Neuroimage Processing

Instructor: Moo K. Chung

mkchung@wisc.edu

Lecture 2-3.

General Linear Models (GLM)

Voxel-based Morphometry (VBM)

September 11, 2009

What is GLM

The general linear model (GLM) is a very flexible and general statistical framework encompassing a wide variety of fixed effect models such as the multiple regressions, the analysis of variance (ANOVA), the multivariate analysis of variance (MANOVA), the analysis of covariance (ANCOVA) and the multivariate analysis of covariance (MANCOVA)

The parameters of the model are mainly estimated by the least squares estimation and has been implemented in many statistical packages such as R or Splus and brain imaging packages such as SPM and fMRI-STAT.

Model

Let y_i be the response variable, which is mainly coming from images and $\mathbf{x}_i = (x_{i1}, \cdots, x_{ip})$ to be the variable of interest and $\mathbf{z}_i = (z_{i1}, \cdots, z_{ik})$ to be nuisance variables corresponding to the i-th subject. We assume there are n subjects. Then we have a GLM

$$y_i = \mathbf{z}_i \lambda + \mathbf{x}_i \beta + \epsilon_i$$

where $\lambda = (\lambda_1, \cdots, \lambda_k)'$ and $\beta = (\beta_1, \cdots, \beta_p)'$ are unknown parameter vectors to be estimated. We assume ϵ to be the usual zero mean Gaussian noise. Then we determine the significance of the variable of interests $\mathbf{x}_i$ by testing the null hypothesis

$$H_0 : \beta = 0 \text{ vs. } H_1 : \beta \neq 0.$$

Inference

The fit of the reduced model corresponding to $\beta = 0$, i.e. $y_i = \mathbf{z}_i\lambda$, is measured by the sum of the squared errors (SSE):

$$\mathrm{SSE}_0 \quad = \quad \sum_{i=1}^{n}(y_i - \mathbf{z}_i\widehat{\lambda}_0)^2,$$

where $\widehat{\lambda_0}$ is the least squares estimation obtained from the reduced model. Similarly the fit of the full model corresponding to $\beta \neq 0$, i.e. $y_i = \mathbf{z}_i\lambda + \mathbf{x}_i\beta$, is measured by

$$\mathrm{SSE}_1 \quad = \quad \sum_{i=1}^{n}(y_i - \mathbf{z}_i\widehat{\lambda}_1 - \mathbf{x}_i\widehat{\beta}_1)^2,$$

where $\widehat{\lambda}_1$ and $\widehat{\beta}_1$ are estimated from the full model.

Then under H_0, the test statistic is the ratio

$$F = \frac{(\mathrm{SSE}_0 - \mathrm{SSE}_1)/p}{\mathrm{SSE}_0/(n-p-k)} \sim F_{p,n-p-k}.$$

The larger the F value, it is more unlikely to accept H_0.

Estimation

The unknown parameters are estimated via the least squares method. The detailed exposition of the least squares estimation using the matrix inversion is necessary for numerical implementation in `MATLAB` and it is the basis of SPM and fMRI-STAT. The reduced model (2.1) can be written in a matrix form

$$\underbrace{\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}}_{\mathbf{y}} = \underbrace{\begin{pmatrix} z_{11} & \cdots & z_{1k} \\ \vdots & \ddots & \vdots \\ z_{n1} & \cdots & z_{nk} \end{pmatrix}}_{\mathbf{Z}} \underbrace{\begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{pmatrix}}_{\boldsymbol{\lambda}}. \tag{2.4}$$

By multiplying $\mathbf{Z}'$ on the both sides, we obtain

$$\mathbf{Z}'\mathbf{y} = \mathbf{Z}'\mathbf{Z}\boldsymbol{\lambda}.$$

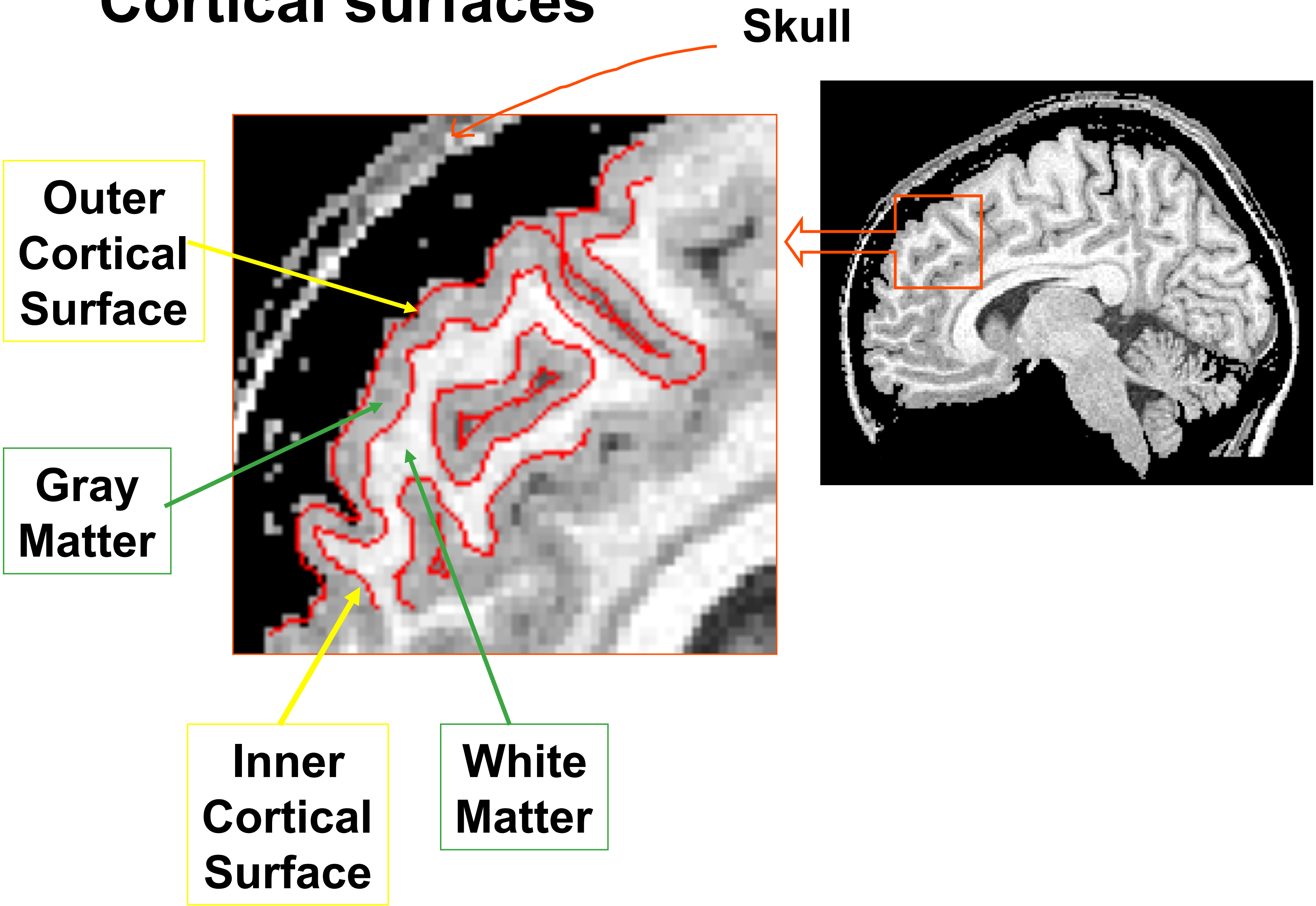
Now the matrix $\mathbf{Z}'\mathbf{Z}$ is a full rank and can be invertible if $n \geq k$. Therefore, the matrix equation can be solved by performing a matrix inversion

$$\widehat{\boldsymbol{\lambda}}_0 = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{y}.$$

Similarly the full model can be written in a matrix form by concatenating the row vectors $\mathbf{z}_i$ and $\mathbf{x}_i$ into a larger row vector $(\mathbf{z}_i, \mathbf{x}_i)$, and the column vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$ into a larger column vector $(\boldsymbol{\lambda}', \boldsymbol{\beta}')'$. Then the full model can be written in a matrix form and solved similarly.

Application:
GLM on cortical thickness

Cortical surfaces



Skull

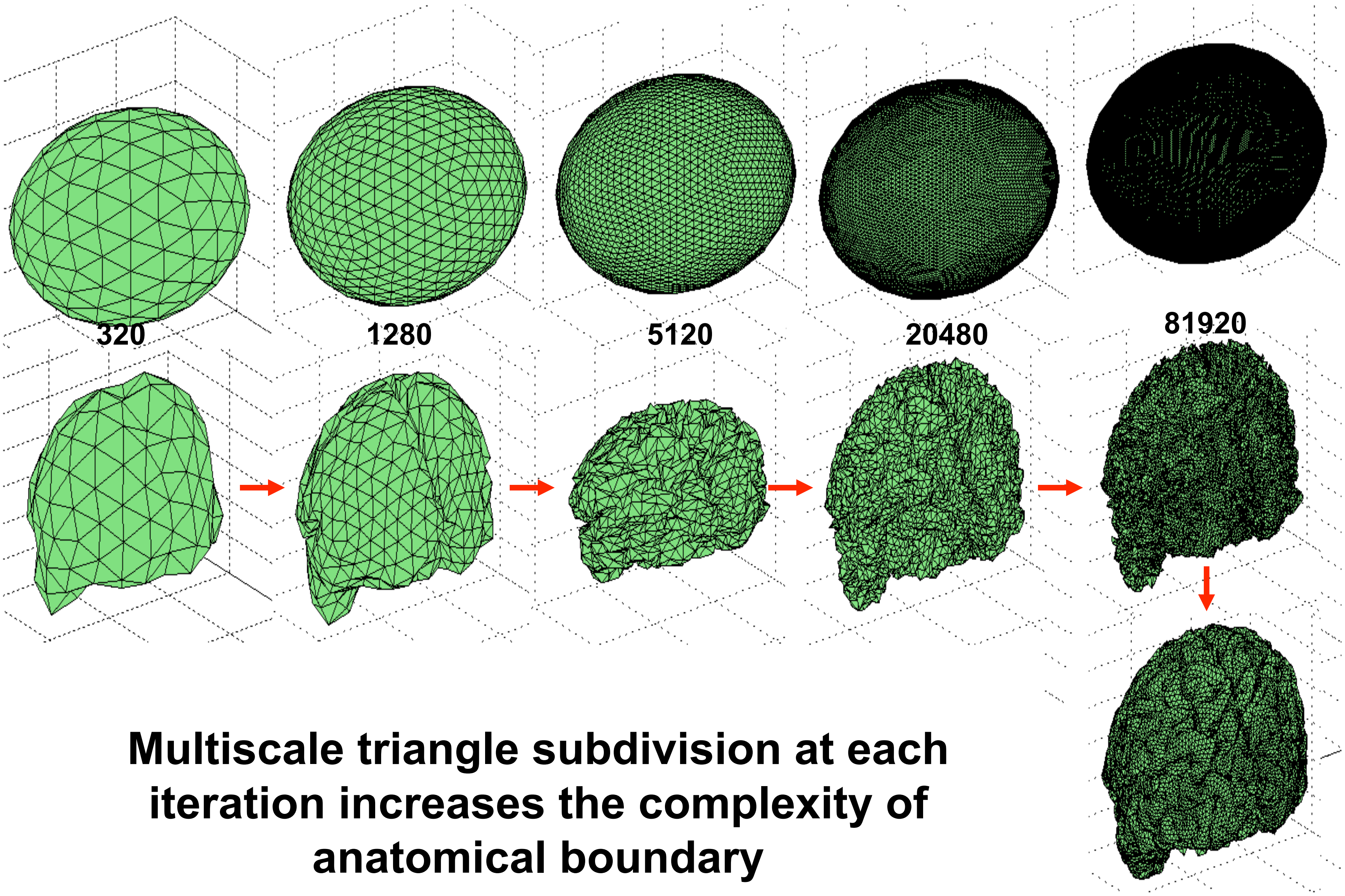
Outer Cortical Surface

Gray Matter

Inner Cortical Surface

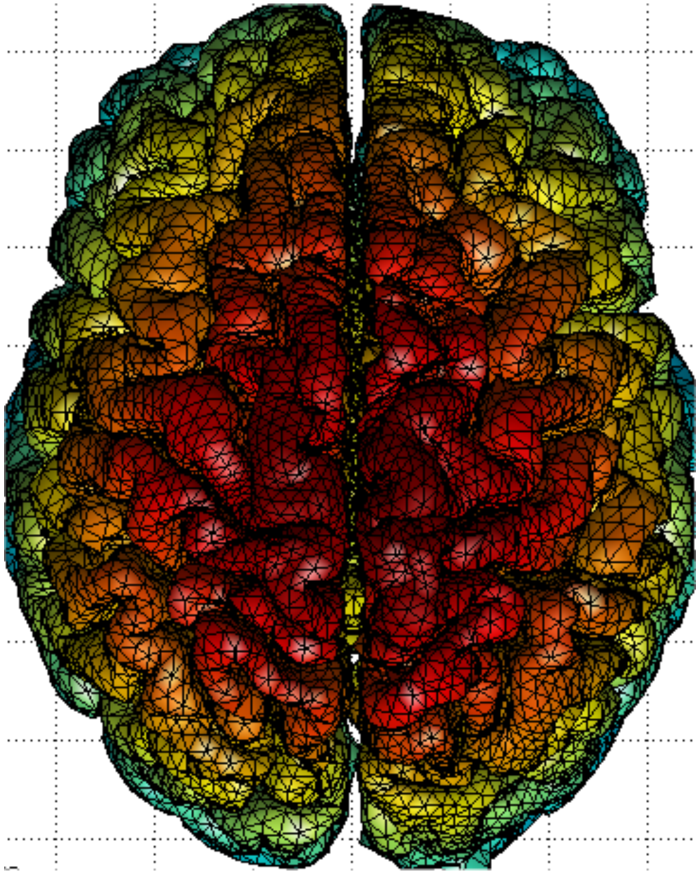
White Matter

Deformable surface algorithm McDonalds *et al*. (2001) NeuroImage



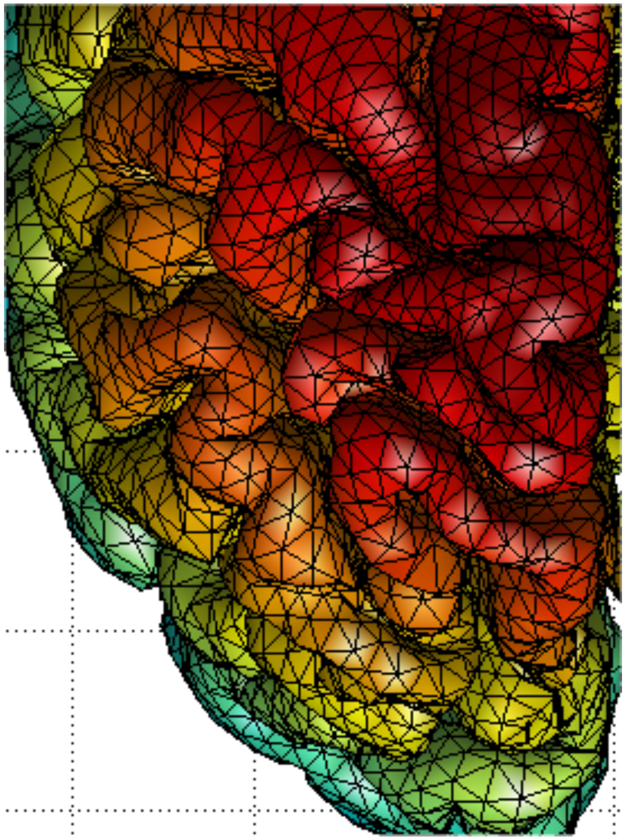
Multiscale triangle subdivision at each iteration increases the complexity of anatomical boundary

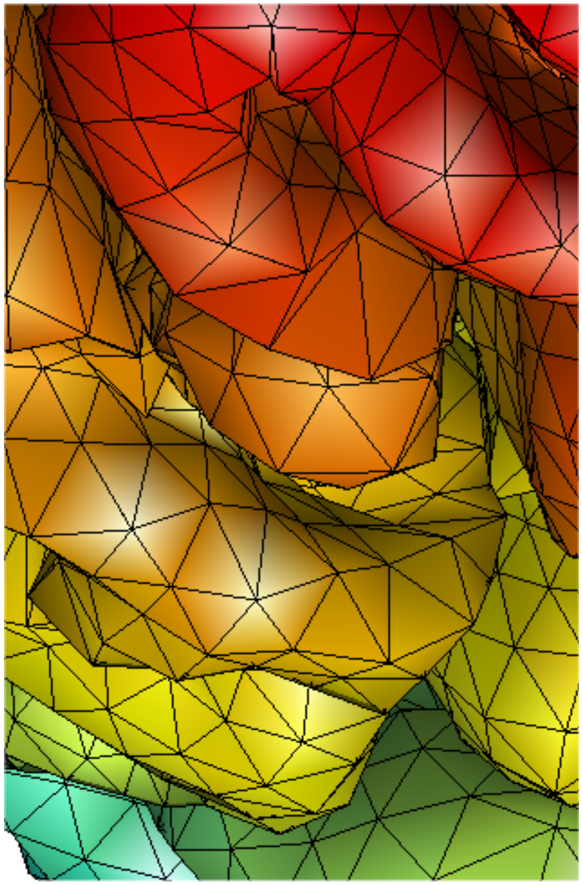
Cortical Surface
Polygonal mesh
Mesh resolution 3mm



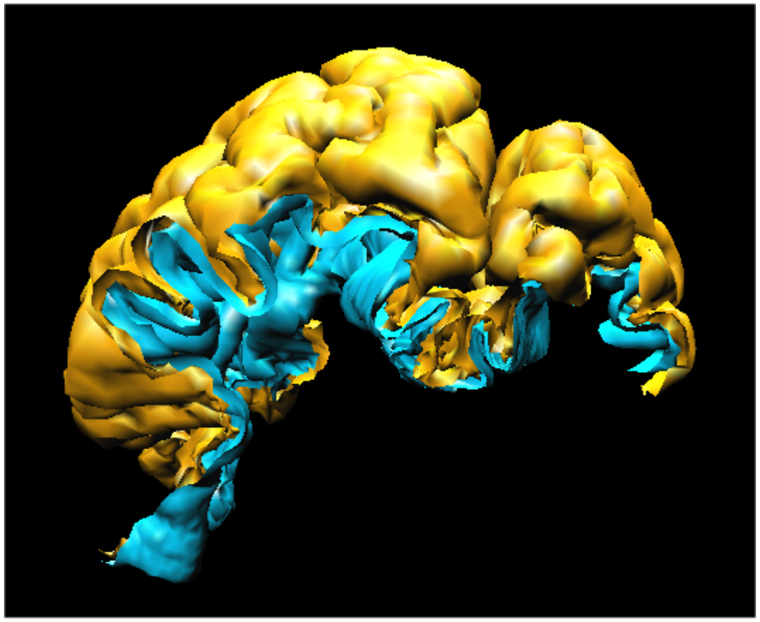
82,190 triangles

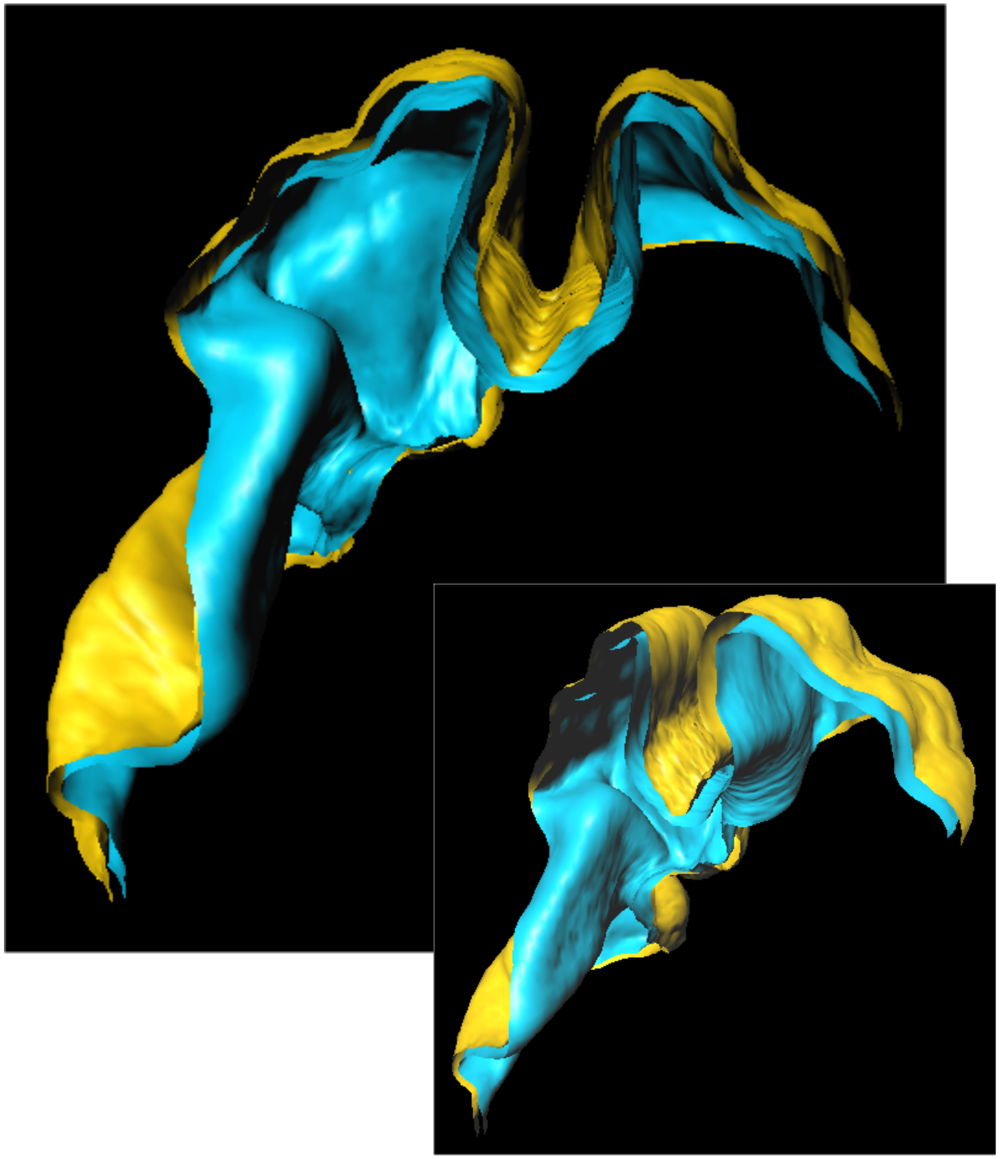
40,962 vertices





Outer and inner cortical surfaces





GLM on thickness

Given two groups (autism vs. control), we are interested in testing the significance of group difference on cortical thickness. So we consider the following GLM:

$$\texttt{thickness}_i = \lambda_1 + \beta_1 \cdot \texttt{group}_i + \epsilon, \tag{2.5}$$

where the dummy variable **group** is 1 for autism and 0 for control. This the case for $k = 1$ and $z_{i1} = 1$, and $p = 1$. The reduced model in this case is

$$\texttt{thickness}_i = \lambda_1.$$

The least squares estimation of λ_1 is simply the sample mean given by

$$\widehat{\lambda}_1 = \frac{1}{n} \sum_{i=1}^{n} \texttt{thickness}_i.$$

The test statistic F is then distributed as $F_{1,n-1-k}$, which is the square of the student t-distribution with $n-1-k$ degrees of freedom, i.e. t^2_{n-1-k}. The advantage of using the t-statistic is that unlike the F-statistic, it has two sides so we can actually use it to test for one sided alternative hypothesis $H_1 : \beta_1 \geq 0$ or $H_1 : \beta_1 \leq 0$. Therefore, the t-statistic map can provides the direction of the difference (if autism is thicker or thinner) that the F-statistic map cannot provide.

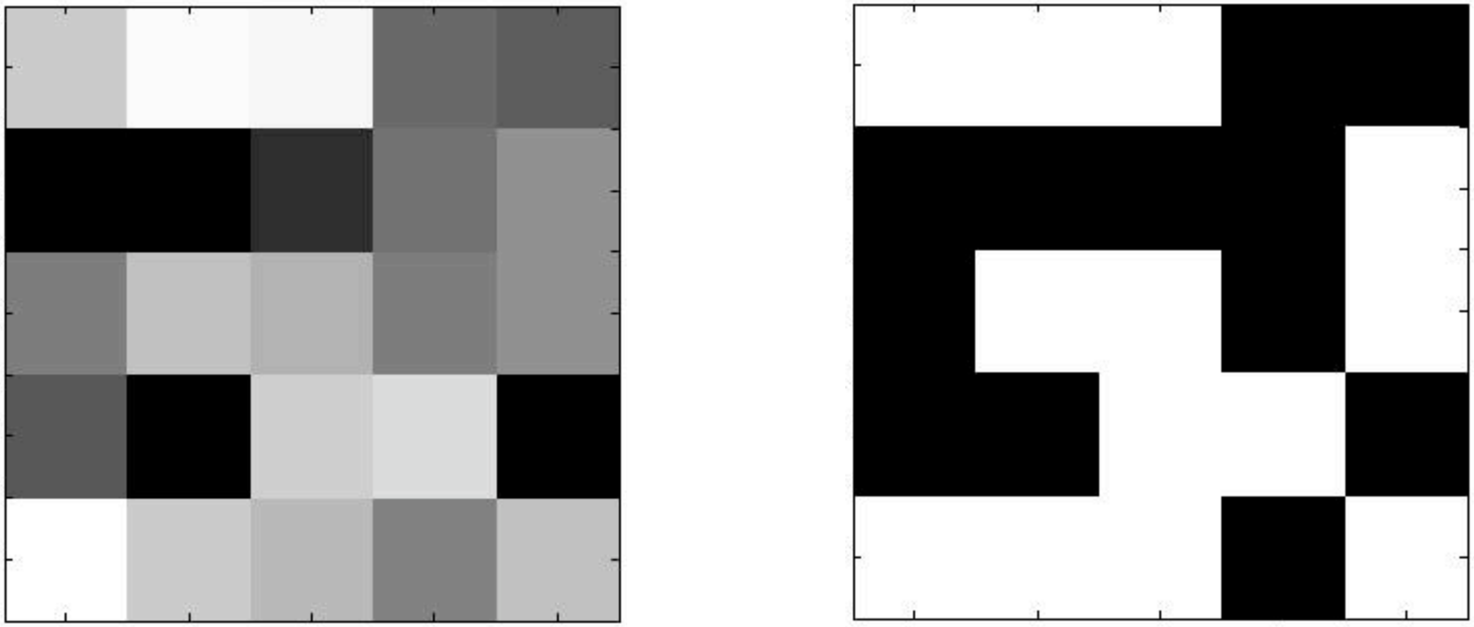
The model (2.5) is not necessarily a proper model since the model did not incorporate the possible confounding effects of brain size and age variations for each subject. In order to control the possible confounding effect of age, we consider consider following GLM:

$$\texttt{thickness}_i = \lambda_1 + \lambda_2 \cdot \texttt{age}_\texttt{i} + \beta_1 \cdot \texttt{group}_\texttt{i} + \epsilon_\texttt{i}, \tag{2.6}$$

MATLAB demonstration

Segmentation

- **Segmentation** is the partition of a digital image into multiple regions according to some criterion.



C. Phillips

Image Segmentation Methods

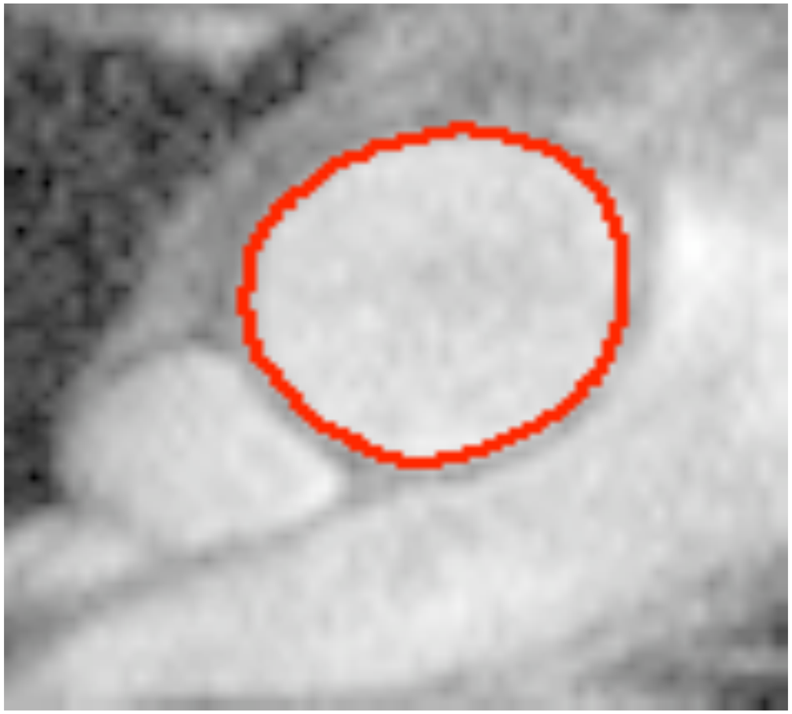
- Intensity histogram based approach

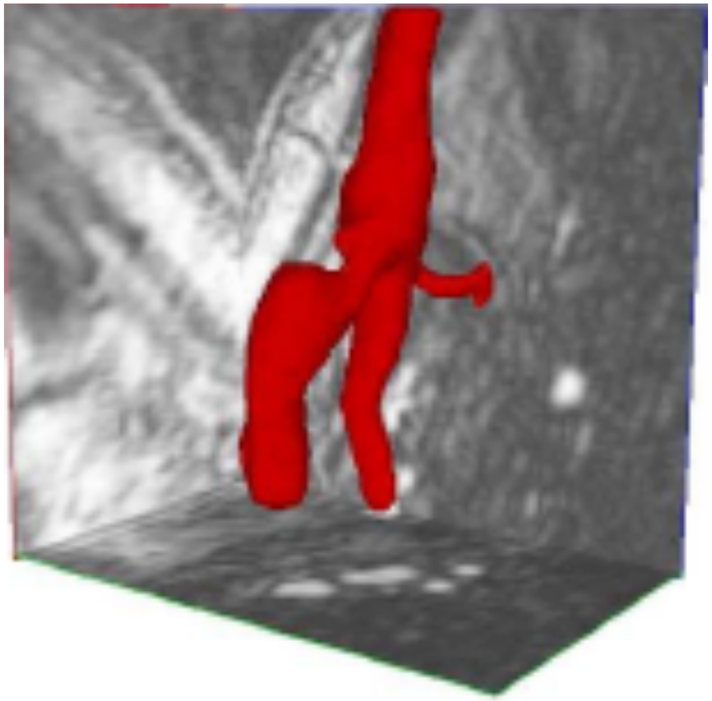
Gaussian mixture modeling → probabilistic output

- Shape based approach (PDE based)

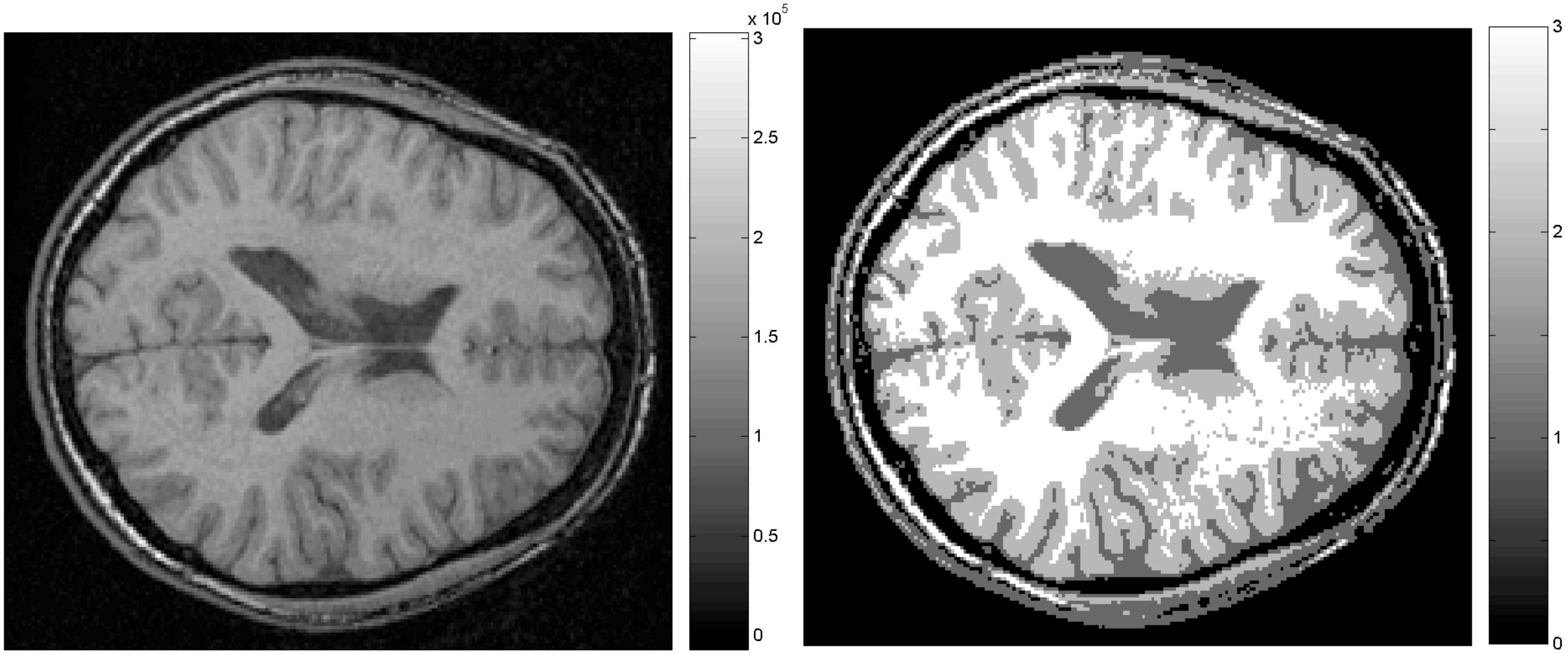
Active contour, deformable surface algorithms, level set → deterministic output

Example: CT image segmentation





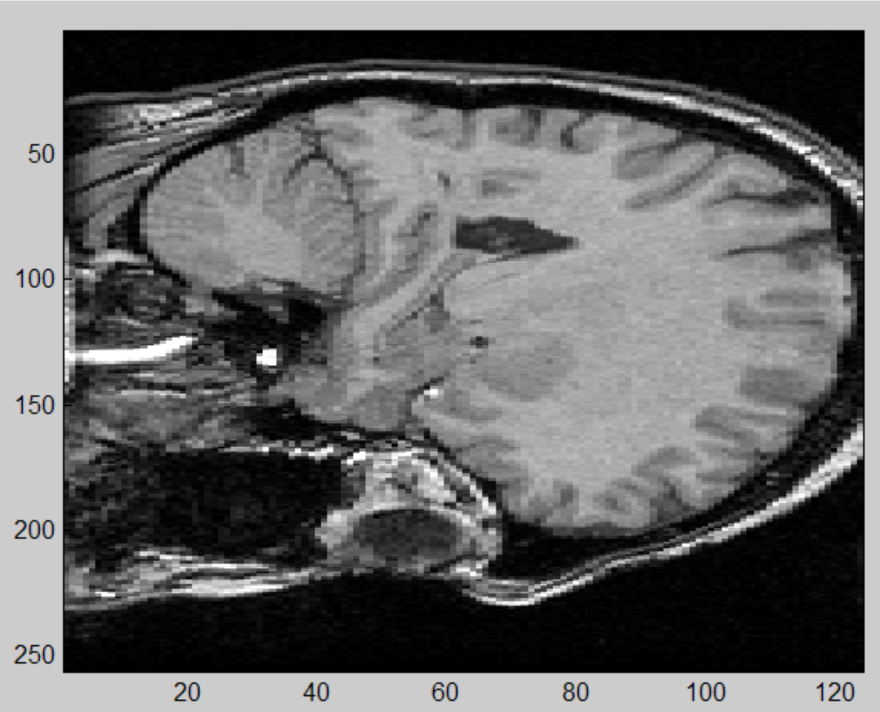
Example: MRI

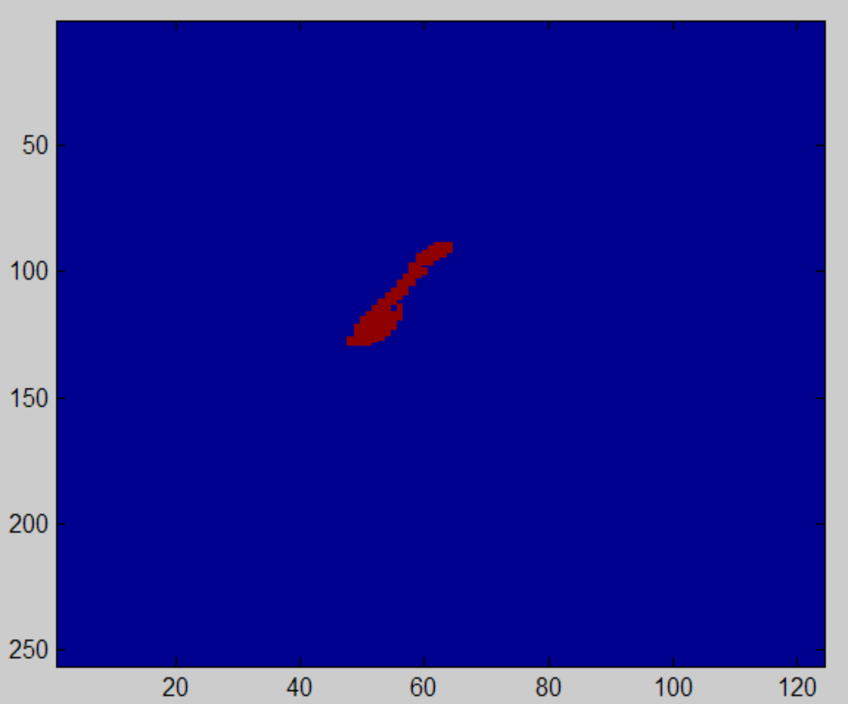


Application of deterministic segmentation → ROI volumetry

- This is a traditional approach
- ROI volumetry measures volume of a segmented region region of interest (ROI).

Example: Hippocampal volumetry





1. Manually or automatically segment hippocampus
2. Count the number of masked voxels
3. #number of voxel x volume of voxel

Probabilistic segmentation

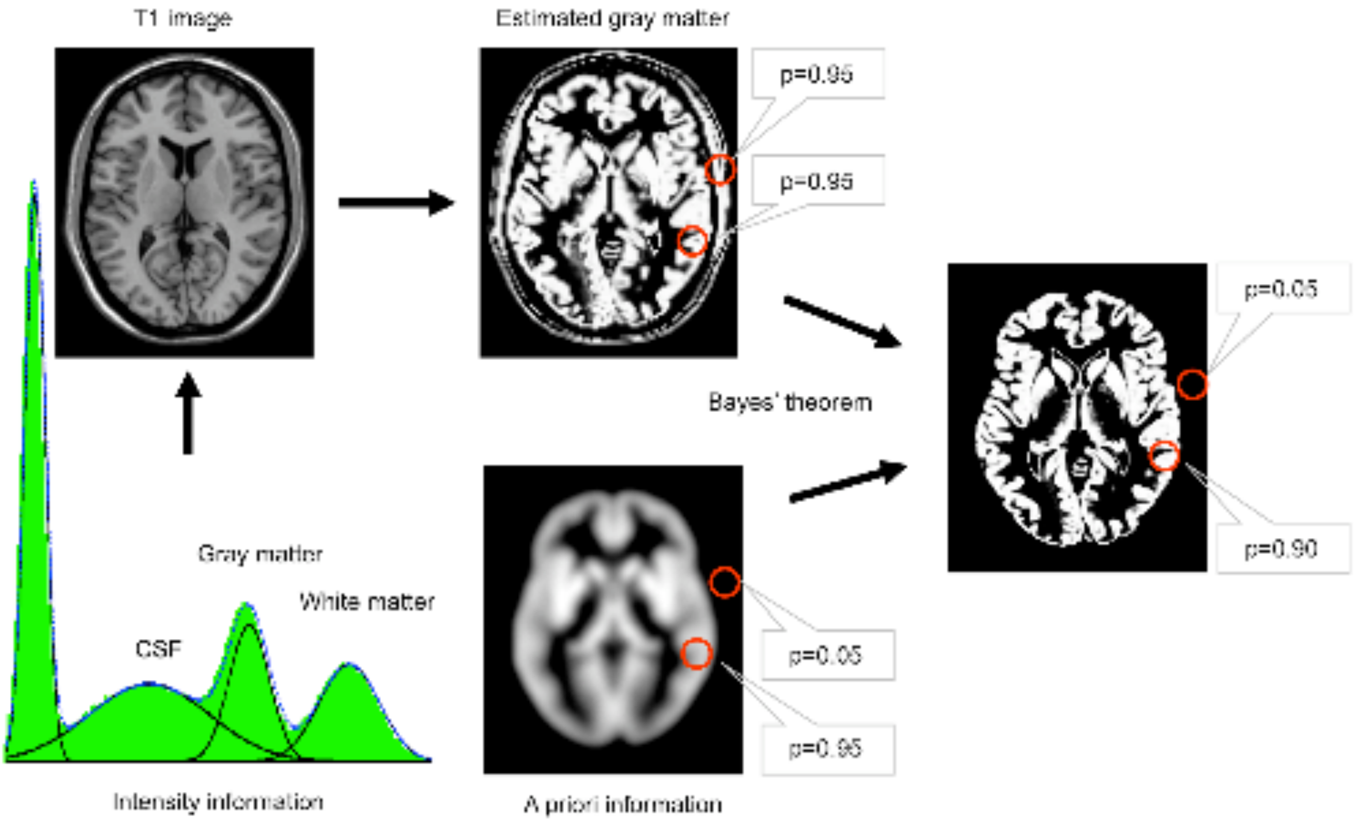
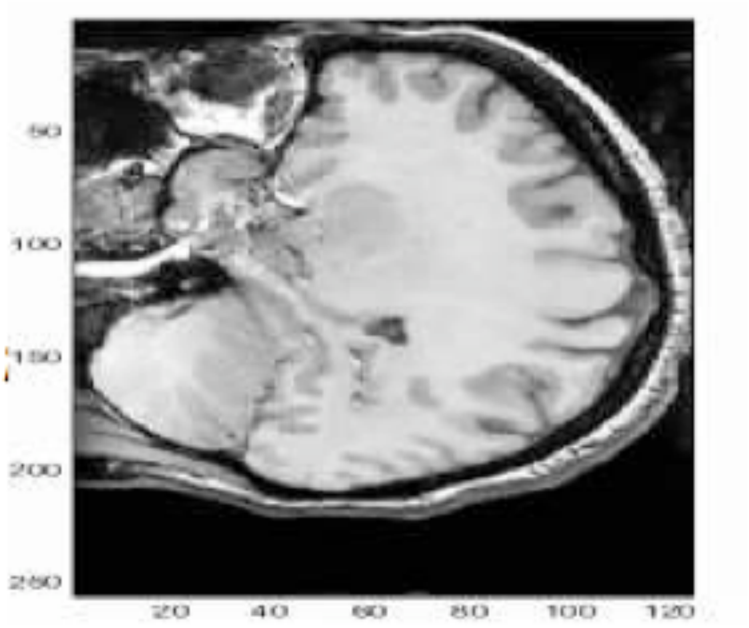
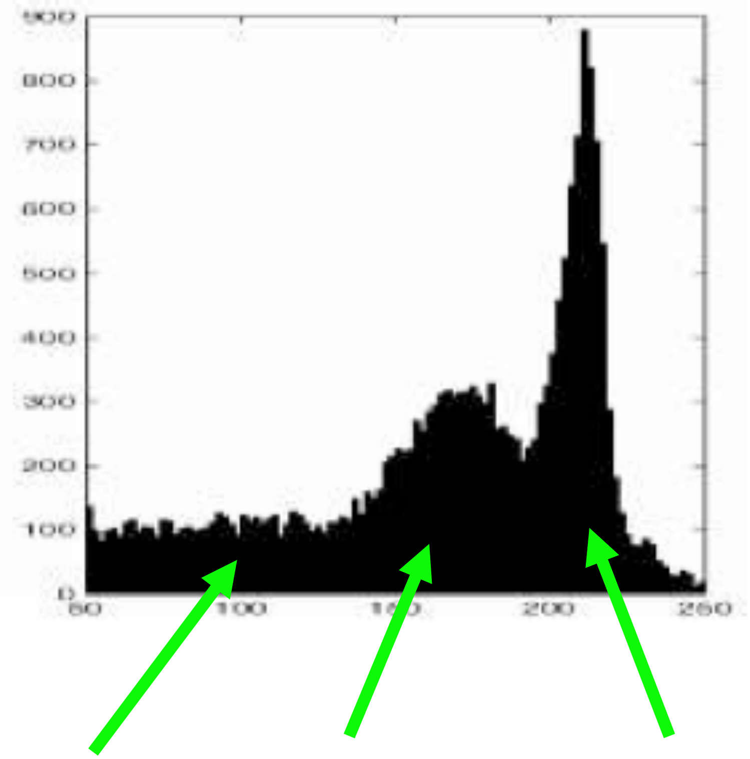


FIGURE 1 | Image segmentation using a priori information. In the first step, the image intensities of theT, image (upper left) are used to plot their frequencies in a histogram. Several peaks – corresponding to different image intensities of the tissue classes – can be differentiated. In the next step, gaussian curves for each tissue class are fitted into the histogram to estimate the probability of a voxel belonging to that tissue class (bottom left). A map for gray matter is shown (upper right) with the estimated probability for two selected locations (red circles). Based solely on a similar image intensity, the cerebral and the extracranial spot exhibit a similar probability for belonging to gray matter. This can be corrected by combining the image intensity-based information with prior information (below), e.g. using a Bayesian approach.

See Mietchen & Gaser, 2009

Image intensity histogram





CSF   gray matter   white matter

Bayesian framework

- Once we obtained all parameters of the Gaussian mixture model, we can compute the posterior tissue probability map

→ Why Bayesian? Provide a better and stable estimate.

Bayesian Framework

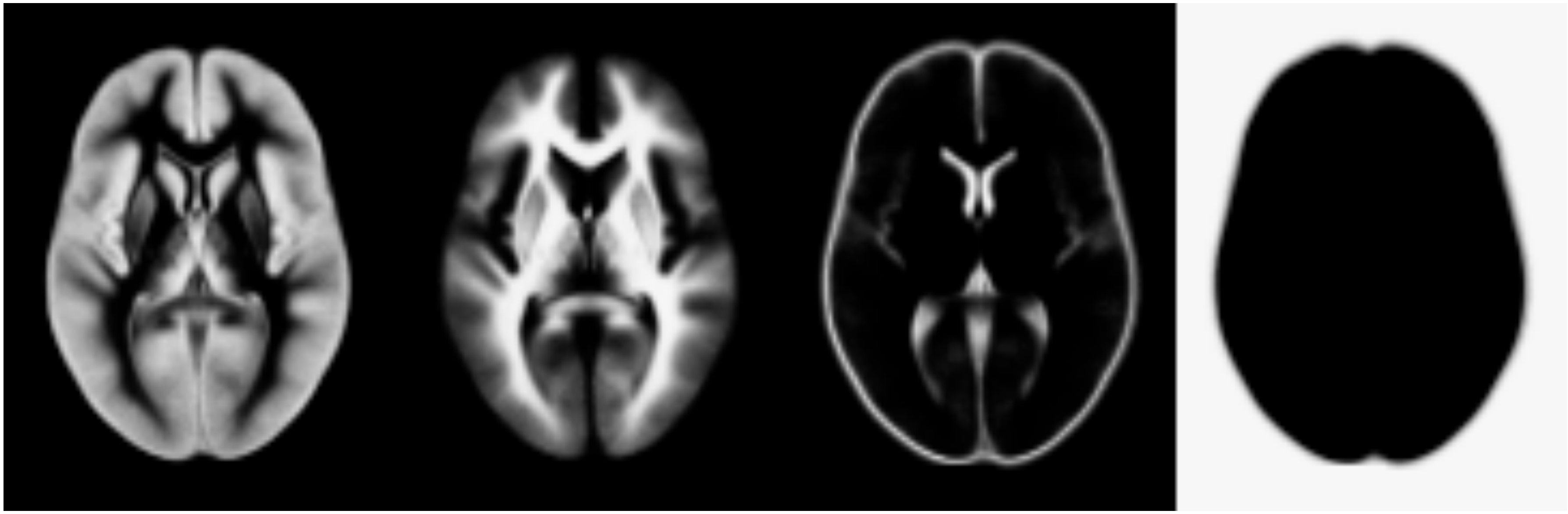
- Posterior probability can be obtained from a prior probability

$$P(class \mid \text{intensity}) = \frac{P(\text{intensity} \mid class)P(class)}{\sum_{class} P(\text{intensity} \mid class)P(class)}$$

Likelihood Prior map

: the probability obtaining image intensity given class.
This can be obtained from our Gaussian mixture model

Prior tissue probability maps



ICBM Tissue Probabilistic Atlases
C. Phillips

Details on Bayesian Framework

Let T be the event that a voxel has a particular image intensity value. This is that we usually observe in T_1-weighted MRI. What we want is the *conditional probability* $P(C|T)$ of the voxel belong to the class C given that we have observed T:

$$P(C|T) = \frac{P(C \cap T)}{P(T)}. \tag{3.1}$$

$P(C|T)$ is interpreted as the probability of the voxel belong to a specific class when it has a particular intensity value. This is what we likely to determine in probabilistic segmentation. The numerator can be written as $P(C \cap T) = P(T|C)P(C)$ while, from the law of total probability, the probability $P(T)$ is given by

$$P(T) = \sum_C P(T \cap C) = \sum_C P(T|C)P(C).$$

Then the conditional probability (3.1) can be written in terms of the prior probability as

$$P(C|T) = \frac{P(T|C)P(C)}{\sum_C P(T|C)P(C)}.$$

The likelihood term $P(T|C)$ is interpreted as the probability of a voxel given the voxel belong to a particular tissue type and obtained from Gaussian mixture modeling.

K-components Mixture

Mixture models have been widely used for segmenting brain images. The k-components mixture model assume the underlying distribution f of data to follow the mixture distribution of the form

$$\text{likelihood term} \longleftarrow \quad f(y|\Theta) = \sum_{i=1}^{k} p_i f_i(y),$$

where f_i is some distribution and the positive mixing proportions p_i add up to one, i.e. $\sum_{i=1}^{k} p_i = 1$. Such distribution can be obtained by conditioning on a multinomial distribution with parameters $(p_1, \cdots, p_k)$. To see this, let $(X_1, \cdots, X_k)$ be a multinomial distribution with parameters $(p_1, \cdots, p_k)$. We further assume $X_1 + \cdots + X_k = 1$. The probability mass function of X is given by

$$f(x_1, \cdots, x_k) = p_1^{x_1} \cdots p_k^{x_k}.$$

Note that if $X_j = 0$ then all other components $X_i = 0$ for $i \neq j$, and subsequently $P(X_j) = p_j$. Then we define the random variable Y conditionally on the event $X_j = 1$ such that $Y \sim f_j$ if $X_j = 1$. This defines the conditional density $f(y|x)$. The joint density $f(x, y)$ is then $f(x_j = 1, y) = p_j f_j(y)$. This can be compactly written as

$$f(x, y) = [p_1 f_1(y)]^{x_1} \cdots [p_k f_k(y)]^{x_k}.$$

The marginal density of Y is trivially then

$$f(y) = \sum_x f(x, y) = \sum_{i=1}^{k} p_i f_i(y).$$

Therefore, the k-components mixture model can be obtained by mixing samples obtained from each distribution f_j with exactly p_j proportion.

For the k-components model, there are $3k-1$ unknown parameters to be estimated. The most widely used methods in parameter estimation is the *maximum likelihood estimation* (MLE). Suppose we have a sample $Y = \{Y_1, \cdots, Y_n\}$ drawn from the distribution $f(y|\Theta)$. The likelihood estimation of Θ is given by maximizing the loglikelihood

$$\widehat{\Theta} = \arg\max_{\Theta} \prod_{i=1}^{n} f(y_i|\Theta) = \arg\max_{\Theta} \sum_{i=1}^{n} \ln f(y_i|\Theta).$$

For the mixture model, the optimization cannot be done analytically and requires a iterative numerical technique called the expectation maximization (EM) algorithm.

Heuristic idea of EM algorithm

- Expectation maximization (EM) algorithm: iterative method for maximizing difficult likelihood functions.

- Instead of maximizing the difficult likelihood directly, we maximize easier likelihood by introducing *latent variables.*

3.3 Expectation Maximization Algorithm

The expectation maximization (EM) algorithm was first introduced by Dempster *et al.* (J. Roy. Statist. Soc. 1997). Read Robert and Casella's Monte Carlo Statistical Methods for the introduction to EM. Flury's A First Course in Multivariate Statistics for the detailed discussion on EM applied to Gaussian mixture model. See Little and Rubin (1987) and McLachlan and Krishnan (1997). EM is widely used in image segmentation. The algorithm proceeds as follows.

Following the notation of Cesella, we augment the observed data Y with with *latent* (unobserved or missing) data Y^m such that the complete data $Y^c = (Y, Y^m)$. The density of the complete data Y^c is denoted as

$$Y^c = (Y, Y^m) \sim f(y^c) = f(y, y^m).$$

The conditional density for the missing data Y^m, condition on observation Y, is

$$f(y^m|y,\Theta) = \frac{f(y,y^m|\Theta)}{f(y|\Theta)}.$$

Taking the logarithm on both sides, we get the loglikelihood for the observed data

$$\ln f(Y|\Theta) = \ln f(Y^c|\Theta) - \ln f(Y^m|Y,\Theta).$$

Now taking the expectation with respect to $f(y^m|y,\Theta_0)$ for some fixed Θ_0 on the both sides, we have

$$\begin{aligned}\mathbb{E}[\ln f(Y|\Theta)|Y,\Theta_0] &= \mathbb{E}[\ln f(Y^c|\Theta)|Y,\Theta_0] && (3.2)\\ &\quad -\mathbb{E}[\ln f(Y^m|Y,\Theta)|Y,\Theta_0]. && (3.3)\end{aligned}$$

Now denote the expected loglikelihood for the complete data as

$$Q(\Theta|\Theta_0, Y) = \mathbb{E}[\ln f(Y^c|\Theta)|Y,\Theta_0].$$

We maximize the likelihood in the following fashion

- E-step: compute $Q(\Theta|\widehat{\Theta}_{j-1}, Y)$.

- M-step: maximize $Q(\Theta|\widehat{\Theta}_{j-1}, Y)$ and take

$$\widehat{\Theta}_j = \arg\max_{\Theta} Q(\Theta|\widehat{\Theta}_{j-1}, Y). \tag{3.4}$$

Bounded monotonically increasing function → converges

Application: Two-components Gaussian mixture model

$$f(y) = pf_1(y) + (1 - p)f_2(y)$$

$$f_1(y) \approx N(\mu_1, {\sigma_1}^2)$$

$$f_2(y) \approx N(\mu_2, {\sigma_2}^2)$$

p = mixing proportion → estimated tissue density

Parameters are estimated by the EM-algorithm

Maximum likelihood function

$$L = \prod_{i=1}^{n} \left[pf_1(y_i) + (1-p) f_2(y_i) \right]$$

$$\log L = \sum_{i=1}^{n} \log \left[pf_1(y_i) + (1-p) f_2(y_i) \right]$$

Solve $\frac{\partial \log L}{\partial p} = 0$ numerically

Let’s avoid brute-force numerical optimization

EM algorithm

Observed variable Y
Missing variable X: Bernoulli(p)
Complete variable Z=(X,Y)

The joint density of Z

$$f(x,y) = [pf_1(y)]^x[qf_2(y)]^{1-x}$$

Conditional density of X given Y

$$f(x|y) = \frac{[pf_1(y)]^x[qf_2(y)]^{1-x}}{pf_1(y) + qf_2(y)}$$

EM algorithm

Conditional expectation

$$\mathbb{E}(X|y,p) = \frac{pf_1(y)}{pf_1(y) + qf_2(y)}$$

E-step: Construction of Q-function:

$$Q(p) = E\left[\log L(X,Y) \mid Y\right]$$

$$= \sum_{i=1}^{n} \frac{p_0 f_1(y_i)}{p_0 f_1(y_i) + q_0 f_2(y_i)} \log \frac{p}{1-p}$$
$$+ n \log(1-p).$$

EM-algorithm

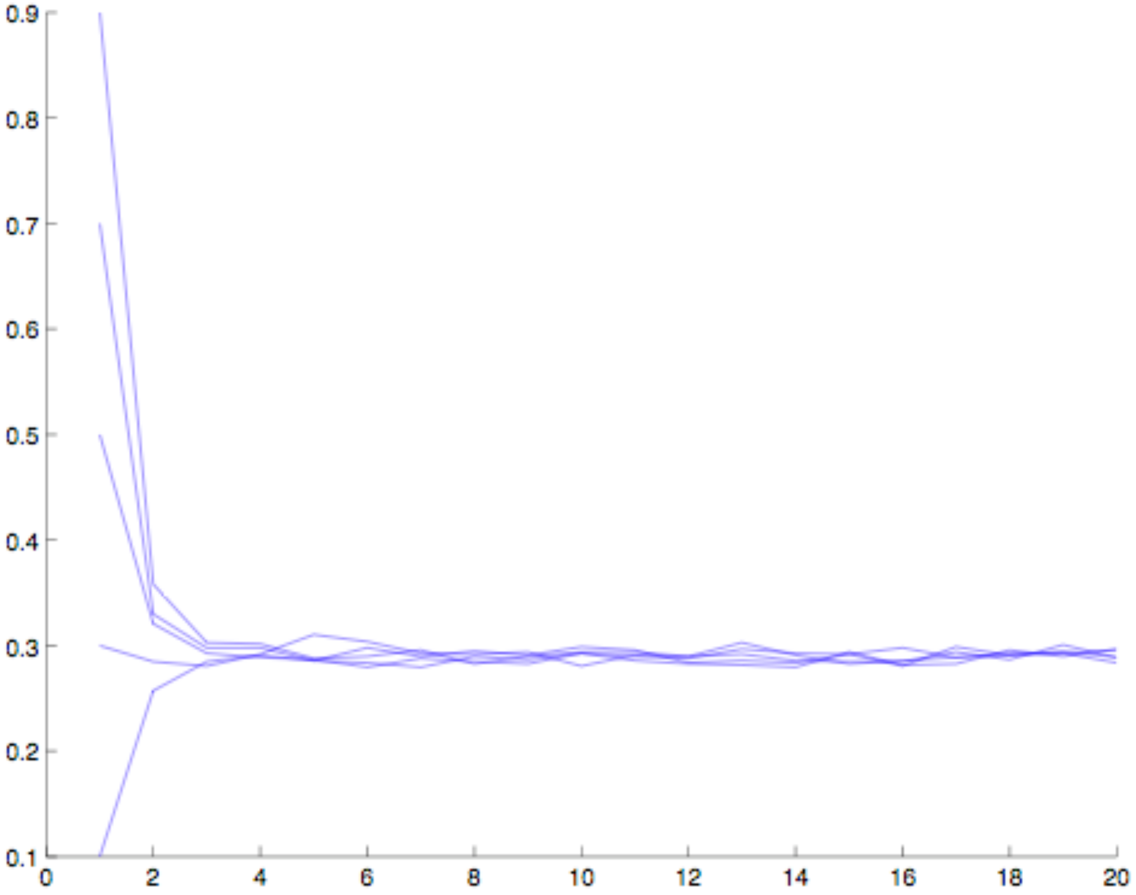
- M-step: $\frac{\partial Q}{\partial p} = 0$

$$p = \frac{1}{n} \sum_{i=1}^{n} \frac{p_0 f_1(y_i)}{p_0 f_1(y_i) + q_0 f_2(y_i)}$$

Iteration: pick any initial value between 0 and 1

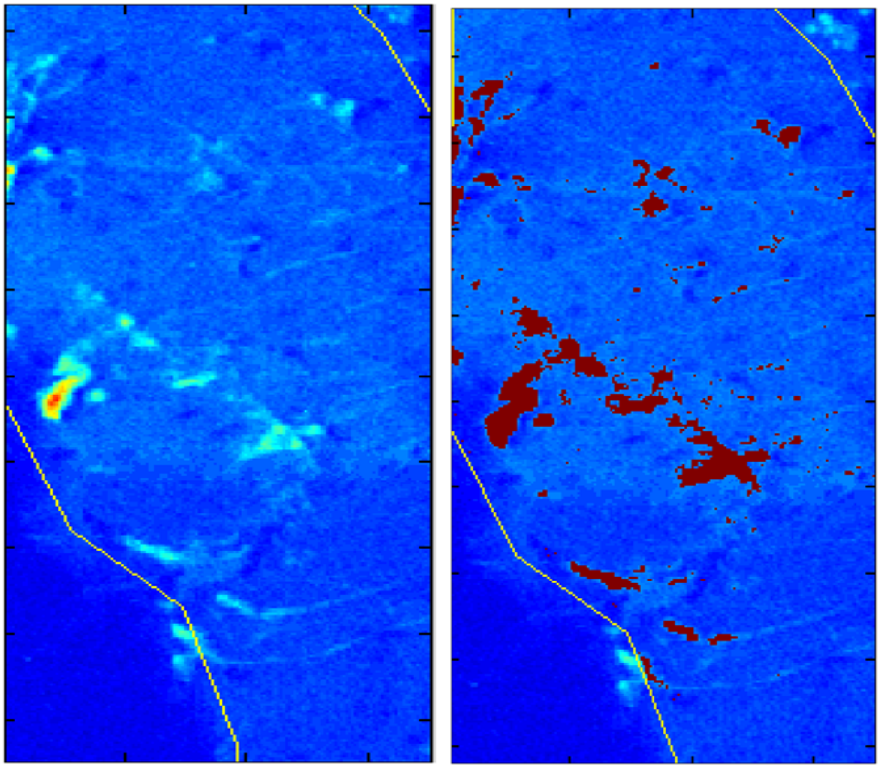
$$\hat{p}_{j+1} = \frac{1}{n} \sum_{i=1}^{n} \frac{\hat{p}_j f_1(y_i)}{\hat{p}_j f_1(y_i) + (1 - \hat{p}_j) f_2(y_i)}$$

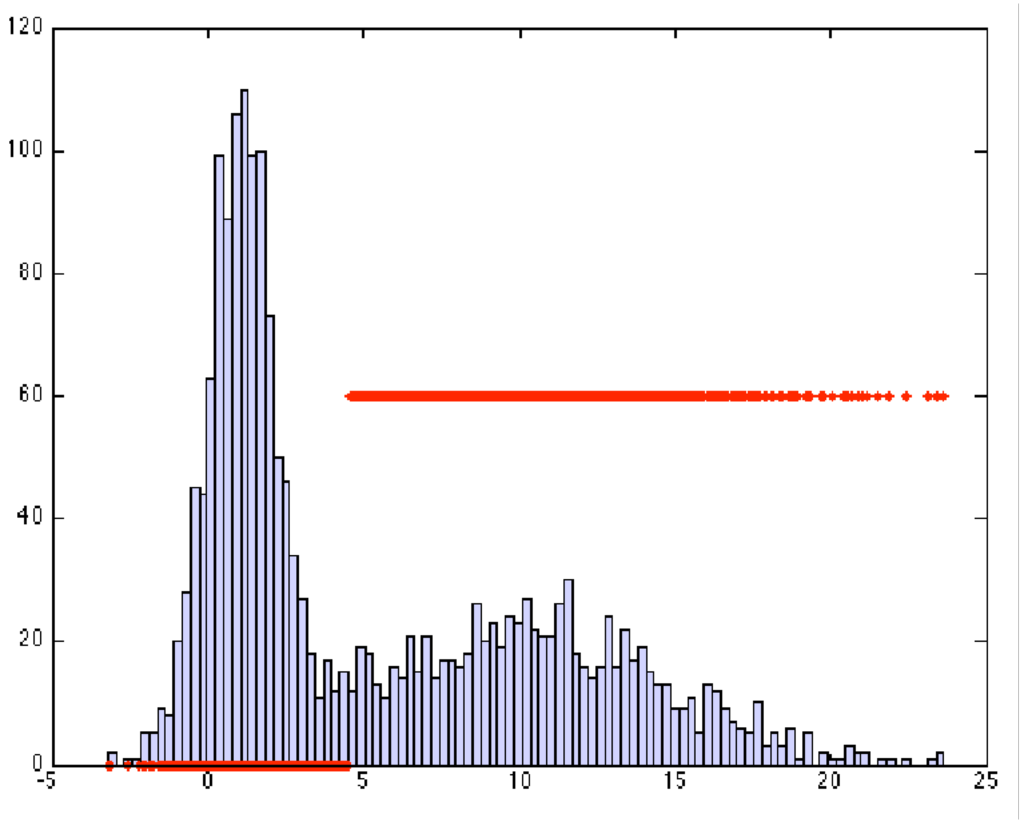
Convergence of EM algorithm



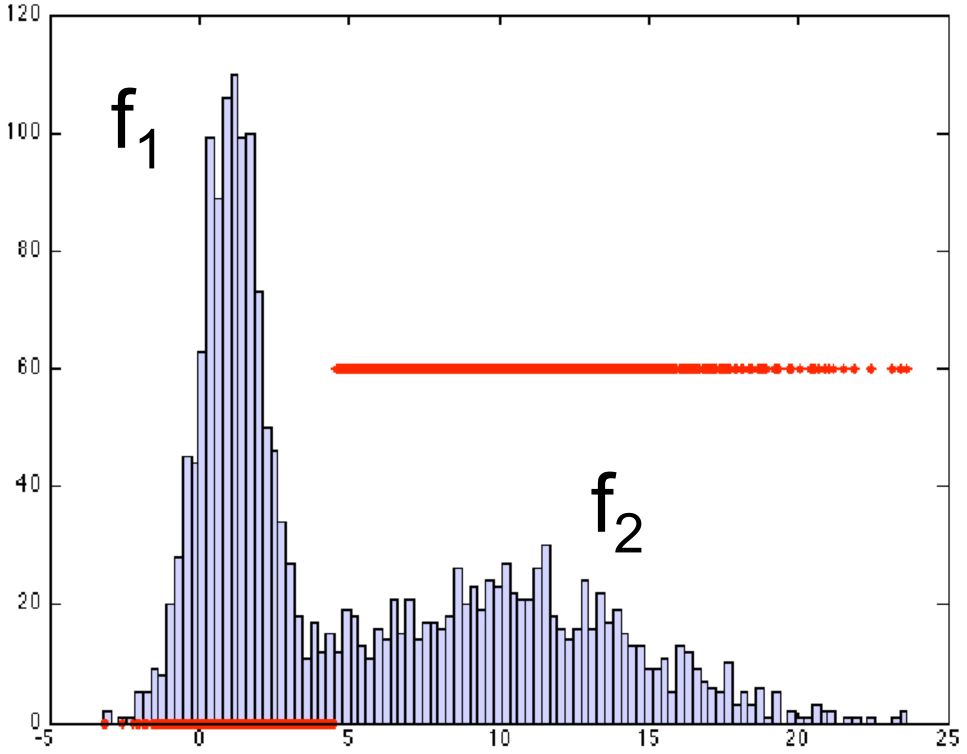
Gaussian mixture model

The pixel intensity values f (integers between 0-255) are modeled as the two component Gaussian mixtures $f = \pi_1 f_1 + \pi_2 f_2$, where π_1 and π_2 are the unknown mixing proportion with $\pi_1 + \pi_2 = 1$, and f_1 and f_2 are independent Gaussian random variables. Each f_i is a Gaussian random variable $N(\mu_i, \sigma_i^2)$ with the unknown mean μ_i and the standard deviation σ_i.





Various decision rules:



Simplistic segmentation
For each voxel intensity value y, check if $f_1(y) > f_2(y)$
→Hard assignment

More complicated one
Mixing proportion
→Bayesian segmentation

MATLAB demonstration
Gaussian mixture & EM algorithm

Application of probabilistic segmentation:

Voxel-based morphometry (VBM)

- A new approach (Ashburner & Friston, 2000)

Not really new anymore

- No ROI segmentation required.
- Anatomical difference is characterized at each voxel.

Introduction to VBM

- Fully automated image analysis technique allowing identification of regional differences in gray matter (GM) and white matter (WM) between populations without a prior ROI.

- Implemented in the SPM package (Wellcome Department of Cognitive Neurology, London, UK, http://www.fil.ion.ucl.ac.uk/spm)

Advantages of VBM

- It does not require *a priori* knowledge of the ROI to perform the morphological analysis (Davatzikos, 1999; Ashburner and Friston, 2000; Chung et al., 2001).

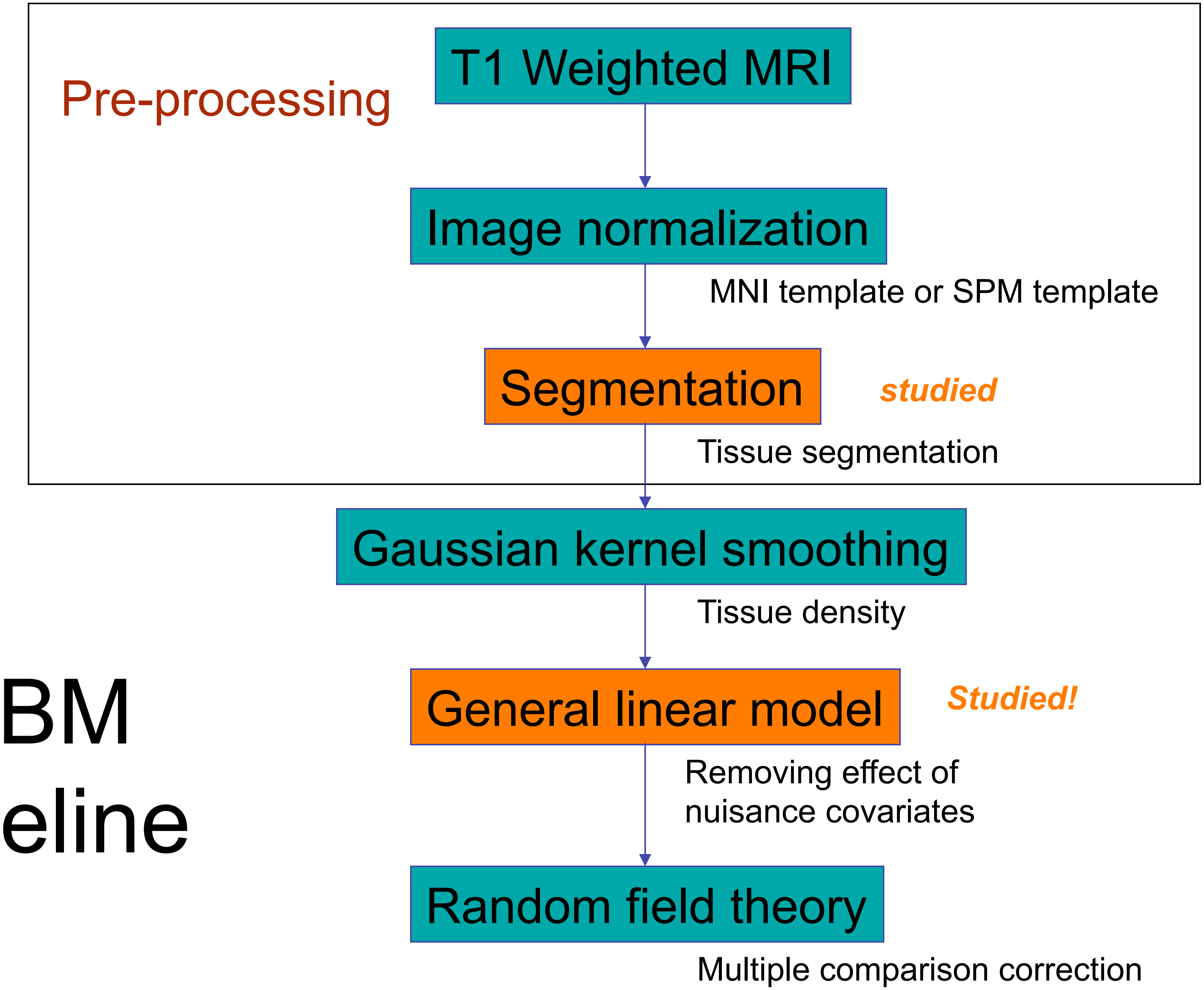
- No need for time consuming either manual or automatic segmentation of ROI.

- Anatomical differences can be detected at a voxel level within ROI itself giving additional localization power that ROI-based approaches lack.

VBM procedures

- Normalize structural MRIs to the standard SPM template
- Segment the normalized images into white and gray matter and cerebrospinal fluid (CSF) based on a Gaussian mixture model (Ashburner and Friston, 1997, 2000).
- The final output: the probability of each voxel belonging to a particular tissue type. This probability is usually referred to as the *gray/ white matter density*.

VBM pipeline



Pre-processing

T1 Weighted MRI

Image normalization

MNI template or SPM template

Segmentation

studied

Tissue segmentation

Gaussian kernel smoothing

Tissue density

General linear model

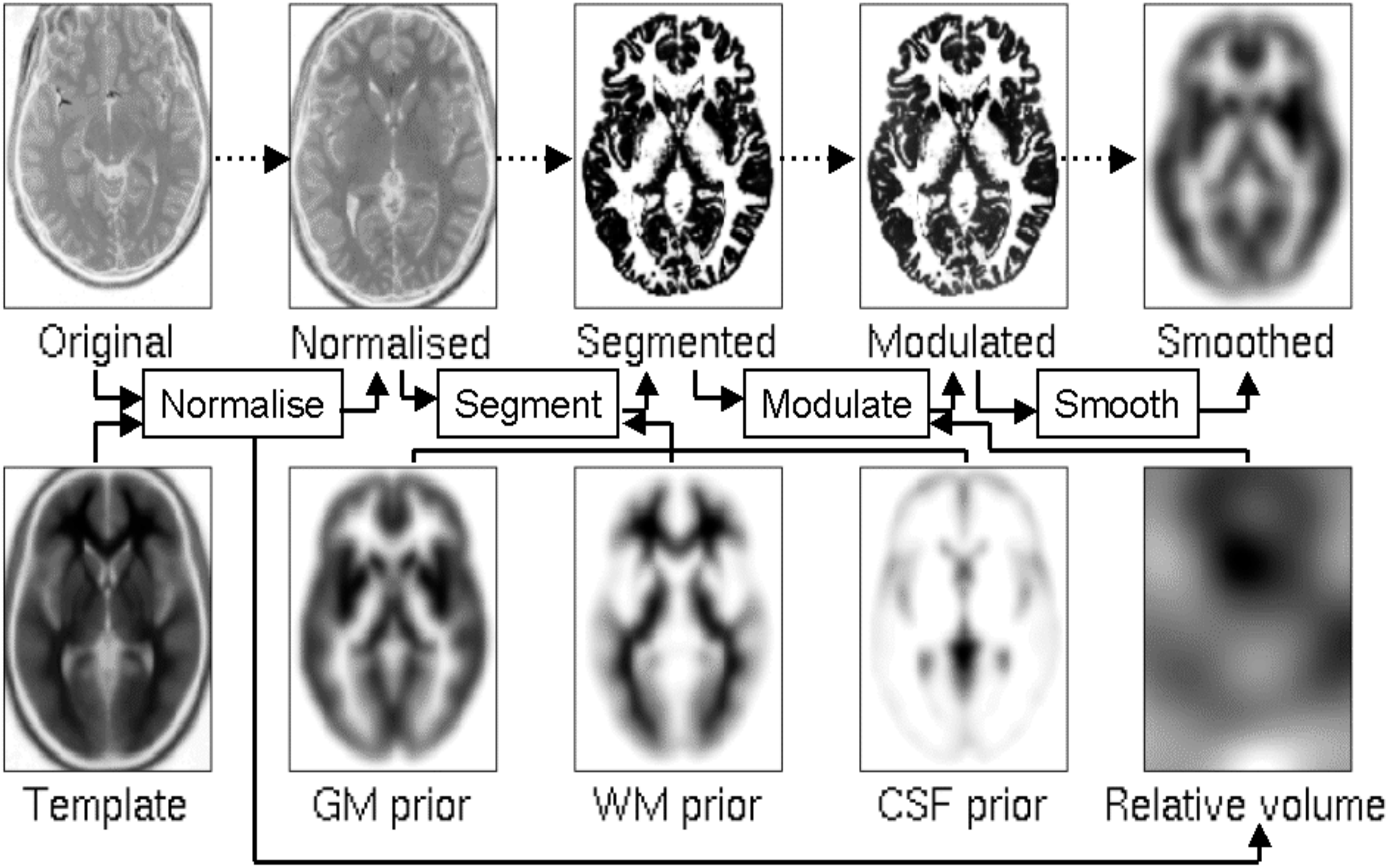
Studied!

Removing effect of nuisance covariates

Random field theory

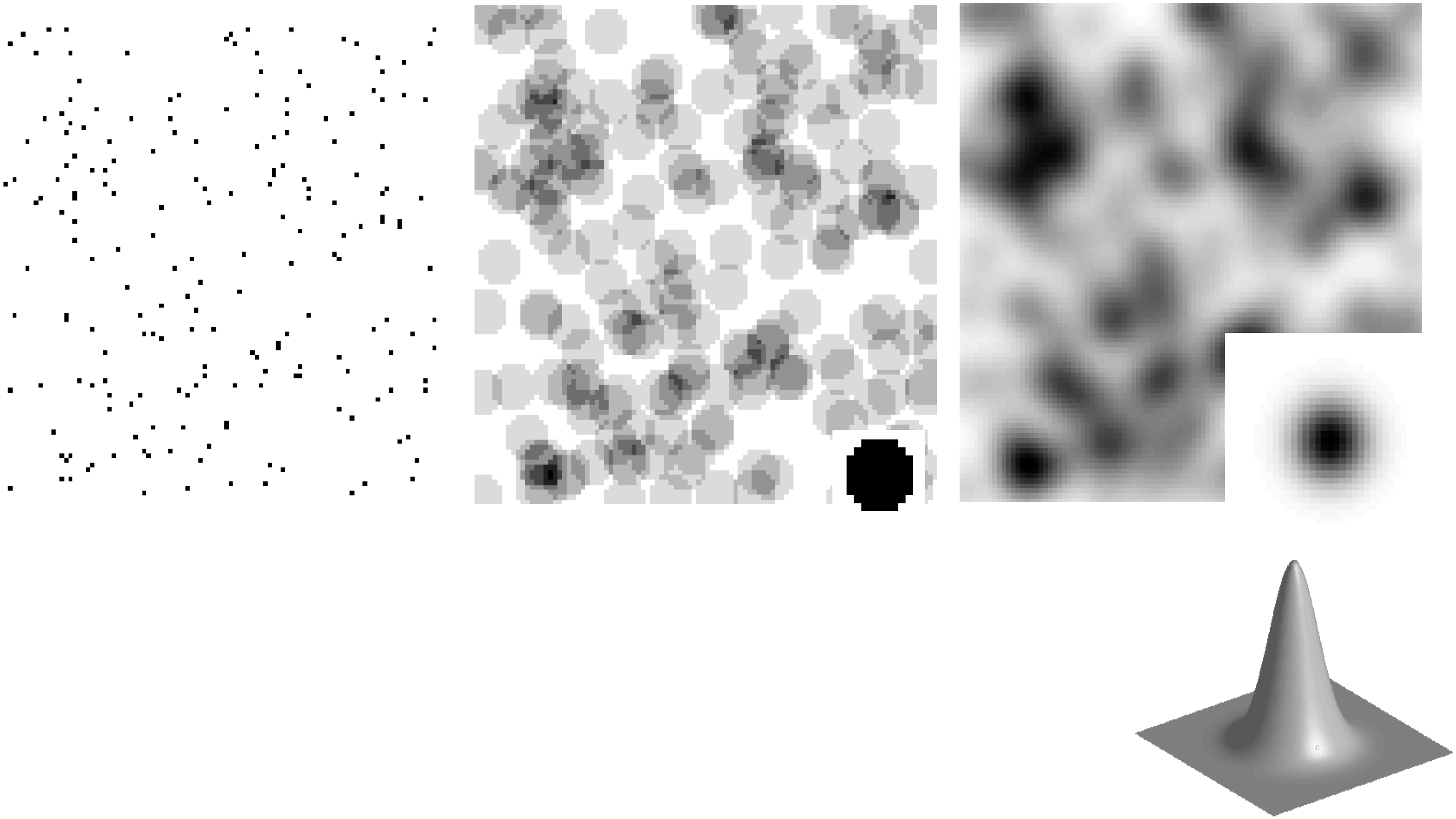
Multiple comparison correction

Pre-processing for VBM



John Ashburner

Smoothing

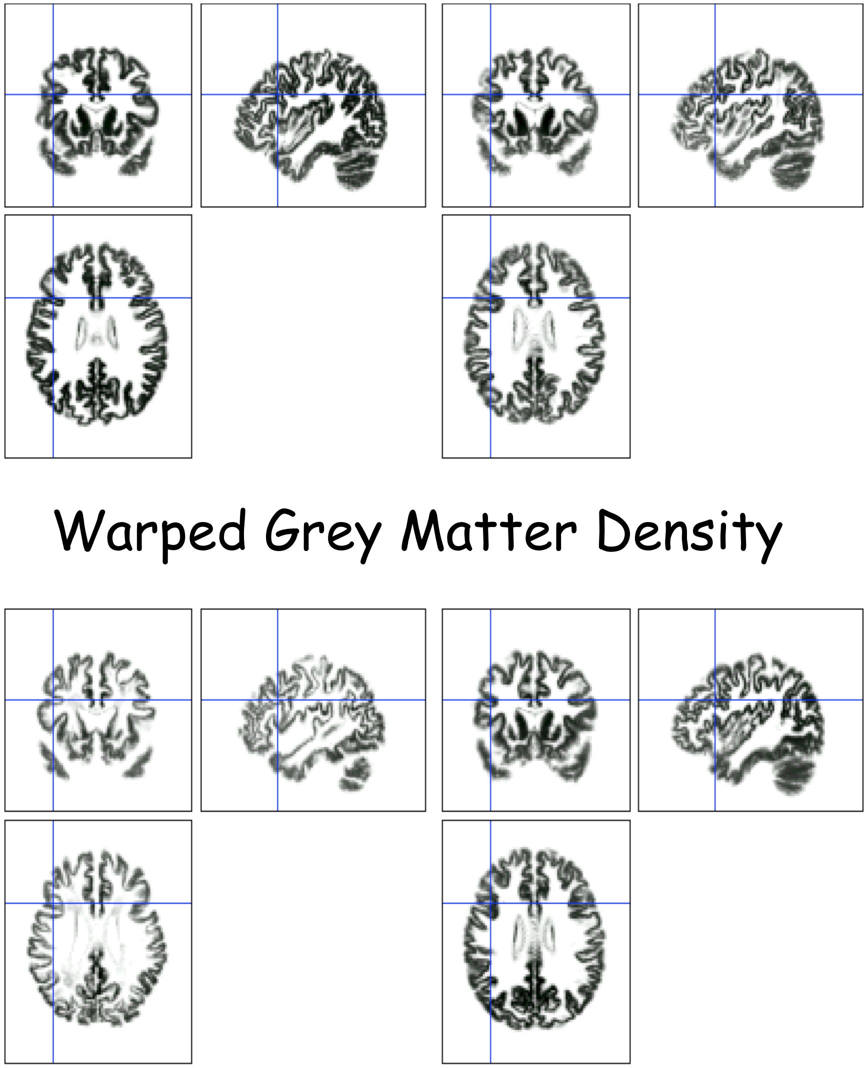


Before convolution

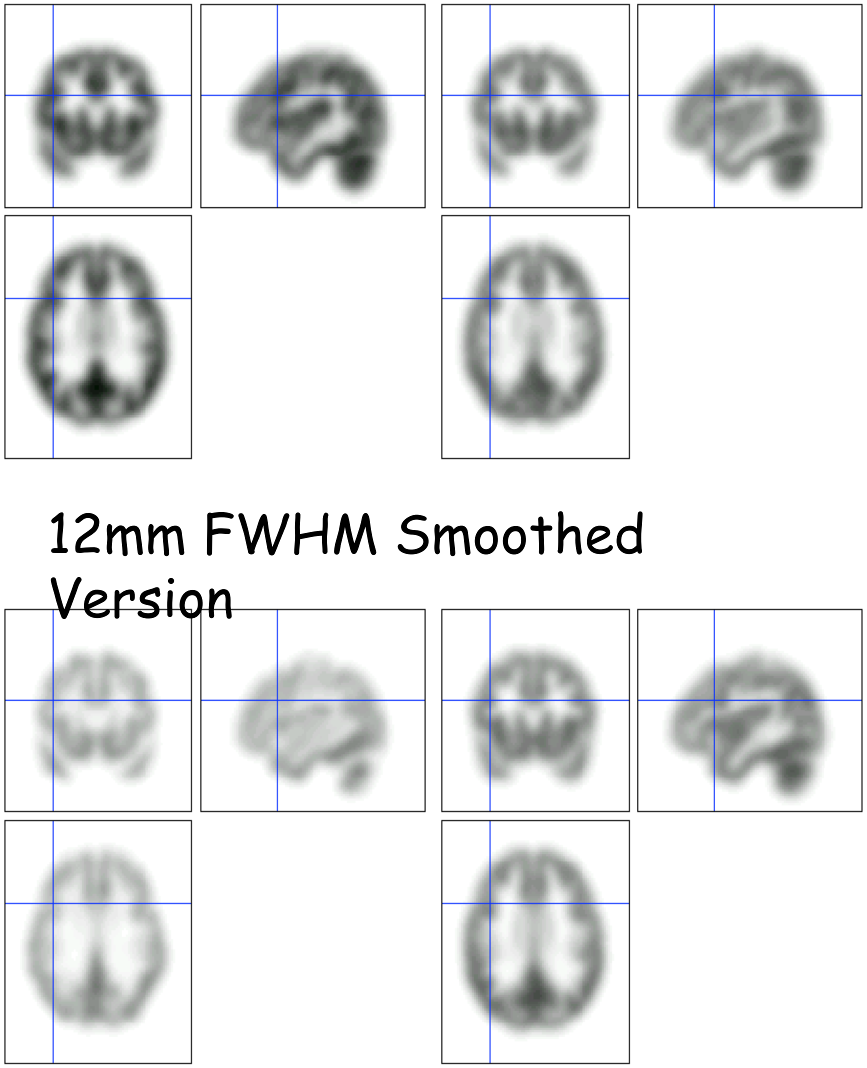
Convolved with a circle

Convolved with a Gaussian

Examples of four subjects

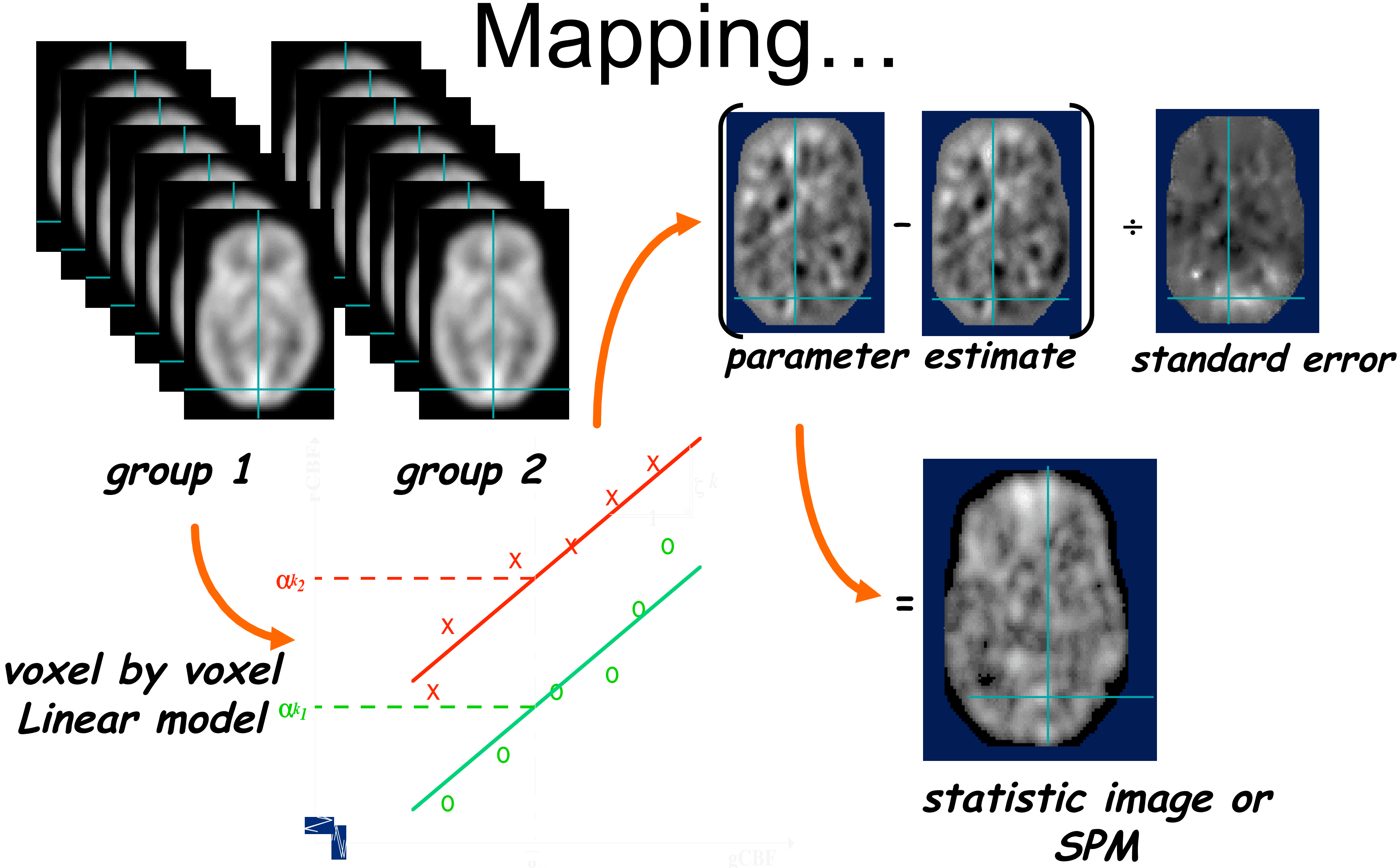


Warped Grey Matter Density



12mm FWHM Smoothed Version

Statistical Parametric Mapping…



group 1

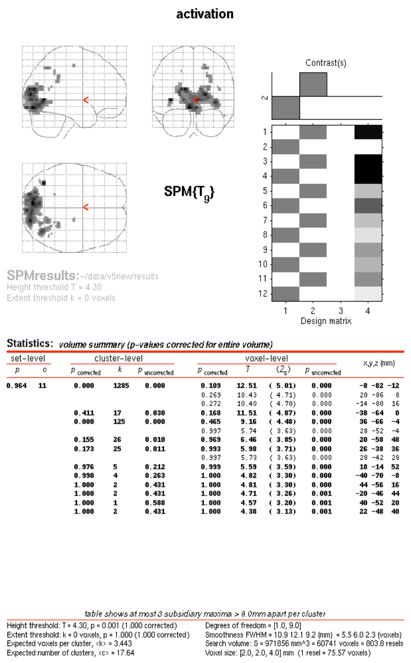
group 2

voxel by voxel
Linear model

[parameter − estimate] ÷ standard error

= statistic image or SPM

SPM results



SPMresults:~/data/v5new/results
Height threshold T = 4.30
Extent threshold k = 0 voxels

Statistics: *volume summary (p-values corrected for entire volume)*

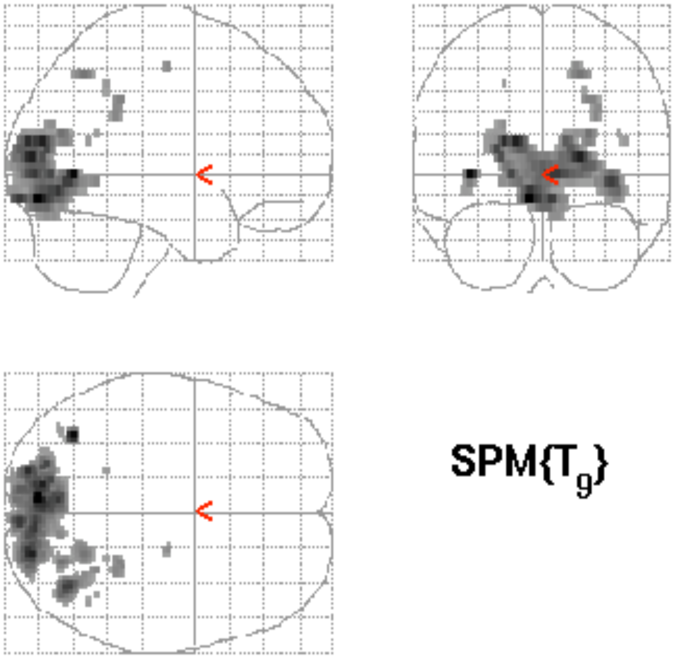
set-level		cluster-level			voxel-level				x,y,z {mm}
p	c	$p_{corrected}$	k	$p_{uncorrected}$	$p_{corrected}$	T	$(Z_{\equiv})$	$p_{uncorrected}$	
0.964	**11**	**0.000**	**1285**	**0.000**	**0.109**	**12.51**	**(5.01)**	**0.000**	**-8 -82 -12**
					0.269	10.43	(4.71)	0.000	20 -86 8
					0.272	10.40	(4.70)	0.000	-14 -80 16
		0.411	**17**	**0.030**	**0.168**	**11.51**	**(4.87)**	**0.000**	**-38 -64 0**
		0.000	**125**	**0.000**	**0.465**	**9.16**	**(4.48)**	**0.000**	**36 -66 -4**
					0.997	5.74	(3.63)	0.000	28 -52 -4
		0.155	**26**	**0.010**	**0.969**	**6.46**	**(3.85)**	**0.000**	**20 -58 48**
		0.173	**25**	**0.011**	**0.993**	**5.98**	**(3.71)**	**0.000**	**26 -38 36**
					0.997	5.73	(3.63)	0.000	28 -42 28
		0.976	**5**	**0.212**	**0.999**	**5.59**	**(3.59)**	**0.000**	**18 -14 52**
		0.990	**4**	**0.263**	**1.000**	**4.82**	**(3.30)**	**0.000**	**-40 -70 -8**
		1.000	**2**	**0.431**	**1.000**	**4.81**	**(3.30)**	**0.000**	**44 -56 16**
		1.000	**2**	**0.431**	**1.000**	**4.71**	**(3.26)**	**0.001**	**-20 -46 44**
		1.000	**1**	**0.588**	**1.000**	**4.57**	**(3.20)**	**0.001**	**40 -52 20**
		1.000	**2**	**0.431**	**1.000**	**4.38**	**(3.13)**	**0.001**	**22 -48 40**

table shows at most 3 subsidiary maxima > 8.0mm apart per cluster

Height threshold: T = 4.30, p = 0.001 (1.000 corrected)
Extent threshold: k = 0 voxels, p = 1.000 (1.000 corrected)
Expected voxels per cluster, <k> = 3.443
Expected number of clusters, <c> = 17.64

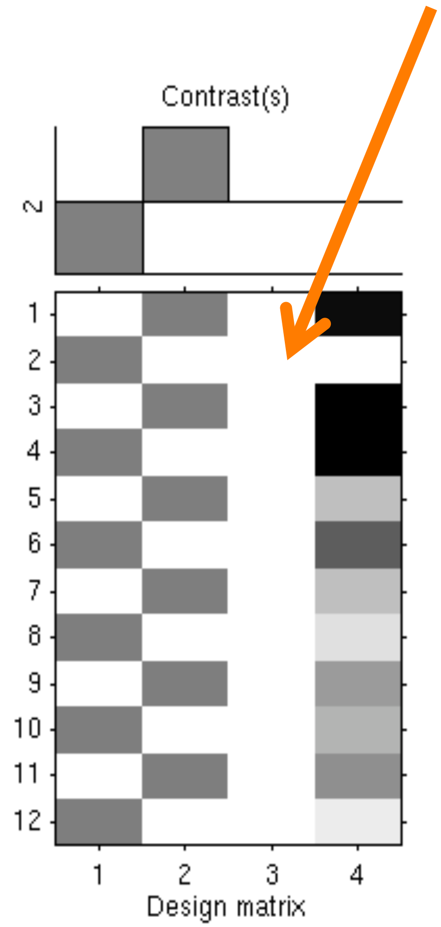
Degrees of freedom = [1.0, 9.0]
Smoothness FWHM = 10.9 12.1 9.2 {mm} = 5.5 6.0 2.3 {voxels}
Search volume: S = 971856 mm^3 = 60741 voxels = 803.8 resels
Voxel size: [2.0, 2.0, 4.0] mm (1 resel = 75.57 voxels)

activation



SPM{T9}

Studied!



SPMresults:~/data/v5new/results
Height threshold T = 4.30
Extent threshold k = 0 voxels

Statistics: *volume summary (p-values corrected for entire volume)*

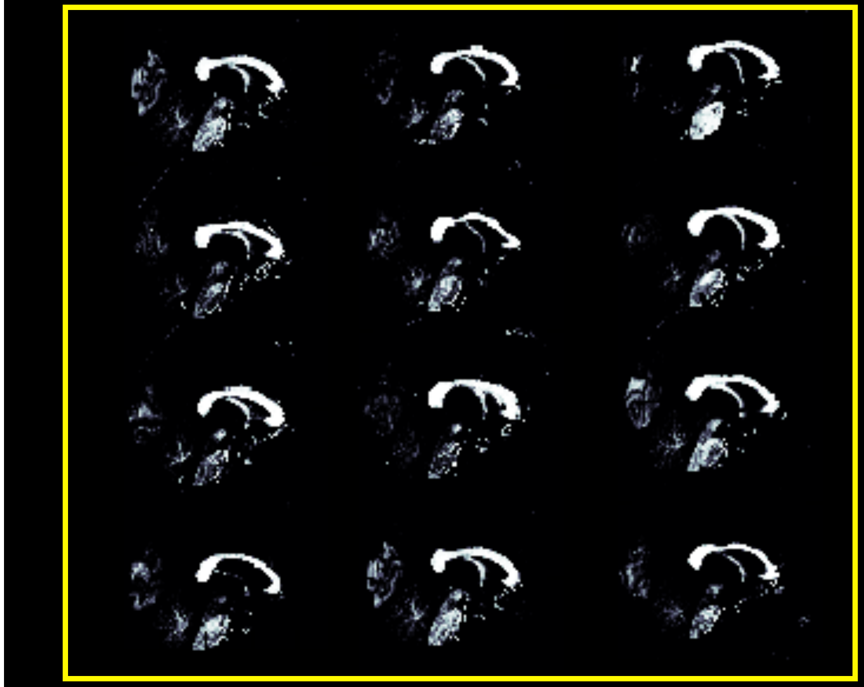
set-level		cluster-level			voxel-level				x,y,z {mm}
p	c	$p_{corrected}$	k	$p_{uncorrected}$	$p_{corrected}$	T	$(Z_{\equiv})$	$p_{uncorrected}$	
0.964	**11**	**0.000**	**1285**	**0.000**	**0.109**	**12.51**	**(5.01)**	**0.000**	**-8 -82 -12**
					0.269	10.43	(4.71)	0.000	20 -86 8
					0.272	10.40	(4.70)	0.000	-14 -80 16
		0.411	**17**	**0.030**	**0.168**	**11.51**	**(4.87)**	**0.000**	**-38 -64 0**
		0.000	**125**	**0.000**	**0.465**	**9.16**	**(4.48)**	**0.000**	**36 -66 -4**
					0.997	5.74	(3.63)	0.000	28 -52 -4

More on GLM and VBM
and Image Smoothing

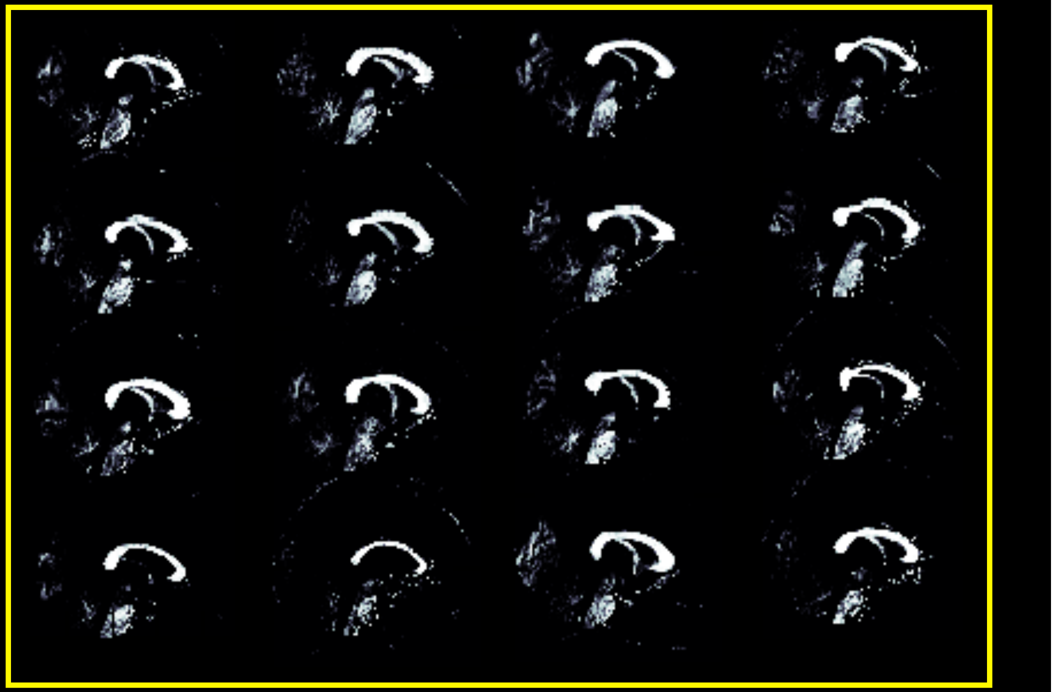
Read ashburner.2000…..
hagler.2006…
Johns.2005…

In relation to the above topics

Real Example: 2D version of VBM

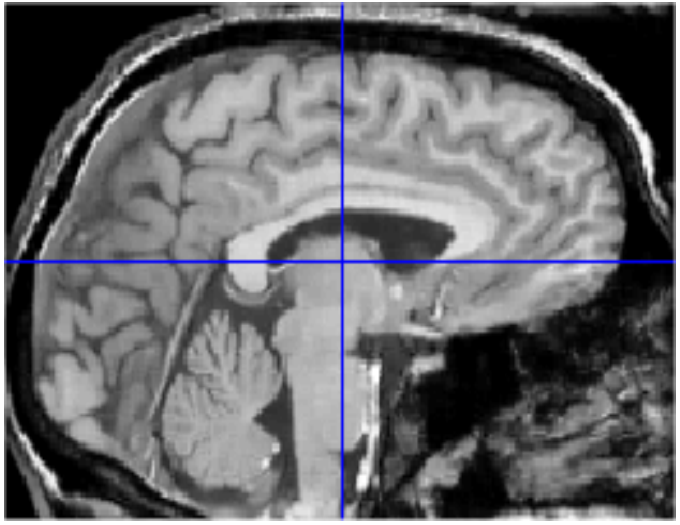


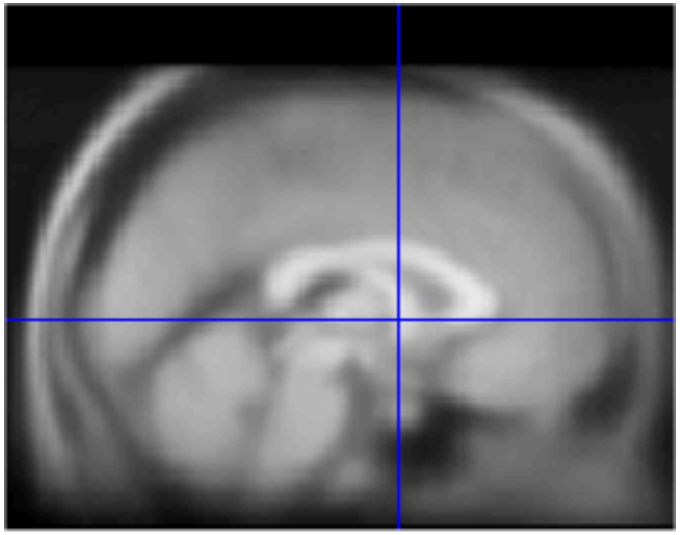
m=12 normal controls



n=16 autistic subjects

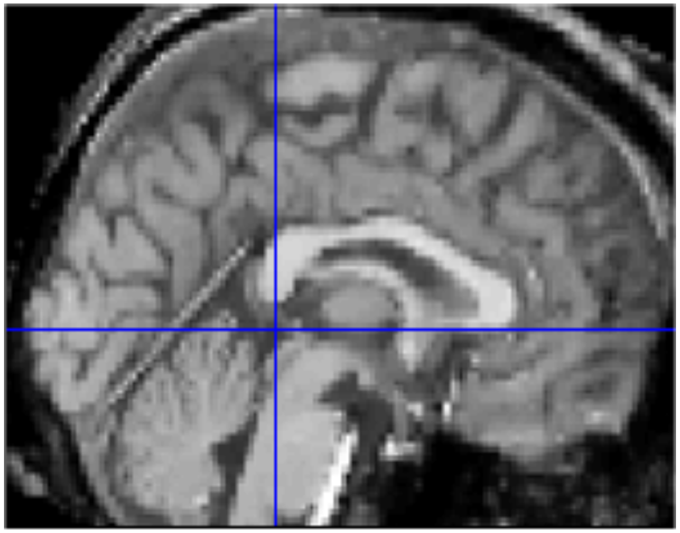
Pre-processing
Nonlinear image registration
(Normalization)



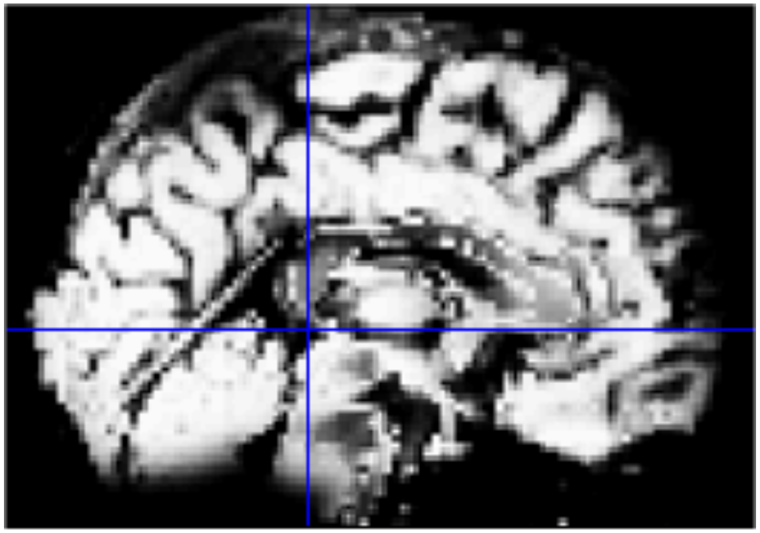


300 subjects MNI template

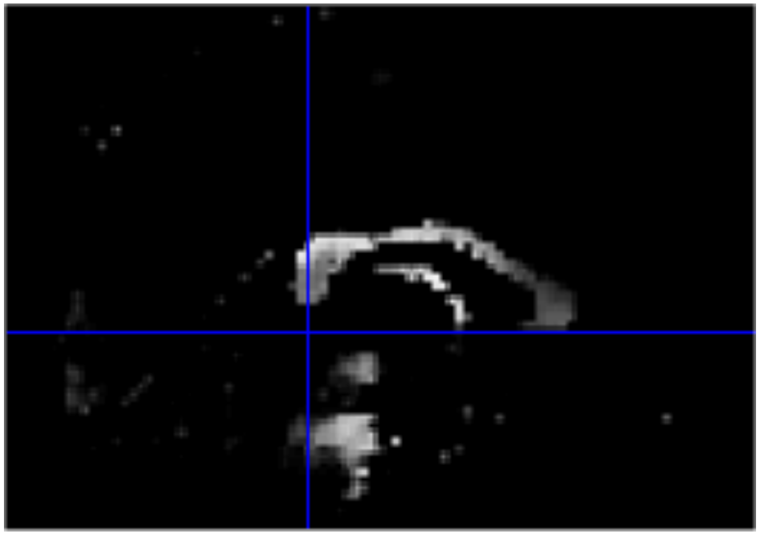
Each subject undergoes this process to reduce positional variability.



White matter segmentation

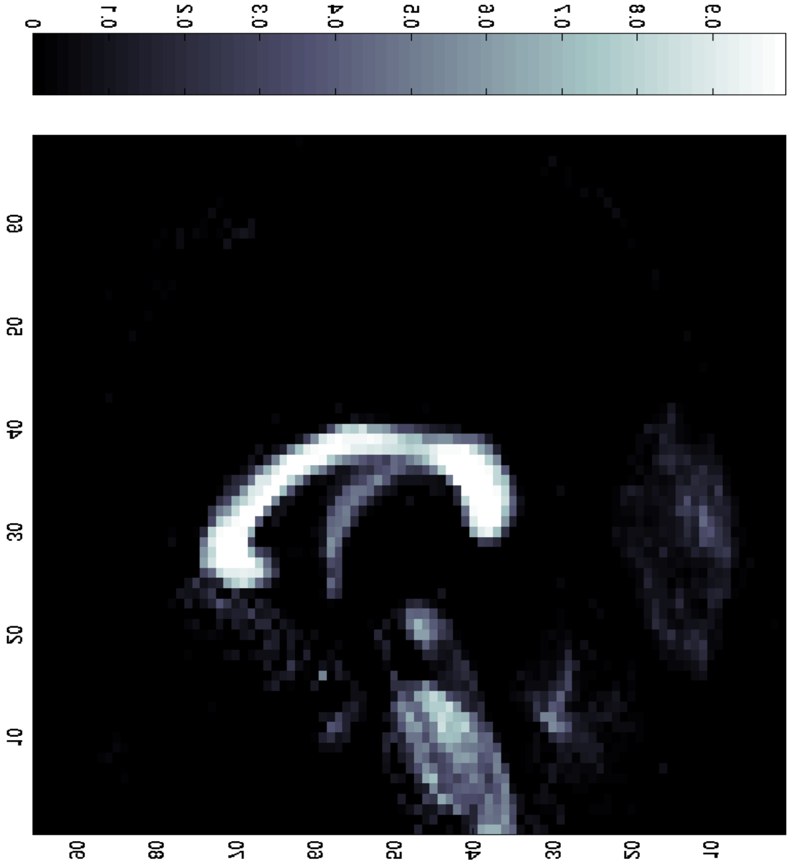


Normalized image

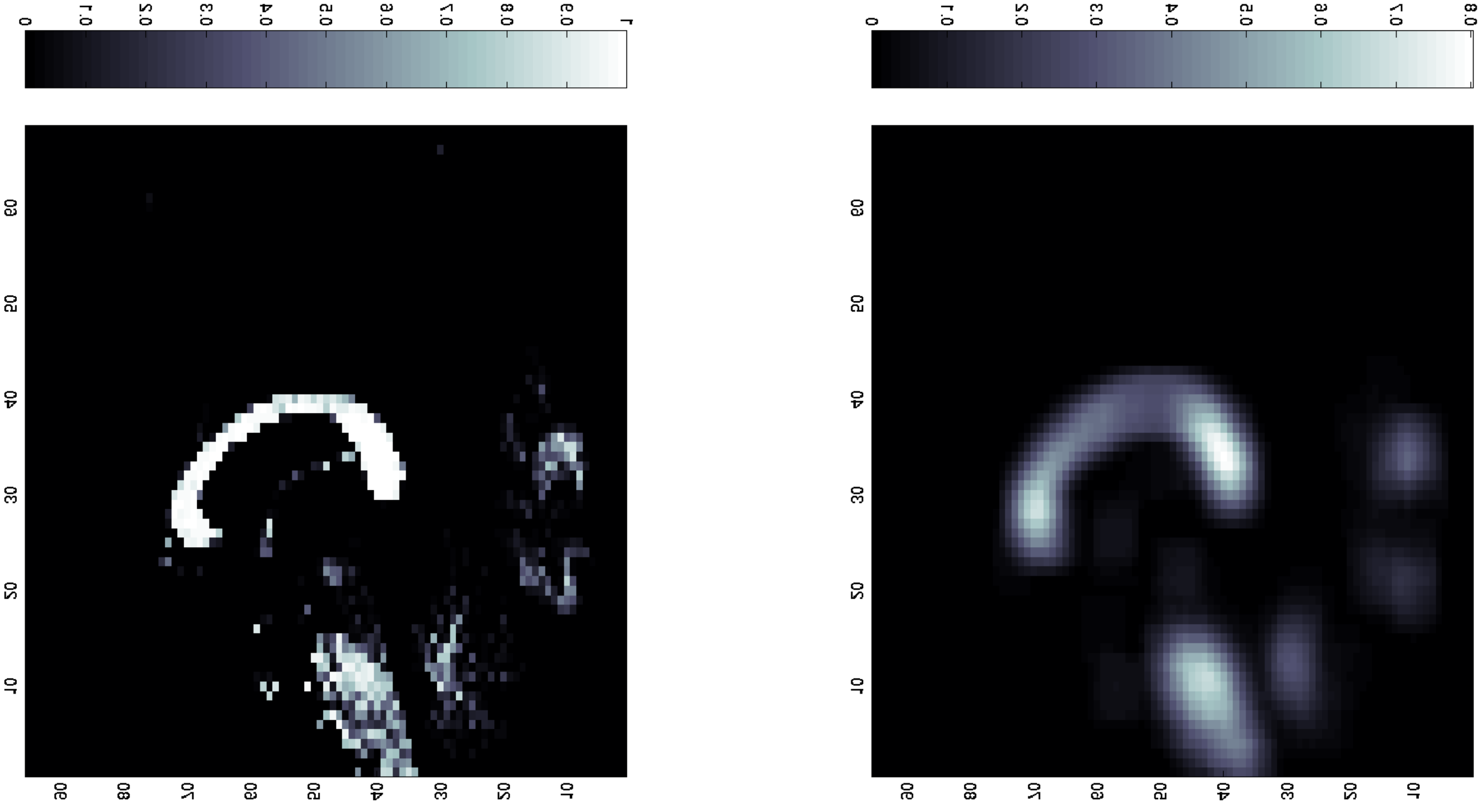


Segmentation of midsagittal corpus callosum region

Mid-sagittal section of brain



Average of 12 normalized mid sagittal segmented images showing well defined corpus callosum. This is our template.



2D Gaussian kernel smoothing on white matter density map before any statistical analysis. Why?

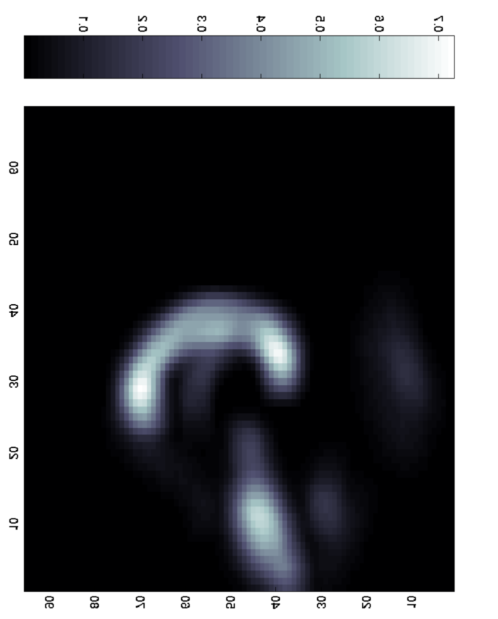
Ashuberner suggested to use logit transform on tissue density

Is *logit* transform necessary?

$$logit(p) : p \rightarrow \frac{1}{2}\log(\frac{p}{1-p})$$

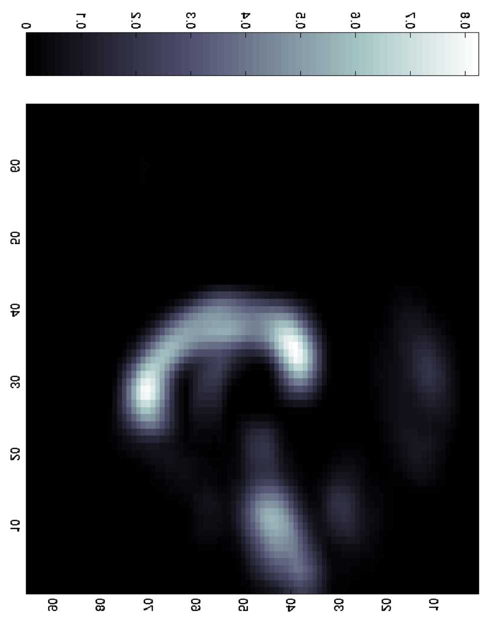
It is not necessary since Gaussian kernel smoothing will make data more Gaussian.

White matter concentration difference



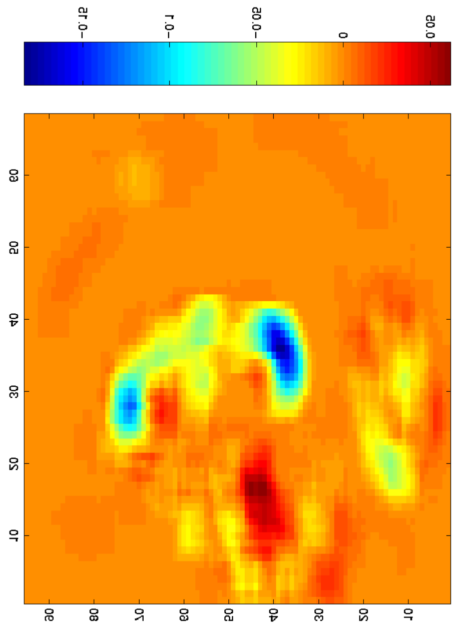
autism

-



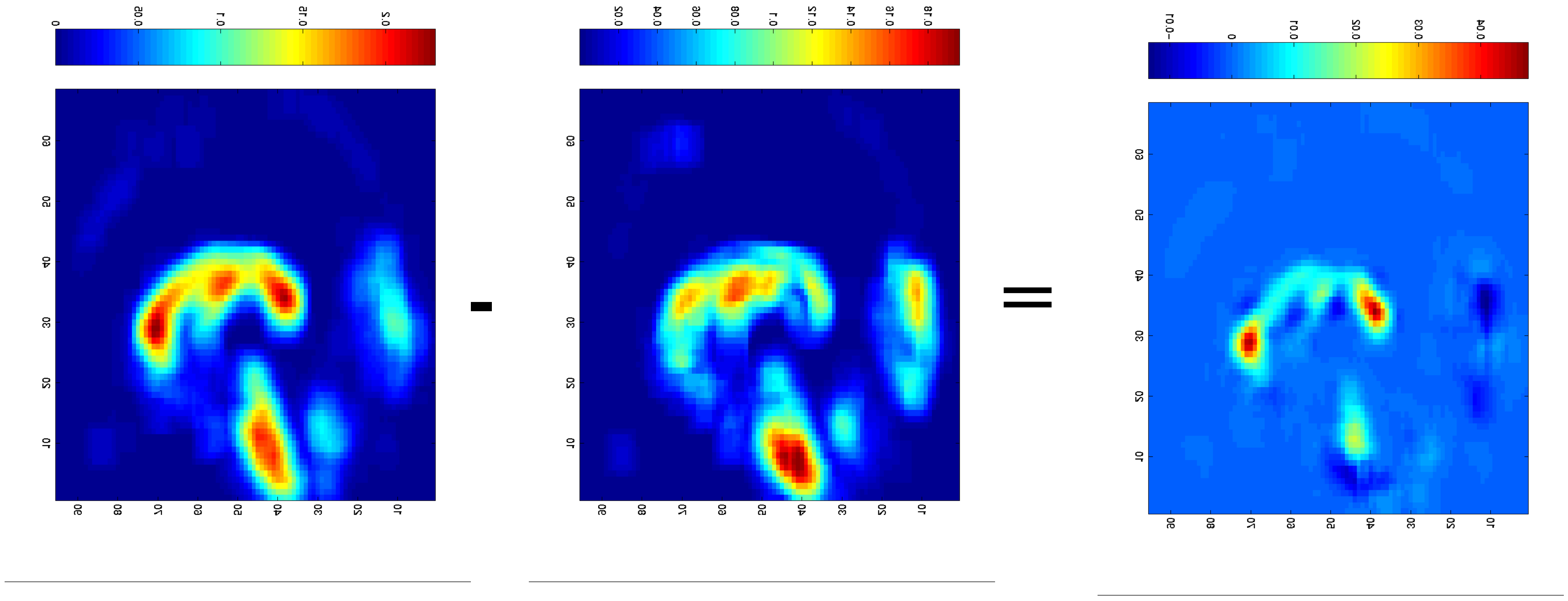
control

=



- Compute the sample mean at each voxel.
- Is the density difference statistically significant ?

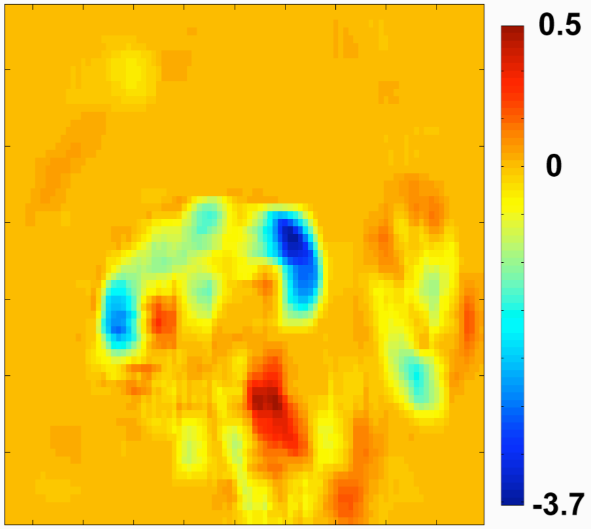
White matter variability difference



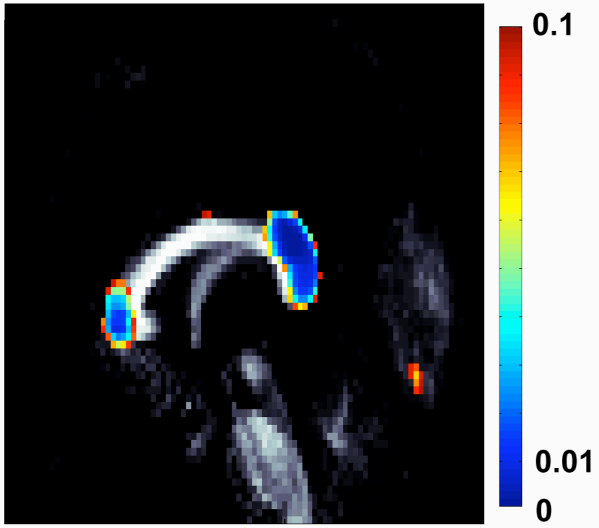
autism

control

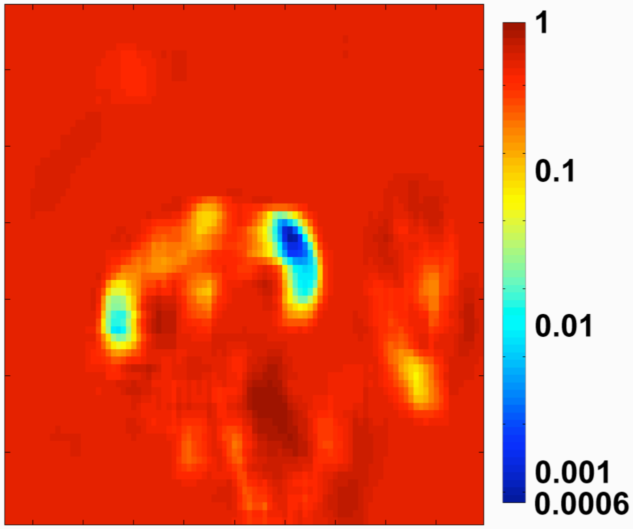
- Compute the sample variance
- Even though the difference shows unequal variance, you used *two sample t-statistic with equal variance assumption*. **Why? It gives us the exact *t random field*.**



t-statistic



P-value



P-value in logscale

But the two sample t test doesn’t seem to be right. There may be possible age effect so it is necessary to remove the age effect. How?

General linear model (GLM)

$$density = \lambda_1 + \lambda_2 age + \beta_1 group + e$$

$$H_0 : \beta_1 = 0 \text{ vs. } H_1 : \beta_1 \neq 0$$

Generalize this model

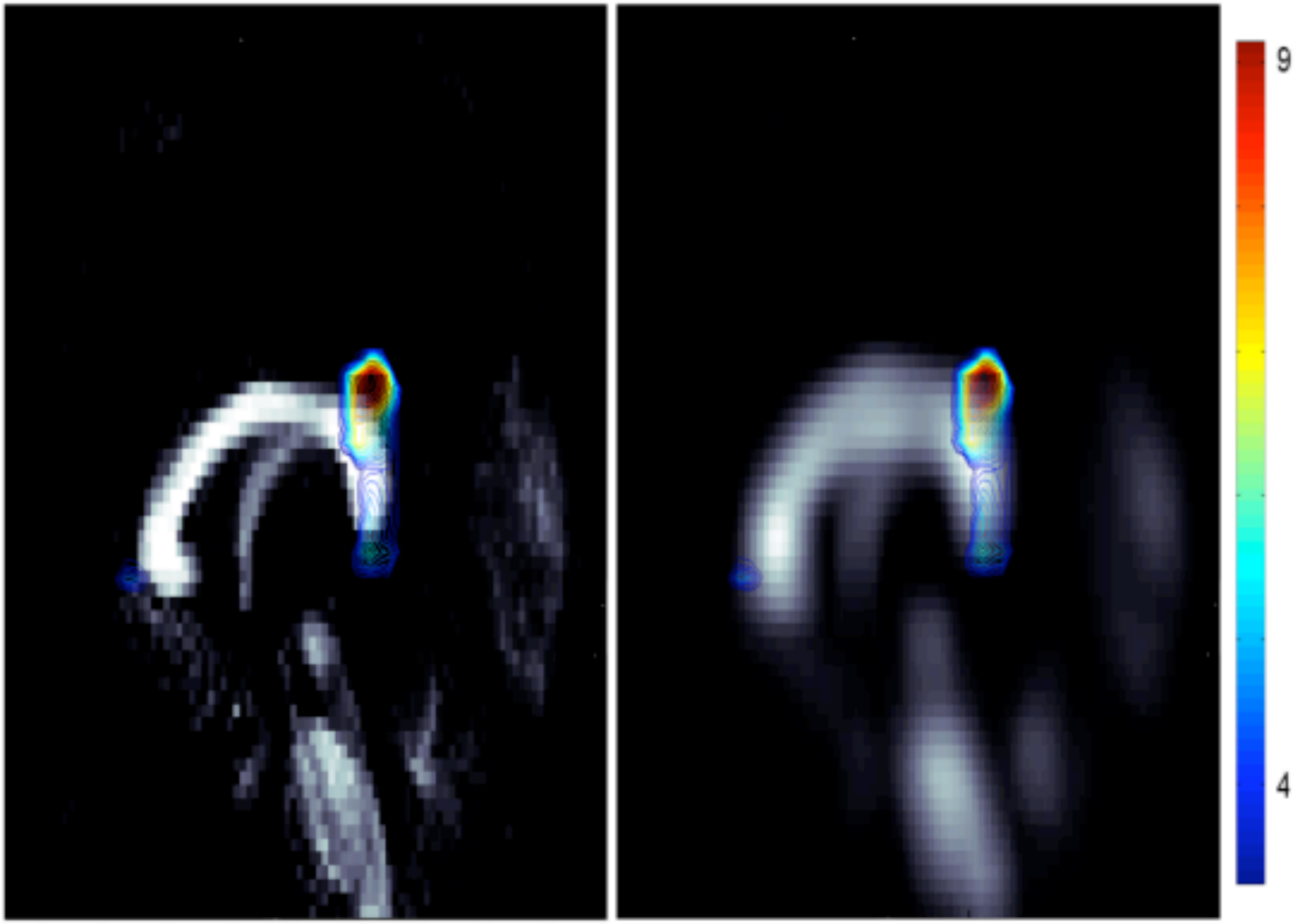
$$Y = \mathbf{z}\lambda + \mathbf{x}\beta + e$$

$\mathbf{x} = (x_1, \cdots, x_p)$ Variables of interest

$\mathbf{z} = (z_1, \cdots, z_k)$ Nuisance covariates

$$F = \frac{(SSE_0 - SSE_1)/p}{SSE_0/(m+n-p-k)} \sim F_{p,m+n-p-k}$$

Computing the sum of squared errors (residuals) requires the least squares estimation (LSE) of unknown parameters.



after removing age effect

Effect of age

Age distribution for autistic subjects:

16.1 (s.d. 4.5)

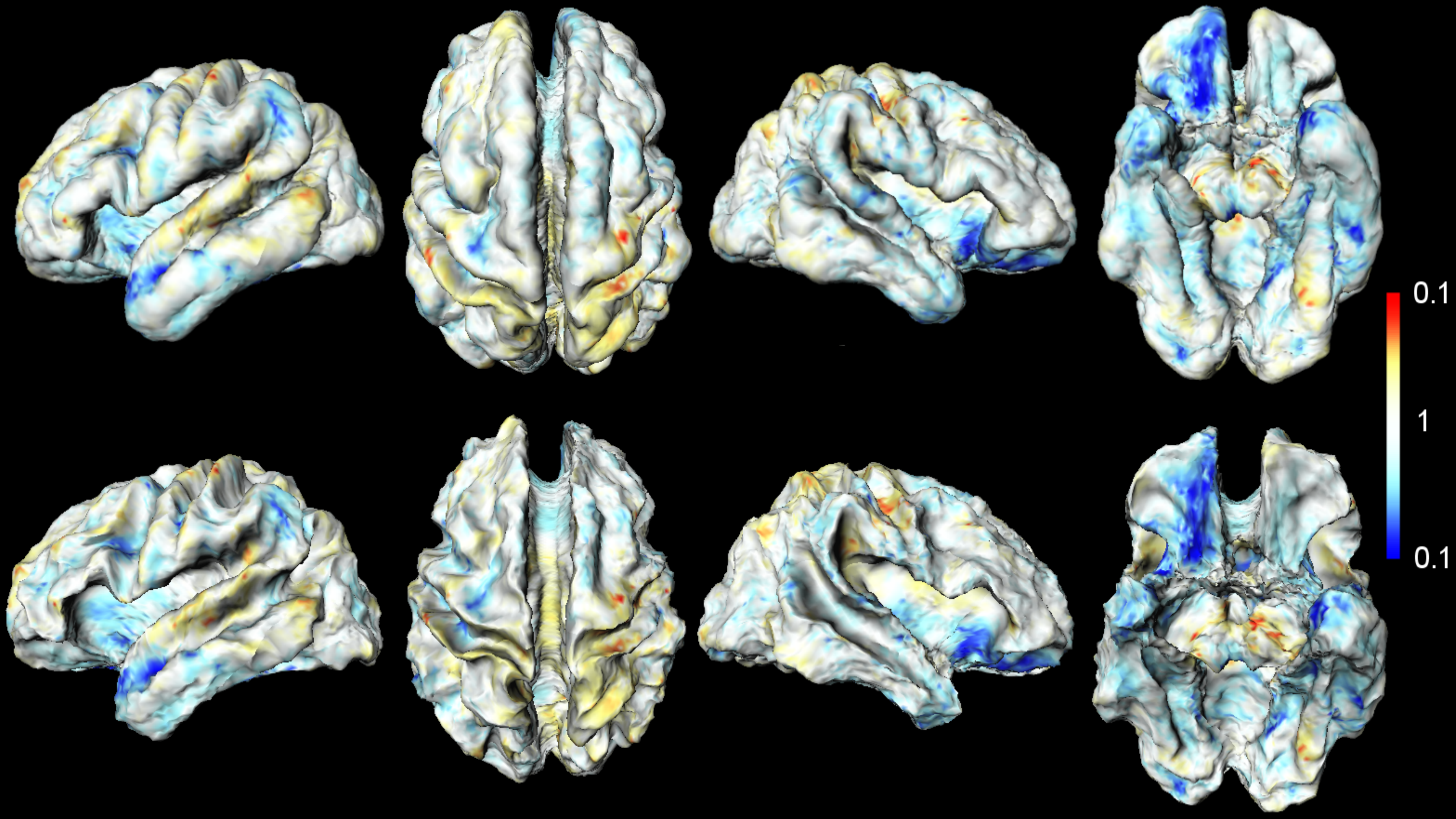
Age distribution for control subjects:

17.1 (s.d. 2.8)

In this study, there is no visible age effect.

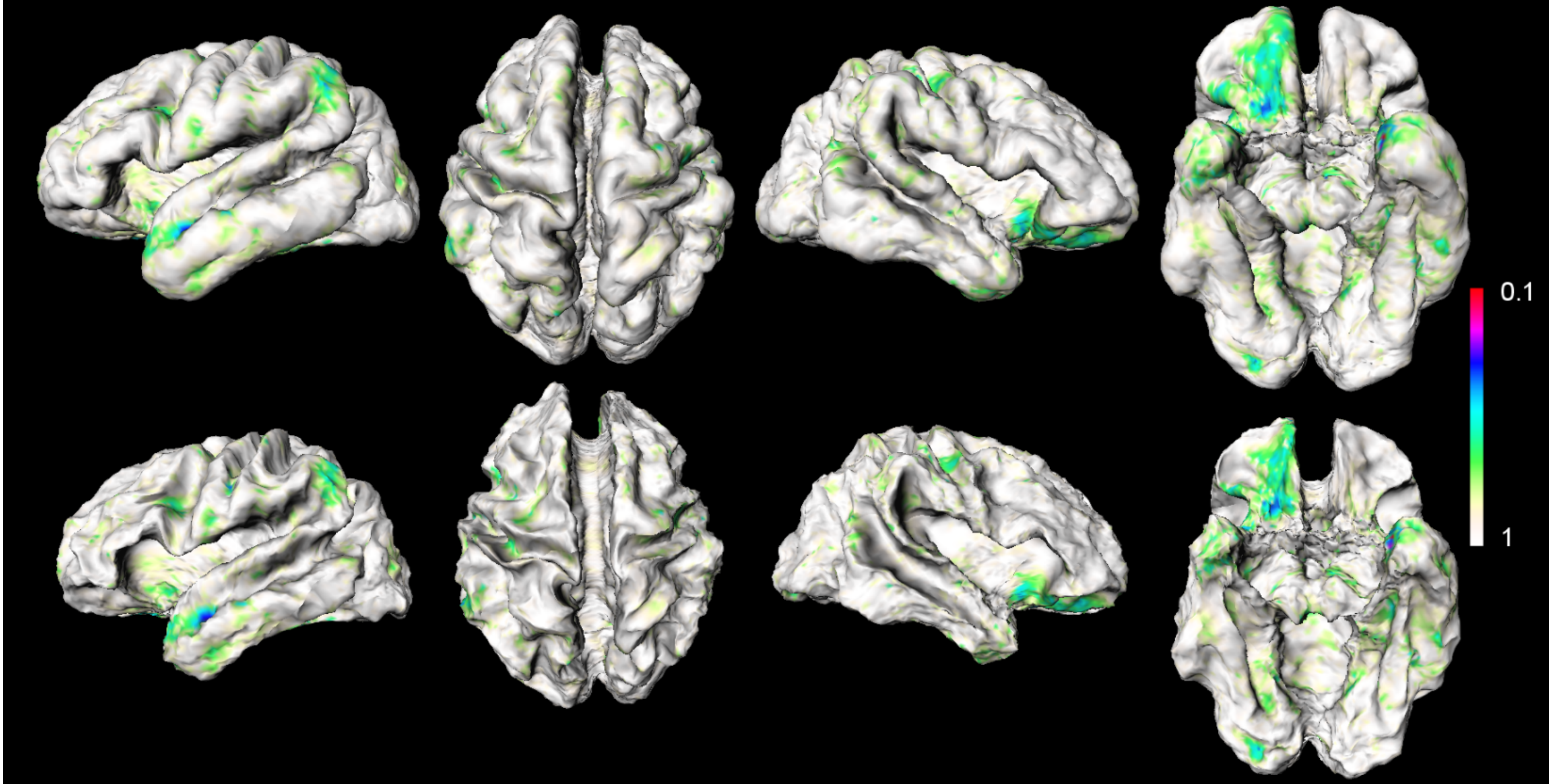
However, a different study on the same data set shows significant age effect.

p-value map for *t*-test on cortical thickness difference



Decrease: left superior temporal sulcus, left occipital-temporal gyrus, right orbital prefrontal
Increase: left superior temporal gyrus, left middle temporal gyrus, left and right postcentral sulci

p-value map for *F*-test removing age effect



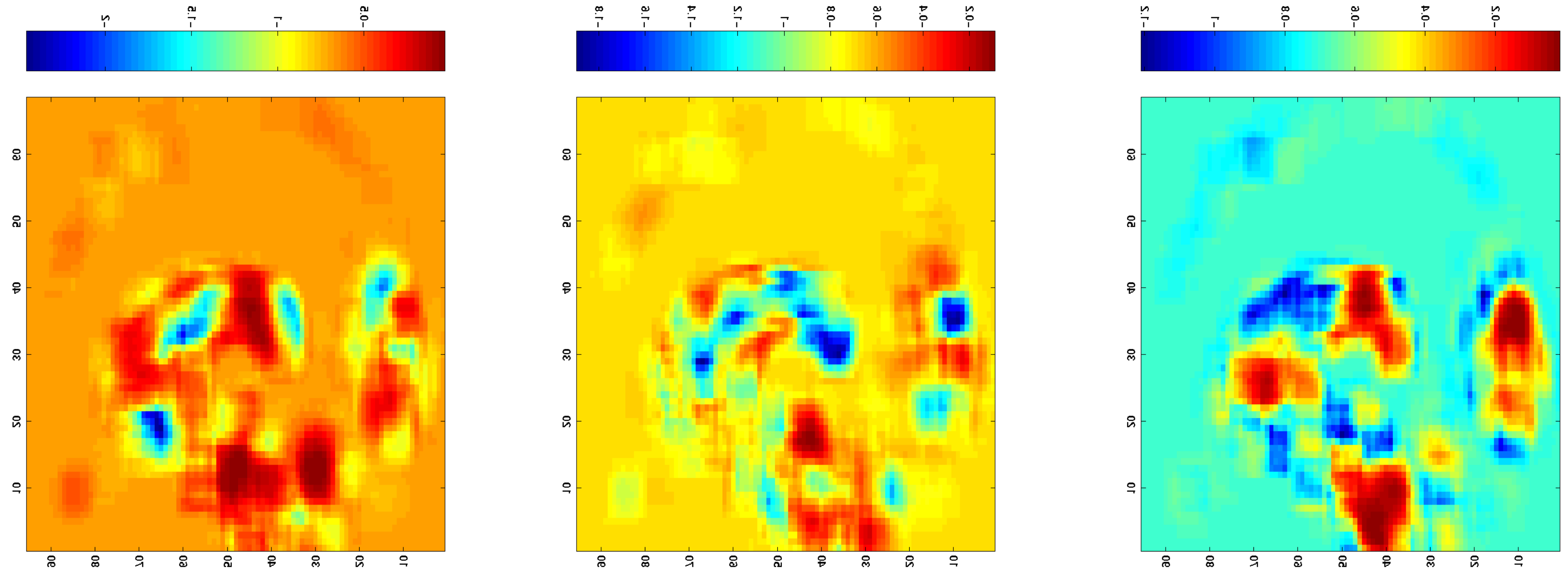
Decrease: **left superior temporal sulcus**
left occipital-temporal gyrus
right orbital prefrontal

It is better to remove the age effect in anatomical data especially for developmental age range (10-20 years).

How to validate VBM framework ?

This is not a permutation test although it looks like it.

Randomly permute 16 autism and 12 controls to generate 14 autism and 14 controls. Our two sample t-test should not detect anything except possible random occurrences.



Above all three random permutations, p-value > 0.3679

Issue of image registration in VBM

- If the registration is perfect, every parts of CC matches perfectly. It will result in similar gray matter density maps.
- Similar measures make differentiating populations difficult. Why? It reduces between-group variability.
- If the registration is coarse, most part of CC will not match and it will result in large within-group variability.
- In order for VBM to work, coarse registration seems to be sufficient.

Modulated VBM:

- There is a way to incorporate the amount of registration into VBM. We will talk about this after tensor-based morphometry.