical maps with t -values for each (brain) voxel (t -maps). The question now becomes which t -values represent significant activation in the brain. To answer this question, a threshold is set above which t -values are *significant*. Significant means that the occurrence of such a high (or low) t -value within the normal (student- t) distribution is very unlikely. Normally, a threshold is set so that significant values are those that have a likelihood of five percent or less to occur in that particular distribution. Using this threshold, we can determine, for each individual voxel, if the t -value has a chance of 5 percent or less to occur in the known distribution. Since there are approximately 16,000 voxels in a brain, 16,000 tests are performed to determine if a particular t -value is significant. In every test, there is a chance of 5 percent that the value will be called 'significant'. So, over the entire brain, setting the threshold at 5 percent will result in 5 percent of the values to be called significant. Even in a completely random brain with 16,000 this would lead to 800 'significant' voxels. A correction is needed if one wants to set the significance threshold at 5 percent for each voxel.

5.2.1 The Bonferroni correction for multiple tests

This method offers a correction for thresholding for multiple tests by dividing the threshold by the number of independent tests that are performed:

$$[3.17] \quad \alpha_{overall} = \frac{\alpha}{N}$$

for N = number of independent tests. So if $\alpha = 0.05$ and $N = 16,000$ (=number of voxels) then:

$$([3.17]) \alpha_{overall} = \frac{\alpha}{N} = \frac{0.05}{16,000} = 0.000003125$$

So voxel-wise, t-values should be thresholded at an $\alpha = 0.000003125$. The corresponding t-value depends upon the degrees of freedom. For an analysis of a single subject, this is number is given by the test that is performed (see [3.14]).

Applying the Bonferroni correction with the number of voxels as the number of independent tests is rather conservative, because the actual number of independent observations IS NOT the number of voxels. Rather, the voxel-values are spatially correlated (see 2.1.5 SMOOTHING). In practice, this means that if one particular voxel-value is not-significant, it is likely that its immediate neighbour also is not significant. The number of resels (see 2.1.5 SMOOTHING) is a better estimation of the true number of independent elements. (If one chooses to use resels for the Bonferroni correction, then the resolution of the t-map should be equal to the resel size).

However, the number of resels also does not represent the true number of independent observations since resels themselves are correlated, although this correlation is known. The true number of independent observations is not easy to work out, so a different approach is needed.

5.2.1 EXPLORATION

Why a threshold of $t = 4.51$?

The critical t-value of 4.51, which is commonly applied in the UMC-Utrecht, is based on a voxel-wise alpha of 10 percent. This value reflects a two-tailed alpha of 5 percent, since there is no a priori assumption whether the test value is either in the low 5 percent or the high 5 percent of the distribution. This alpha is divided by 16,000 voxels, resulting in an overall alpha of 0.00000625. This value is converted to a t-value using a t-distribution with infinite degrees of freedom. Of course, the number of degrees of freedom is not infinite, rather the number of scans minus the number of factors (minus 1 or 2, depending on the contrast). However, above 500 degrees of freedom, the exact number does not matter that much anymore. (actually, the critical t-value based on this alpha an number of voxels should be **4.526**)

And how about $t = 3.09$?

Whereas a threshold of $t = 4.51$ reflects an alpha of 5 percent corrected for multiple tests, a threshold of $t = 3.09$ refers to an two-tailed alpha of 0.002, NOT corrected for multiple tests. This threshold is often used for exploratory purposes. For example, when one wants to know if there is any trend present in the data.

How can I calculate the correct threshold for my own data ?

You can calculate your very own critical t-value using the t-inverse (T.INV) function in MS Excel. This might result in a different critical t-value if the number of brain voxels of your data (falling within the brain mask) is significantly lower or higher than the 16,000 associated with a t-value of 4.51.

6 Group analysis

6.1 Introduction

Performing a group analysis allows generalization of the results to a population, for example the population of healthy subjects or schizophrenia patients. There are two main strategies for performing a group analysis, namely look at:

1. group effects in the entire brain
2. mean group activation for different conditions in Regions of Interest

The first method involves performing statistics over the entire brain. A disadvantage is the correction of the significance threshold for multiple testing. This might prove to be a very stringent correction if differences between two conditions are very small. As an alternative, one can define Regions of Interest (ROI's) and look at the mean activation differences between conditions within these ROI's, thereby allowing for lower thresholding. A slightly different approach is to define Volumes of Interest. Originally, the volume of interest in the entire brain, consisting of approximately 16,000 voxels. However, if one expects differences between conditions to occur only within the visual cortex, one does not want to correct for voxels in which no effect is expected. To bypass the correction for the entire brain, one can select any area of the brain as the volume of interest (VOI). The subsequent correction for multiple testing is hence less stringent.

6.2 Standard group analysis

Significant values in the group-map *reflect activation that is stable over subjects*. The voxel values do not need to be significant on an individual level for the group voxel value to become significant. For example, consider the example in figure 23. The t-value (see also 6.1 EXPLORATION) for a specific voxel is obtained from each individual brain. Note that none of the individual t-values is above the threshold of 't = 4.51' (see 4.5.1 EXPLORATION). The next step is to calculate the group t-value, which represents the stability of the individual t-values. One possibility to obtain the group-value for this voxel is to perform a standard one-sample t-test:

$$[6.1] \quad t_{group} = \sqrt{n} \frac{\bar{t}}{SD(t)} \text{ with } df = n - 1, \text{ and 'n' is the number of subjects.}$$

[figure 23]

Applying this formula to the data of figure 23 gives:

$$([6.1]) \quad t_{group} = \sqrt{n} \frac{\bar{t}}{SD(t)} = \sqrt{7} \frac{2.74}{0.2356} = 30.38$$

The resulting t-value for the group seems to be rather high (t = 30.38), compared to the low individual t-values. However, because these values do not differ so much between individuals, the standard deviation (SD) is rather low. Subsequently, the group t-value can be high, reflecting the fact that the activation in that particular voxel is present in all subjects in a stable manner.

To determine the significance of this value (actually, to assess the chance that such a t-value occurs in a particular distribution), one needs to know from which distribution this t-value is obtained. There are two ways to describe the distribution for this t-value:

1. use the distribution from the group of t-values over which the test is performed (*voxel-based standard deviation*)
2. use the average distribution over all voxels (*pooled standard deviation*)

6.1 EXPLORATION

Should a group analysis be performed over individual b or t values?

In the example discussed above, t-values were used to calculate the group t-value. The advantage of using t-values is that these are normalized for the error variance in each subject. Subjects who have high b-values, but also high error variance, will have low t-values, reflecting the fact that the high b-values are not so valid. So then the group analysis will be done over the significances of the b-values.

However, consider for example two groups; a patient group and a healthy control group. For the sake of the example, let's assume that the error variance in the patient group is higher (maybe due to movement related noise). It is possible that a patient and a healthy control have the same effect-size (b-value for a specific factor), but differ in the amount of error. While the b-values are equal, the t-values will differ.

6.3 Standard-deviation in group analysis

The voxel-based standard deviation (SD) is calculated using the individual values for that voxel in the group:

$$[6.2] \quad SD_{\text{voxelbased}} = \sqrt{\frac{\sum (x - \bar{x})^2}{n - 1}}$$

where n is the number of subjects and x is either t or b statistic (see 6.1 EXPLORATION).

In contrast, the pooled SD is merely the mean SD over all voxels in the brain:

$$[6.3] \quad SD_{\text{pooled}} = \frac{\sum SD_N}{N - 1}$$

where N is the number of voxels.

[figure 21] & [figure 22]

Using the voxel-based SD allows for differences in SD between voxels. In contrast, the pooled SD method assumes equal SD in all (brain) voxels. The reason to assume equality is that the data in all voxels is the same, namely a number of normalized values (t-values). So there is, a priori, no reason to assume differences in SD between voxels. However, using the pooled SD will lead to more degrees of freedom used to describe the distribution. Whereas the voxel-based SD yields n-1 degrees of freedom (formula [6.1]), the pooled SD method yields N-1 degrees of freedom, where N is the number of voxels. As becomes clear from figures 21 and 22, the more degrees of freedom a t-distribution has, the more it will resemble a standardized

distribution. A standardized distribution, or Z-distribution, has a mean value of zero and an SD of 1. Because of this overlap between t and Z, the group t-map is also called a Z-map. Z-values are calculated using an adaptation of the formula for a standard one-sample t-test:

$$([6.1]) \quad Z_{group} = \sqrt{n} \frac{\bar{t}}{SD_{pooled}} \quad \text{with } df = N - 1$$

where n = number of subjects and N is the number of voxels.

(Note, however, that since the pooled SD is not always 1, the resulting value is not a true Z value.)

The advantage of having many degrees of freedom ($N-1$) is that compared to having only $n-1$ degrees of freedom is that lower t-values are considered more significant.

6.4 Region of Interest analysis

Region of Interest (ROI) analysis is used when the Bonferroni correction for multiple tests is too stringent for the expected effects (differences) between task conditions. This approach can be 'justified' by arguing that one has to correct only for areas (voxels) which are interesting from the perspective of the task. Defining which areas are of interest can be done on the basis of (a) *previous findings in the literature*, (b) *own pilot data*, or (c) *the experimental data*.

Defining the ROI's based on the experimental data requires some form of group analysis which will show these areas. For example, suppose one has a task consisting of three different conditions. One could perform a group analysis in which task activation is contrasted with rest. Activation on the contrast map then reflects task-related activation.