

Data Mining: Large Databases and Methods

Or . . .

Brian D. Ripley
*Professor of Applied Statistics
University of Oxford*

ripley@stats.ox.ac.uk
<http://www.stats.ox.ac.uk/~ripley>

Finding Needles in Haystacks:

Finding unusual patterns
in large data sets

Fifteen years ago *data mining* was a pejorative phrase amongst statisticians, but the English language evolves and that sense is now encapsulated in the phrase *data dredging*.

In its current sense *data mining* means finding structure in large-scale databases.

It is one of many newly-popular terms for this activity, another being *KDD* (Knowledge Discovery in Databases), and is a subject at the boundaries of statistics, engineering, machine learning and computer science.

Data Mining Principles 1

Witten & Franke (2000) *Data Mining: Practical Machine Learning Tools with Java Implementations*:

Data mining is about solving problems by analyzing data already present in databases.

Data mining is defined as the process of discovering patterns in data.

The process must be automatic or (more usually) semi-automatic.

The patterns discovered must be meaningful in that they lead to some advantage, usually an economic advantage.

The data is invariably present in substantial quantities.

It is about finding and describing structural patterns in data.

And most of the techniques we cover have developed in a field known as *machine learning*.

Data Mining Principles 2

Hand, Mannila & Smyth (2001) *Principles of Data Mining*:

The science of extracting useful information from large data sets or databases is known as data mining.

Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner.

The statistical reader may be inclined to say “well this data mining material seems very similar in many ways to a course in applied statistics!”. And this is indeed somewhat correct

Such phrases are to a large extent fashion, and finding structure in datasets is emphatically *not* a new activity. In the words of Witten & Franke (p. 26)

What’s the difference between machine learning and statistics? Cynics, looking wryly at the explosion of commercial interest (and hype) in this area, equate data mining to statistics plus marketing.

What is new is the scale of databases that are becoming available through the computer-based acquisition of data, either through new instrumentation (fMRI machines can collect 100Mb of images in a hour’s session) or through the by-product of computerised accounting records (for example, spotting fraudulent use of credit cards or telephones, linking sales to customers through ‘loyalty’ cards).

Statistical Data Mining

This is a talk on *statistical* data mining. As such we will not cover the aspects of data mining that are concerned with querying very large databases, although building efficient database interfaces to statistical software is becoming an important area in statistical computing.

We will always need to bear in mind the ‘data dredging’ aspect of the term. When (literally) mining or dredging, the proportion of good material to dross is usually very low, and when mining for minerals can often be too low to cover the costs of extraction.

Exactly the same issues occur in looking for structure in ‘small’ data: it is all too easy to find structure that is only characteristic of the particular set of data to hand. We want *generalization* in the terminology of the psychologists, that is to find structure that will help with future examples too.

To paraphrase provocatively, ‘machine learning is statistics *minus* any checking of models and assumptions’.

‘Large Databases’

What is ‘large’ about large databases as used in data mining?

Normally just one of two aspects

- Many cases
 - motor insurance database with 66 million drivers (about 1/3 of all US drivers).
 - Sales data from Amazon, or an airline.
 - Credit-card transactions.
- Many observations
 - screening 10,000+ genes.
 - fMRI SPM map of t statistics for 100,000 voxels (per session, with less than 100 sessions).

An unusual example which has both is so-called CRM, e.g. supermarket sales records. Note the predominance of discrete observations.

Necessary Assumptions

In many (most?) cases we have lots (thousands to millions) of observations on a few subjects. Linear methods such as *principal component analysis* are not going to be well-determined.

There is an implicit assumption of a *simple explanation*. It is like model selection in regression: out of many regressors we assume that only a few are acting, individually or in combination.

Finding the genetic basis for say, a disease, is like this. We screen a few hundred people for 10,000 (even ‘all’ 30,000) genes. We have to assume at most a handful of genes are involved.

There is also a selection of clustering algorithms, the first of which is called ‘EM’ (and not in the Ted Harding/Tony Rossini sense — EM is widely used in engineering fields to mean fitting mixture distributions, a task better done in other ways).

‘and Methods’

What are the methods commonly considered as part of data mining (and which Witten & Franke claim are ‘developed’ in machine learning)?

- Rudimentary rule induction
- Statistical modelling, mainly Naïve Bayes.
- Decision trees
- Rule induction (via trees)
- Association rules
- Linear models
- ‘Instance-based learning’ (nearest-neighbours)

Their software, WEKA, covers a variety of tree methods (including random trees and random forests), some simple Bayes methods, linear, LMS and logistic regression, neural networks (including RBF nets), rule induction, and a host of ‘meta’ methods such as bagging, boosting and stacking.

Methods in R

R is pretty well off for these methods, although that reflects the research and teaching interests of a tiny number of contributors. Apart from base R, consider packages `class`, `e1071`, `mclust`, `nnet`, `rpart`, `tree` (and many others).

Because R is a programmable environment, it is easy to implement a meta-method, if Torsten Hothorn and colleagues have not already got there in package `ipred`.

Another area that is mentioned in both books is *visualizing data mining*. Base R has many of the techniques mentioned (including PCA and density estimation) and the VR bundle has most of the rest (MDS, SOM).

The major lack is rule induction, especially association rules. That is almost certainly not accidental, but R also lacks the methods social statisticians use to handle large sets of categorical variables, possibly including latent (continuous or categorical) variables.

The Seduction of Non-linearity

Why use *non-linear methods* such as neural networks, tensor splines, GAMs, classification trees, support vector machines, ... ?

- Because the computation has become feasible
- Because some of them are heavily promoted
- Because the scope of linear methods is little understood (interactions)
- Because a little non-linearity leads to universal approximators
- Because there is money in it!

Used well they can out-perform older methods.

Used by non-experts they can seriously under-perform older methods.

Non-linear visualization methods (multidimensional scaling, Kohonen's SOM) are under-used.

Don't Forget the Rest of Statistics

Normal statistical thinking is at least as important, including

- Sensible experimental design
- Data visualization
- Outlier detection
- Robustification
- Checking of assumptions
- Performance assessment

Needles in Haystacks

The analogy is rather helpful. We probably know what the 'hay' looks like. If we really know what 'needles' look like, the task is purely computational pattern matching. But in practice we either

- have seen some past examples of needles: supervised pattern recognition, or
- are interested in anything which is not hay: unsupervised pattern recognition.

or just possibly both. The second is much harder, and really we are only interested in some departures (e.g. in fraud detection in those which may lose us income).

Magnetic Resonance Imaging examples

Joint work with Jonathan Marchini.

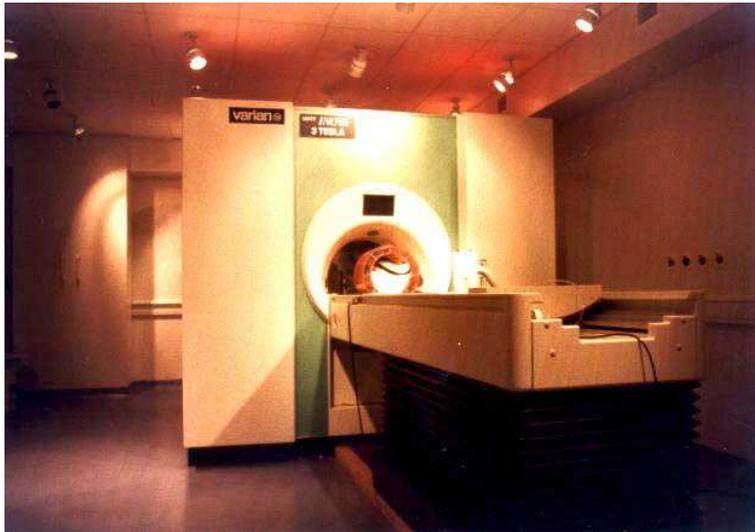
Part 1: Magnetic Resonance Imaging of Brain Structure

Data, background and advice provided by Peter Styles (MRC Biochemical and Clinical Magnetic Resonance Spectroscopy Unit, Oxford)

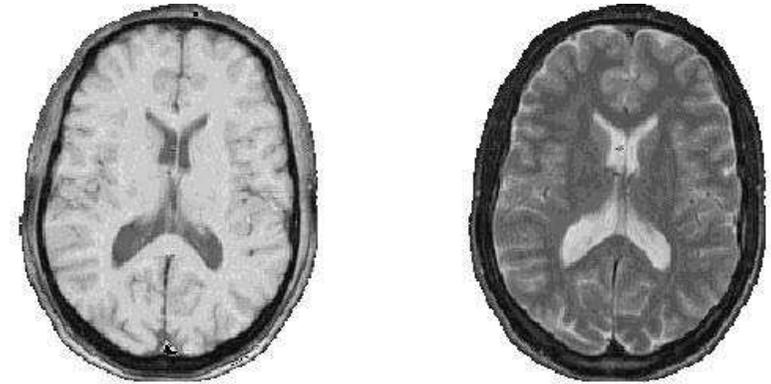
Part 2: Statistical Analysis of Functional MRI Data

Data, background and advice provided by Stephen Smith (Oxford Centre for Functional Magnetic Resonance Imaging of the Brain) and Nick Lange (McLean Hospital, Harvard).

Magnetic Resonance Imaging



Some Data



T1 (left) and T2 (right) MRI sections of a 'normal' human brain.
This slice is of 172×208 pixels. Imaging resolution was $1 \times 1 \times 5$ mm.

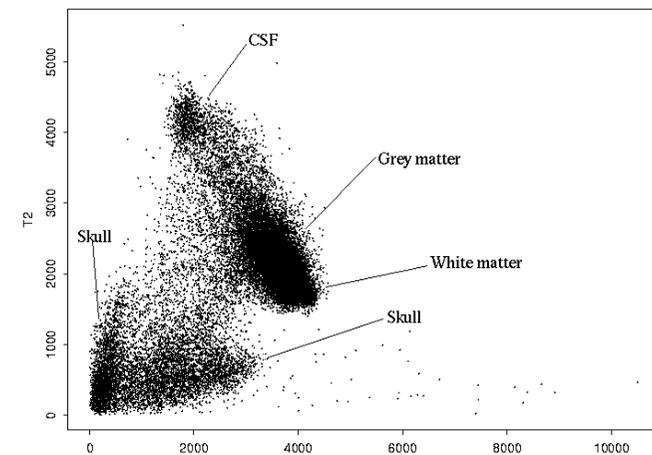
Neurological Change

Interest is in the change of tissue state and neurological function after traumatic events such as a stroke or tumour growth and removal. The aim here is to identify tissue as normal, impaired or dead, and to compare images from a patient taken over a period of several months.

Our initial task was exploring 'T1' and 'T2' images (the conventional MRI measurements) to classify brain tissue automatically, with the aim of developing ideas to be applied to spectroscopic measurements at lower resolutions.

Consider image to be made up of 'white matter', 'grey matter', 'CSF' (cerebro-spinal fluid) and 'skull'.

Initial aim is reliable automatic segmentation. Since applied to a set of patients recovering from severe head injuries.



Data from the same image in T1-T2 space.

Modelling

Our basic model is

$$\log Y_{ij} = \mu + \beta_{\text{class}(ij)} + s(i, j) + \epsilon_{ij}$$

for the intensity at voxel (i, j) , studied independently for each of the T1 and T2 responses. Here $s(x, y)$ is a spatially smooth function.

Of course, the equation depends on the classification, which will itself depend on the predicted bias field. This circularity is solved by iterative procedure, starting with no bias field.

Estimation

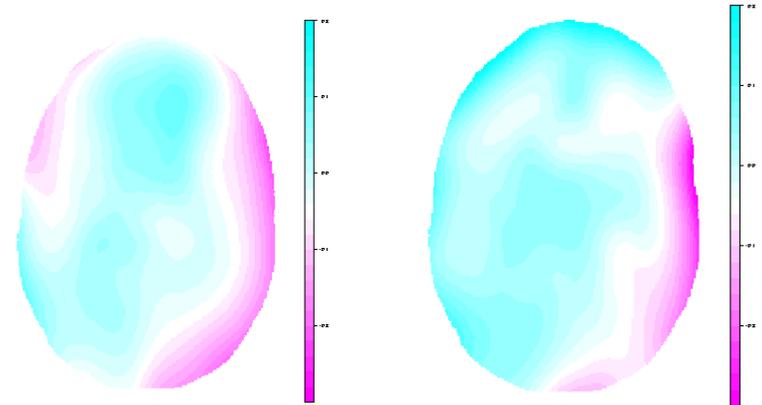
If the classification were known we would use a robust method that fits a long-tailed distribution for ϵ_{ij} , unconstrained terms β_c for each class, and a ‘smooth’ function s . We cope with unknown class in two ways. In the early stages of the process we only include data points whose classification is nearly certain, and later we use

$$\log Y_{ij} = \mu + \sum_{\text{class } c} \beta_c p(c | Y_{ij}) + s(i, j) + \epsilon_{ij}$$

that is, we average the class term over the posterior probabilities for the current classification.

For the smooth term s we initially fitted a linear trend plus a spline model in the distance from the central axis of the magnet, but this did not work well, so we switched to *loess*. Loess is based on fitting a linear surface locally plus approximation techniques to avoid doing for the order of 27 000 fits.

Fits of bias fields



Fitted ‘bias fields’ for T1 (left) and T2 (right) images.

The bias fields for these images are not large and change intensity by 5–10%.

Modelling the data

Each data point (representing a pixel) consists of one T1 and one T2 value. Observations come from a mixture of sources so we use a finite normal mixture model

$$f(y; \Psi) = \sum_{c=1}^g \pi_c \phi(y; \mu_c, \Sigma_c)$$

where the mixing proportions, π_c , are non-negative and sum to one and where $\phi(y; \mu_c, \Sigma_c)$ denotes the multivariate normal p.d.f with mean vector μ and covariance matrix Σ .

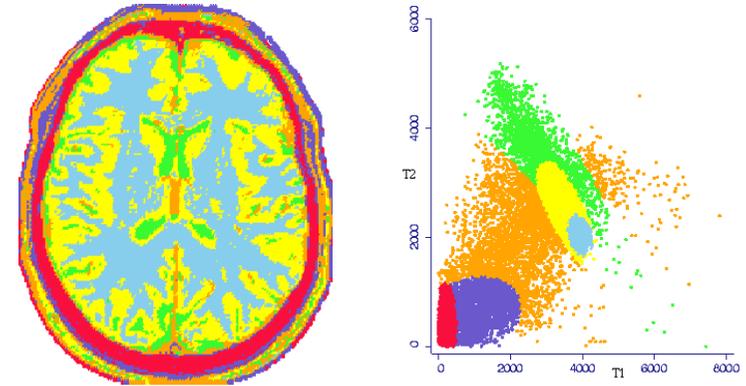
This is applied to bias-corrected data.

Application/Results

6 component model

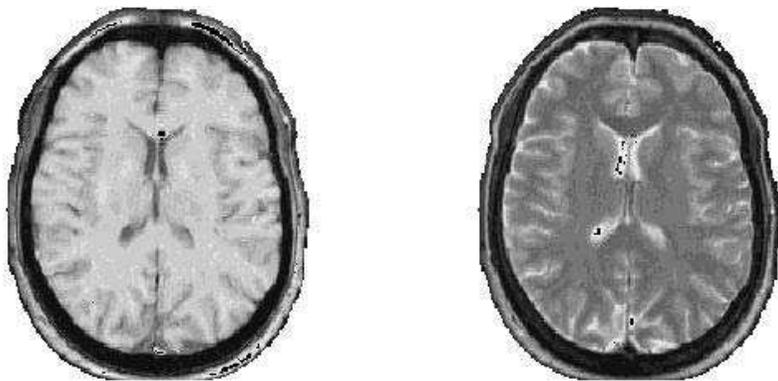
- CSF
- White matter
- Grey matter
- Skull type 1
- Skull type 2
- Outlier component (fixed mean and large variance)

Initial estimates chosen manually from one image and used in the classification of other images.



Classification image (left) and associated T1/T2 plot (right), training the 6-component mixture model from its fit on the reference subject.

A Second Dataset



T1 (left) and T2 (right) MRI sections of another 'normal' human brain.

Outliers and anomalies

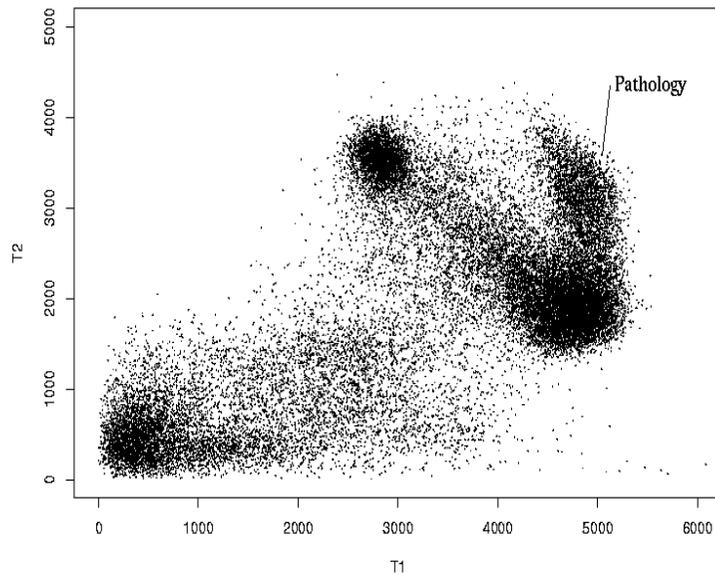
We have found our scheme to be quite robust to variation in imaging conditions and to different 'normal' subjects. The 'background' class helps greatly in achieving this robustness as it 'mops up' the observations which do not agree with the model.

However, outliers can be more extreme:

'Functional' MRI

Functional PET (positron emission spectroscopy: needs a cyclotron) and MRI are used for studies of brain function: give a subject a task and see which area(s) of the brain 'light up'.

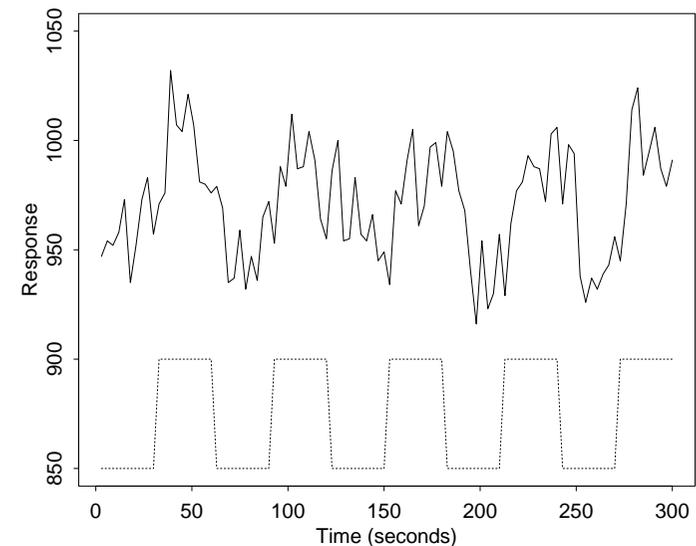
fMRI has a higher spatial and temporal resolution. Most commonly stimuli are applied for a period of 10–30 secs, images taken around every 3 secs, with several repeats of the stimulus being available for one subject. Down to $1 \times 1 \times 3$ mm voxels.



T1–T2 plot of a brain slice of a brain with a pathology.

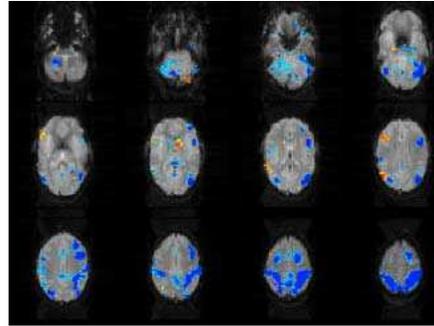
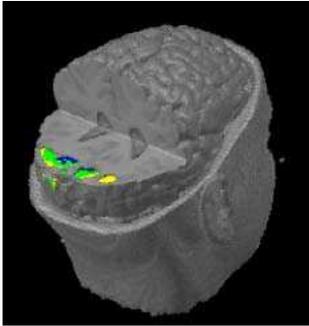
This illustrates the dangers of classifying all the points. This is a particularly common mistake when neural networks are used for classification, and we have seen MRI brain scans classified by neural networks where common sense suggested an 'outlier' report was the appropriate one.

The procedure presented here almost entirely ignores the spatial nature of the image. For some purposes this would be a severe criticism, as *contextual* classification would be appropriate. However, our interest in these images is not a pretty picture but is indeed in the anomalies, and for that we prefer to stay close to the raw data. The other interest is in producing summary measures that can be compared across time.



A real response (solid line) from a 100-scan (TR=3sec) dataset in an area of activation from the visual experiment. The periodic boxcar shape of the visual stimulus is shown below.

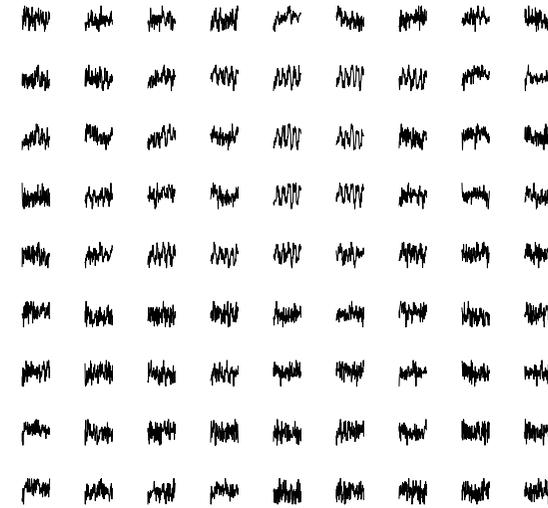
The commonly addressed statistical issue is ‘has the brain state changed’, and if so where?



Left: A pain experiment. Blue before drug administration, green after, yellow both.

Right: A verbal/spatial reasoning test, averaged over 4 subjects. 12 slices, read row-wise from bottom of head to top. Blue=spatial, red=verbal.

A Closer Look at some Data



A 10×10 grid in an area of slice 5 containing activation.

Multiple comparisons

Finding the voxel(s) with highest ‘ t ’ values should detect the areas of the brain with most change, but does not say they are significant changes. The t distribution *might* apply at one voxel, but it does not apply to the voxel with the largest response.

Conventional multiple comparison methods (e.g. Bonferroni) may over-compensate if the voxel values are far from independent.

Three main approaches:

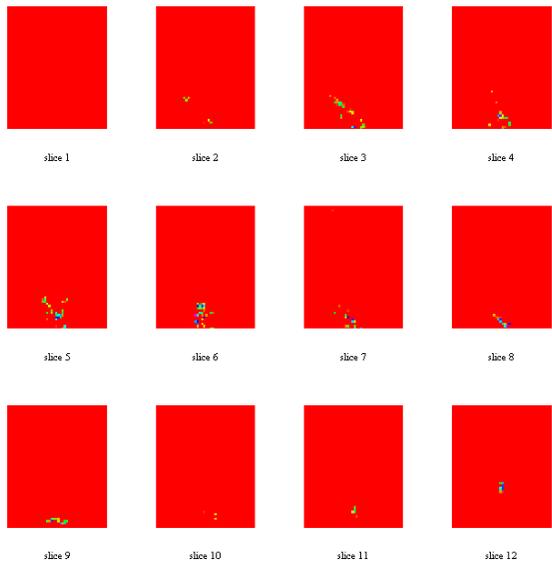
1. Randomization-based analysis (Holmes *et al*) across replications.
2. (High) level crossings of Gaussian stochastic processes (Worsley *et al*): *Euler characteristics*.
3. Variability within the time series at a voxel.

We only discuss the third here.

Principles of Our Analyses

- Work with raw data.
- Non-parametric robust de-trending, Winsorizing if required.
- Work in spectral domain.
- Match a filter to the expected pattern of response (square wave input, modified by the haemodynamic response).
- Non-parametric smooth estimation of the noise spectrum at a voxel, locally smoothed across voxels.
- Response normalized by the noise variance should be Gumbel (with known parameters) on log scale.

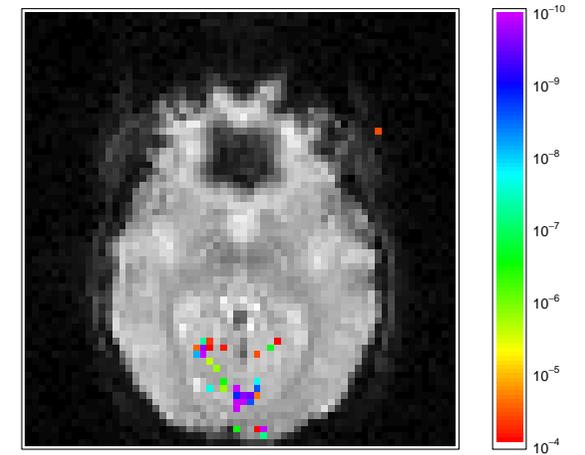
This produced much more extreme deviations from the background variation, and compact areas of response. ca 10 minutes for a whole brain (in R on a 1Ghz PC).



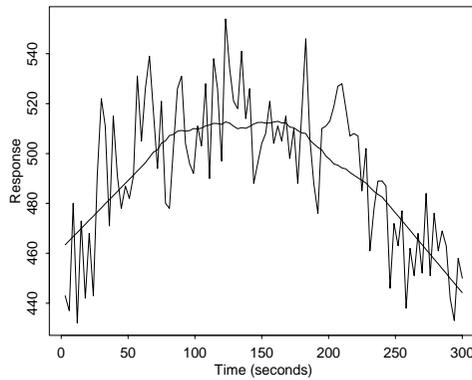
Log abs filtered response, with small values coloured as background (red). Threshold for display is $p < 10^{-5}$ (and there are ca 20,000 voxels inside the brain here).

Plotting p values

p -value image of slice 5 thresholded to show p -values below 10^{-4} and overlaid onto an image of the slice. Colours indicate differential responses within each cluster. An area of activation is shown in the visual cortex, as well as a single 'false-positive', that occurs outside of the brain.



Trend-removal



A voxel time series from the dataset showing an obvious non-linear trend.

We used a running-lines smoother rejecting outliers (and Winsorizing the results).

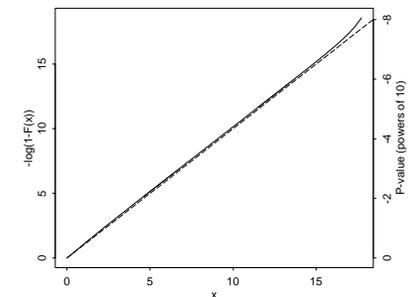
Calibration

Before we worry about multiple comparisons, are the t -statistics (nearly) t -distributed?

Few people have bothered to check, and those who did (Bullmore, Brammer *et al*, 1996) found they were not.

We can use null experiments as some sort of check.

In our analysis we can use other frequencies to self-calibrate, but we don't need to:



Conclusions

- Look at your data (even if it is on this scale: millions of points per experiment).
- Data ‘cleaning’ is vital for routine use of such procedures.
- You need to be sure that the process is reliable, as no one can check on this scale.
- Successful data mining very often depends on making the right high-level assumptions and designing the study well enough.
- It is amazing what can be done in high-level languages such as R on cheap computers.