Confidence and Uncertainty

Daniel Alexander

Uncertainty

- We would like to characterize the confidence or uncertainty in parameter estimates.
- Provide confidence intervals on $\mathbf{x}$.
- Reconstruct the posterior distribution on $\mathbf{x}$.

- Why?
- Laplace's method
- Bootstrap resampling techniques
  - Parametric
  - Non-parametric
  - Residual/Wild
- Markov Chain Monte Carlo (MCMC)

Basic parametric mapping

ELSEVIER

NeuroImage

www.elsevier.com/locate/ynimg
NeuroImage 27 (2005) 436 – 441

Three-dimensional quantitative magnetisation transfer imaging of the human brain

Mara Cercignani,[a,*] Mark R. Symms,[b] Klaus Schmierer,[a] Philip A. Boulby,[b] Daniel J. Tozer,[a] Maria Ron,[a] Paul S. Tofts,[a] and Gareth J. Barker[c]

Eg: quantitative magnetization transfer study in the normal human brain.

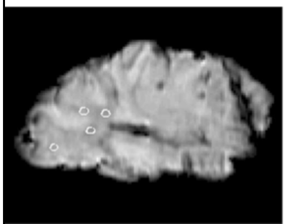


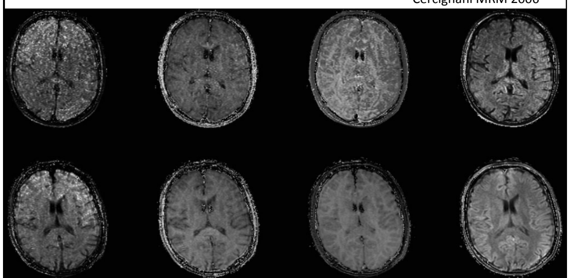
Table 1
Average quantitative MT parameters from the 3D-MTSPGR sequence, measured in the brain of six healthy volunteers

Area			f [%]	T_2^B [µs]	T_1^A [ms]	T_{1obs} [ms]	T_2^A [ms]
Corpus callosum	Genu	Mean	10.1	11.4	726	728	56.7
		SD	0.8	0.8	68	68	10.2
	Splenium	Mean	9.7	11.9	711	713	73.0
		SD	1.2	0.4	83	83	43.4
	Body	Mean	8.0	11.4	853	854	58.3
		SD	0.8	1.0	74	74	18.5
Internal capsule	Genu	Mean	9.0	12.7	749	751	54.8
		SD	1.2	1.2	67	67	12.7
	Posterior limb	Mean	9.3	13.8	734	736	56.7
		SD	1.1	8.9	47	47	4.9
Corona radiata		Mean	8.8	13.7	730	732	66.2
		SD	0.7	7.9	46	46	7.8
Optic radiation		Mean	9.4	12.3	661	663	61.1
		SD	0.9	8.4	44	44	7.7
Pedunculus cerebri		Mean	9.2	13.4	745	747	72.5
		SD	1.3	0.9	61	60	29.9
Cerebellar WM		Mean	9.4	11.8	718	719	59.0
		SD	0.8	0.5	51	51	3.8
Pons		Mean	8.1	11.7	825	827	77.3
		SD	0.9	1.1	70	70	17.3
Frontal WM		Mean	9.1	13.4	719	720	53.5
		SD	0.6	0.8	41	40	5.0
Thalamus		Mean	6.6	11.8	975	976	69.8
		SD	0.7	0.8	61	61	8.5
Caudate		Mean	5.0	11.4	1164	1164	82.8
		SD	0.6	0.8	117	117	19.0
Putamen		Mean	5.2	10.9	1092	1092	78.6
		SD	0.7	0.6	85	85	9.9
Frontal GM		Mean	4.7	11.8	1279	1279	107.6
		SD	0.6	1.6	128	128	26.2

f = macromolecular proton fraction; A = free pool; B = restricted pool; SD = standard deviation; GM = grey matter; WM = white matter. See text for further details.
With the exception of the corpus callosum and the pons, the values were measured bilaterally.

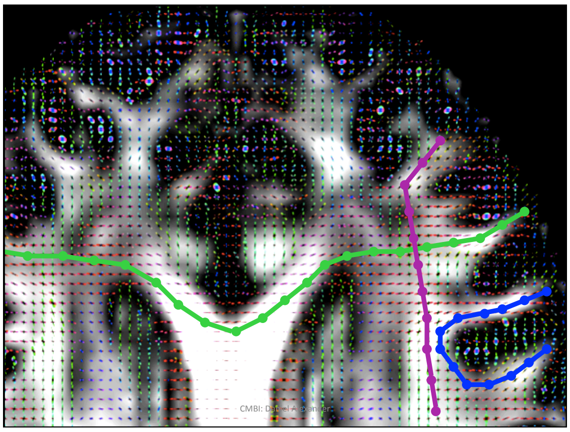
qMT maps

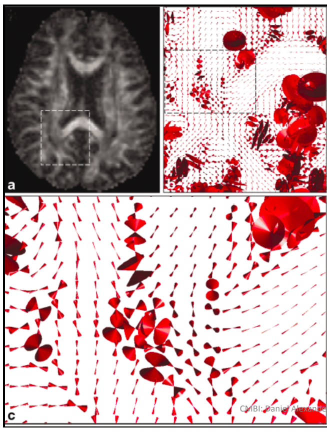
Cercignani MRM 2006



RM_0^A F/R_A T_2^B $1/R_AT_2^A$

Tracto-graphy





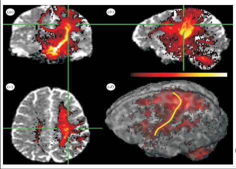
D.K. Jones
Determining and Visualizing Uncertainty in Estimates of Fiber Orientation From Diffusion Tensor MRI *Magnetic Resonance in Medicine* 2004.

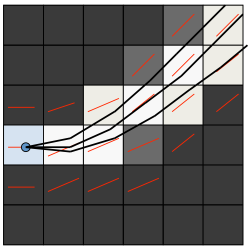
FIG. 2. Cones of uncertainty (showing the 95% confidence angle) at the level of the splenium of the corpus callosum. **a:** Fractional anisotropy. **b:** Cones of uncertainty in the region indicated by the dashed lines in **a**. This region is further magnified in **c**. The zoomed area highlights a region where fibers cross and the uncertainty in ε_1 is large.

CMBI: Daniel Alexander 8

Probabilistic Tractography

- Model uncertainty of fiber orientations
- Repeat:
  - Sample orientations
  - Track a streamline
- Count streamlines in each voxel





This is the basic PICo algorithm by Parker et al, JMRI 2003.

CMBI: Daniel Alexander 9

Laplace's method

- Generally, a method to approximate integrals.
- In our context: approximates the posterior distribution as a Gaussian.
- The mean is the MAP estimate.
- The covariance is

$$\Sigma = -\left(\frac{d^2}{d\mathbf{x}^2}\log p(\tilde{\mathbf{x}} \mid \mathbf{A})\right)^{-1} = -\left(H(\tilde{\mathbf{x}})\right)^{-1}$$

- Diagonal elements provide 2σ range.

CMBI: Daniel Alexander 10

Help

fminunc

Search Documentation

Optimization Toolbox | Nonlinear Programming | Unconstrained Optimization

fminunc

Find minimum of unconstrained multivariable function

Equation

Finds the minimum of a problem specified by

$$\min_x f(x)$$

where *x* is a vector and *f*(*x*) is a function that returns a scalar.

Syntax

```
x = fminunc(fun,x0)
x = fminunc(fun,x0,options)
x = fminunc(problem)
[x,fval] = fminunc(...)
[x,fval,exitflag] = fminunc(...)
[x,fval,exitflag,output] = fminunc(...)
[x,fval,exitflag,output,grad] = fminunc(...)
[x,fval,exitflag,output,grad,hessian] = fminunc(...)
```

CMBI: Daniel Alexander 11

Bootstrapping

- Use the data to guess its own distribution.
- Use the data distribution to estimate the parameter distribution.

CMBI: Daniel Alexander 12

Reminder of terms

$$\mathbf{A} = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_K \end{pmatrix} \text{ Data} \qquad \mathbf{S} = \begin{pmatrix} S(\mathbf{y}_1;\tilde{\mathbf{x}}) \\ S(\mathbf{y}_2;\tilde{\mathbf{x}}) \\ \vdots \\ S(\mathbf{y}_K;\tilde{\mathbf{x}}) \end{pmatrix} \text{ Model signals}$$

$$\tilde{\mathbf{x}} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} \text{ Parameter estimate} \qquad \mathbf{r} = \begin{pmatrix} A_1 - S(\mathbf{y}_1;\tilde{\mathbf{x}}) \\ A_2 - S(\mathbf{y}_2;\tilde{\mathbf{x}}) \\ \vdots \\ A_K - S(\mathbf{y}_K;\tilde{\mathbf{x}}) \end{pmatrix} \text{ Residuals}$$

Parametric bootstrap

- Fit model parameters
- Compute model signals
- `For t=1:T`
  - Sample $\mathbf{E} = \eta_1, \ldots, \eta_K$ from noise distribution eg $N(0,\tilde{\sigma})$
  - Synthesize bootstrap data set $\hat{\mathbf{A}}_t = \mathbf{S} + \mathbf{E}$
  - Estimate best fit $\mathbf{x}_t$ to $\hat{\mathbf{A}}_t$
- Output samples $\mathbf{x}_1, \ldots, \mathbf{x}_T$ of $p(\mathbf{x} \mid \mathbf{A})$.

Estimating noise parameters

- Various ways to estimate from the residuals.
- For example

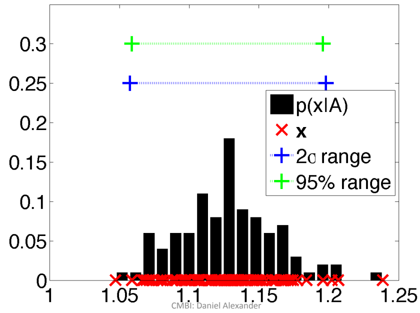
$$\tilde{\sigma}^2 = \frac{1}{K-N}\sum_{i=1}^{K} r_i^2$$

- Better (Davison and Hinkley 1997)

$$\tilde{\sigma}^2 = \frac{1}{K-1}\sum_{i=1}^{K} \frac{r_i^2}{1-h_{ii}^2}$$

- where $H = (h_{ij}) = G\,(G^T G)^{-1}\,G^T$ is the hat matrix.

Example output



Bootstrap samples

- Compute confidence intervals
  - 2σ range appropriate if the distribution is Gaussian
  - 95% range is from the $0.025T$-th sample to the $0.975T$-th sample.
- Allow us to visualize the distribution on $\mathbf{x}$.

Non-parametric bootstrap

- Resample data directly
- In each variant, the overall algorithm is as for parametric bootstrap, but the method for generating the bootstrap sample differs

Classical bootstrap

- Samples from the original measurements with replacement:

  $\hat{A}_{ij} = A_{\lfloor U(1,K+1) \rfloor}$

- The subscript on the right hand size is a uniformly distributed integer in the range [1, K +1].

Repetition bootstrap

- Requires several, R, repeat measurements:

$$\begin{pmatrix} A_{11} & A_{12} & \dots & A_{1R} \\ A_{21} & A_{22} & \dots & A_{2R} \\ : & : & : & : \\ A_{K1} & A_{K2} & \dots & A_{KR} \end{pmatrix}$$

- Each bootstrap data set picks one from each $(A_{k1}, A_{k2}, \dots, A_{kR})$.

  $\hat{A}_{ij} = A_{j\lfloor U(1,R+1) \rfloor}$

Bootknife

- Repetition bootstrap underestimates variance.
- Remove 1 or more (sets of) measurements at random each iteration.

- Hesterberg, Proc. ASA, 2924-2930, 2004.
- Chung et al NeuroImage 33(2) 2006.

Residual bootstrap

- Resamples the residuals rather than the data
- For each bootstrap data point select one of the residuals $r_1, \dots, r_K$ at random and add it to the corresponding model signal.

  $\hat{A}_{ij} = S(y_j; \tilde{\mathbf{x}}) + r_{\lfloor U(1,K+1) \rfloor}$

Wild bootstrap

- Multiplies each residual by a standard normal distribution

  $\hat{A}_{ij} = S(y_j; \tilde{\mathbf{x}}) + r_j N(0,1)$

Issues with bootstrap

- Which kind of bootstrap?
  - Parametric bootstrap assumes noise model is correct
  - Classical bootstrap disrupts experiment design
  - Repetition bootstrap is expensive
  - Residual and wild bootstrap retain experiment design economically, but assume noise model.
  - Wild bootstrap accommodates different noise models on different data points

- How many bootstrap samples do I need?
  - Monitor convergence of statistic of interest

MCMC – Metropolis-Hastings

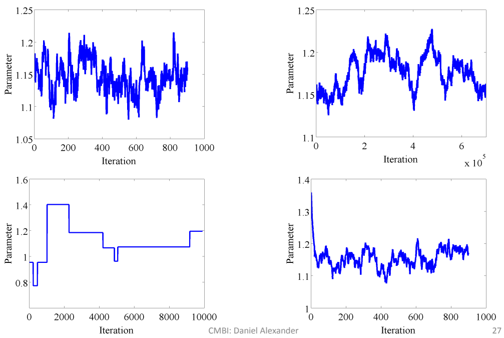
$\mathbf{x}_0$ = start point;
for t=1…T
$\quad \mathbf{x}_c = Q(\mathbf{x}_{t-1})$; % Draw from proposal $q(., \mathbf{x}_{t-1})$.
$\quad$ if $(\alpha(\mathbf{x}_c, \mathbf{x}_{t-1}) > U(0,1))$ % U is uniform.
$\quad\quad \mathbf{x}_t = \mathbf{x}_c$;
$\quad$ else
$\quad\quad \mathbf{x}_t = \mathbf{x}_{t-1}$;

$$\boxed{a(\mathbf{x},\mathbf{y}) = \frac{p(\mathbf{x} \mid \mathbf{A})q(\mathbf{x} \mid \mathbf{y})}{p(\mathbf{y} \mid \mathbf{A})q(\mathbf{y} \mid \mathbf{x})}}$$

Algorithm parameters

- Burn-in
  - Discard the first B samples to allow the chain to converge on the right distribution from the starting point.
- Sampling interval
  - Only keep every I-th sample to ensure independence of consecutive samples
- Proposal distribution $q(., \mathbf{x})$
  - Eg. Gaussian, t-distribution.
  - Perturbation size must reflect parameter scale
  - Rule of thumb: aim for acceptance rate of 20-50%.
  - Covariance and orientation
  - Non-Cartesian
- Number of samples
  - Must be enough to define the statistic of interest precisely.
  - Multiple chains can help identify convergence.
- See Gilks et al "MCMC in practice" Chapman and Hall 1996.

Common problems



MCMC

- Always check the chains
- Run initially for a very long time to gauge behaviour.
- Tune perturbation sizes for each parameter
- Check convergence of your statistic of interest.
- Lots of variations:
  - Reversible jump
  - Gibbs sampling, independence sampling, etc.

Tractography again

- Parker et al JMRI 2003 (Parametric bootstrap)
- Behrens et al MRM 2004 (MCMC)
- Lazar et al NeuroImage 2005 (Repetition bootstrap)
- Friman et al TMI 2006 (Laplace's method… sort of).
- Jones et al TMI 2008 (Wild bootstrap)
- Chung et al Neuroimage 2006; Jeurissen et al HBM 2012 (Bootstrap comparison)

- Is the distribution $p(\mathbf{x}|\mathbf{A})$ really what we want?

Summary

- Uncertainty of parameter estimates is important to know
  - For reliability of inference
  - For post-processing
- Laplace method
- Resampling methods: bootstrap
- MCMC