Bayesian model selection and averaging

Stefan Frässle

Translational Neuromodeling Unit (TNU)
University of Zurich & ETH Zurich

Methods and Models for fMRI analysis, Practical Session
Tuesday, November 26th 2019

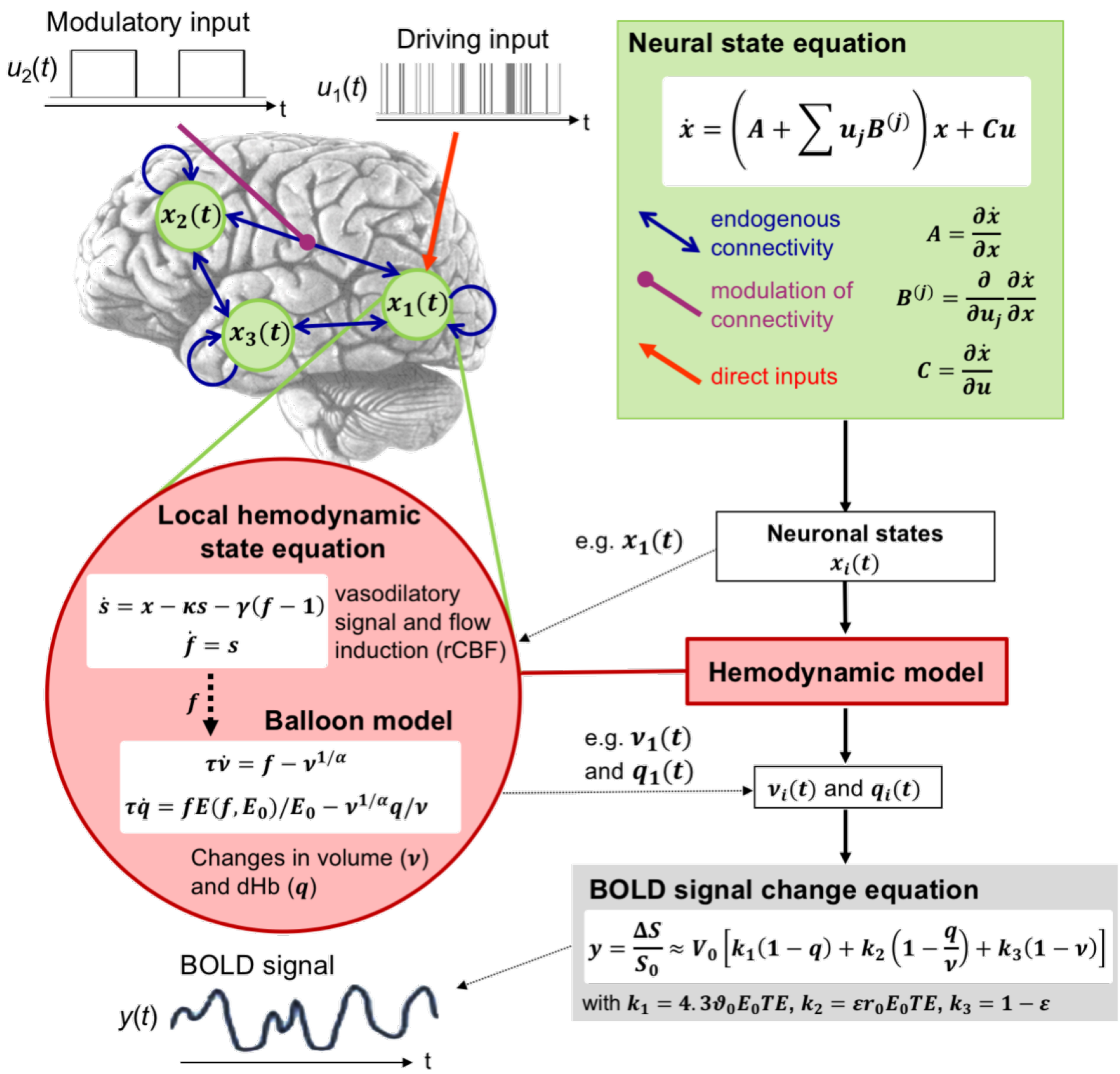
TODAY'S TUTORIAL

Bayesian model selection and averaging:

- Brief teaser on Dynamic Causal Modeling (DCM)
- Fixed- and random-effects Bayesian model selection (BMS)
- Bayesian model averaging (BMA)

TEASER: DCM FOR FMRI

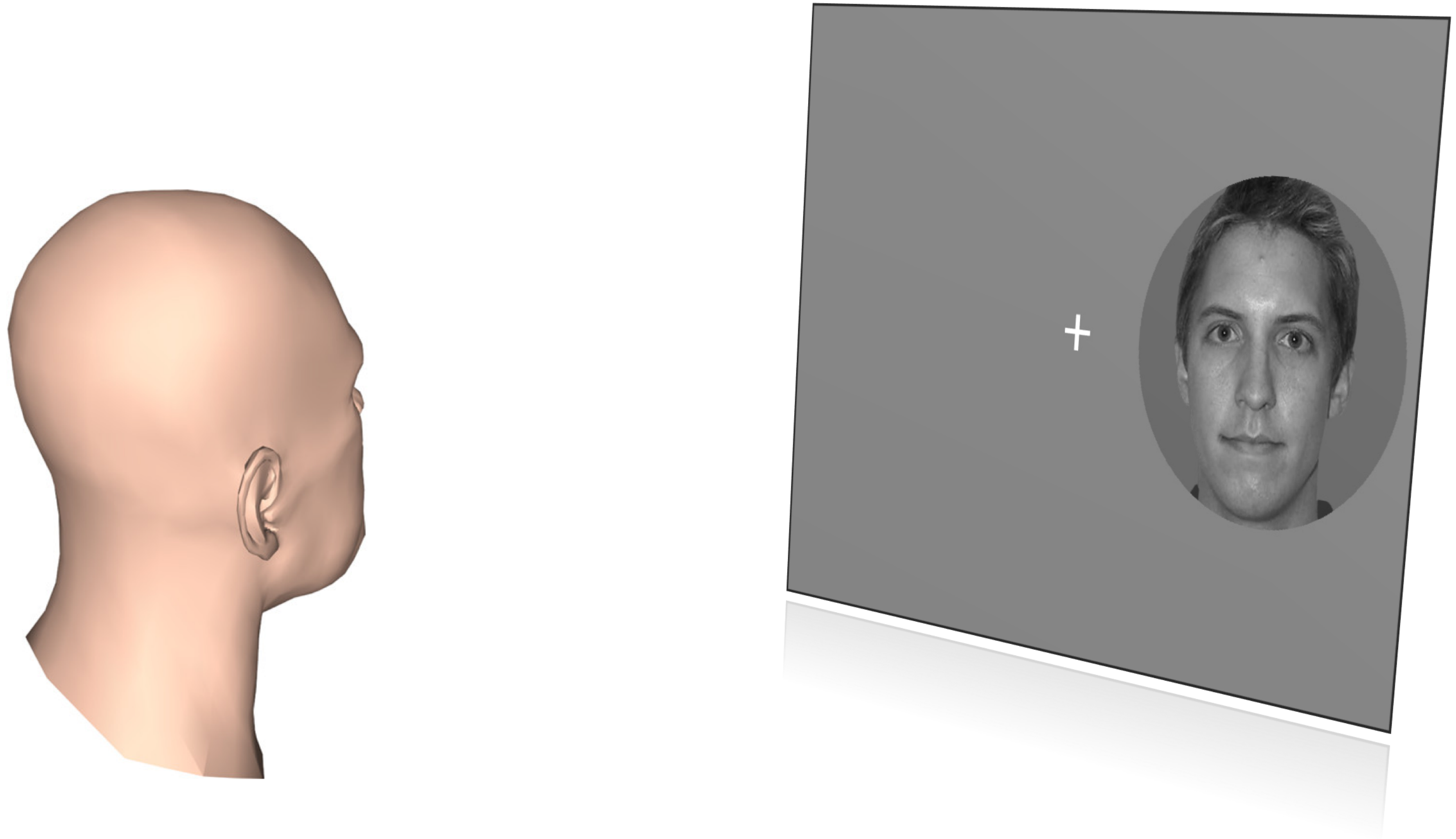
FYI: a more detailed introduction to DCM will follow in two weeks



Friston et al., 2003, *NeuroImage*; Stephan et al., 2015, *Neuron*

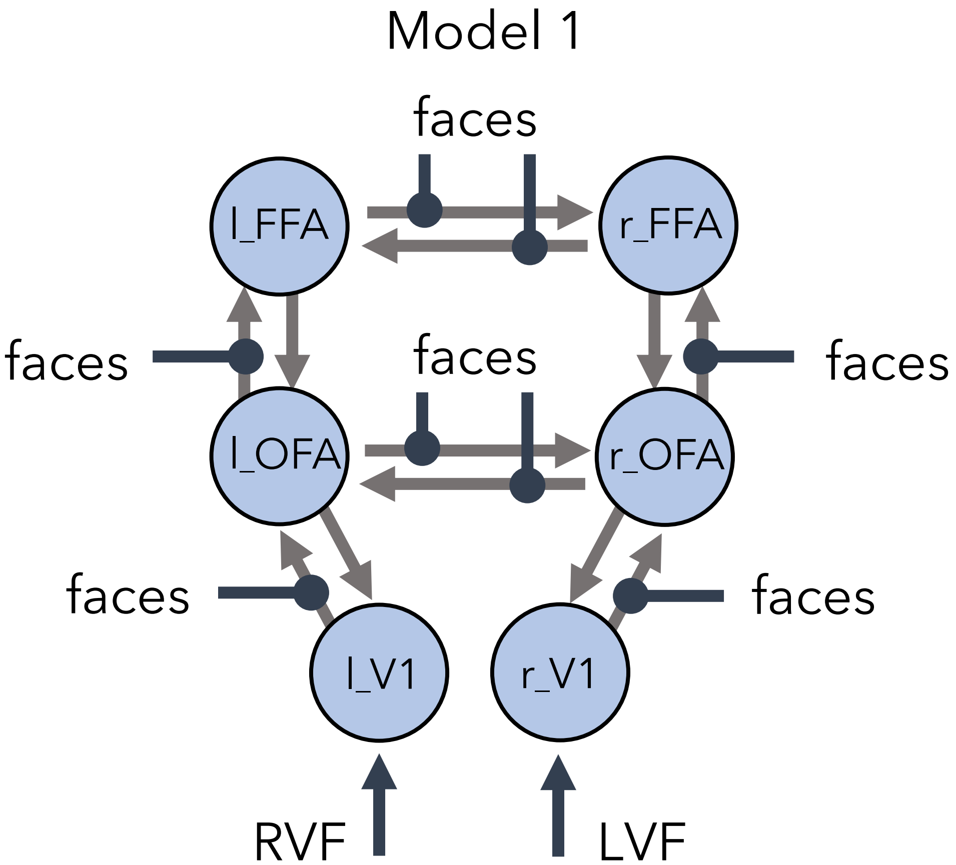
DYNAMIC CAUSAL MODELING

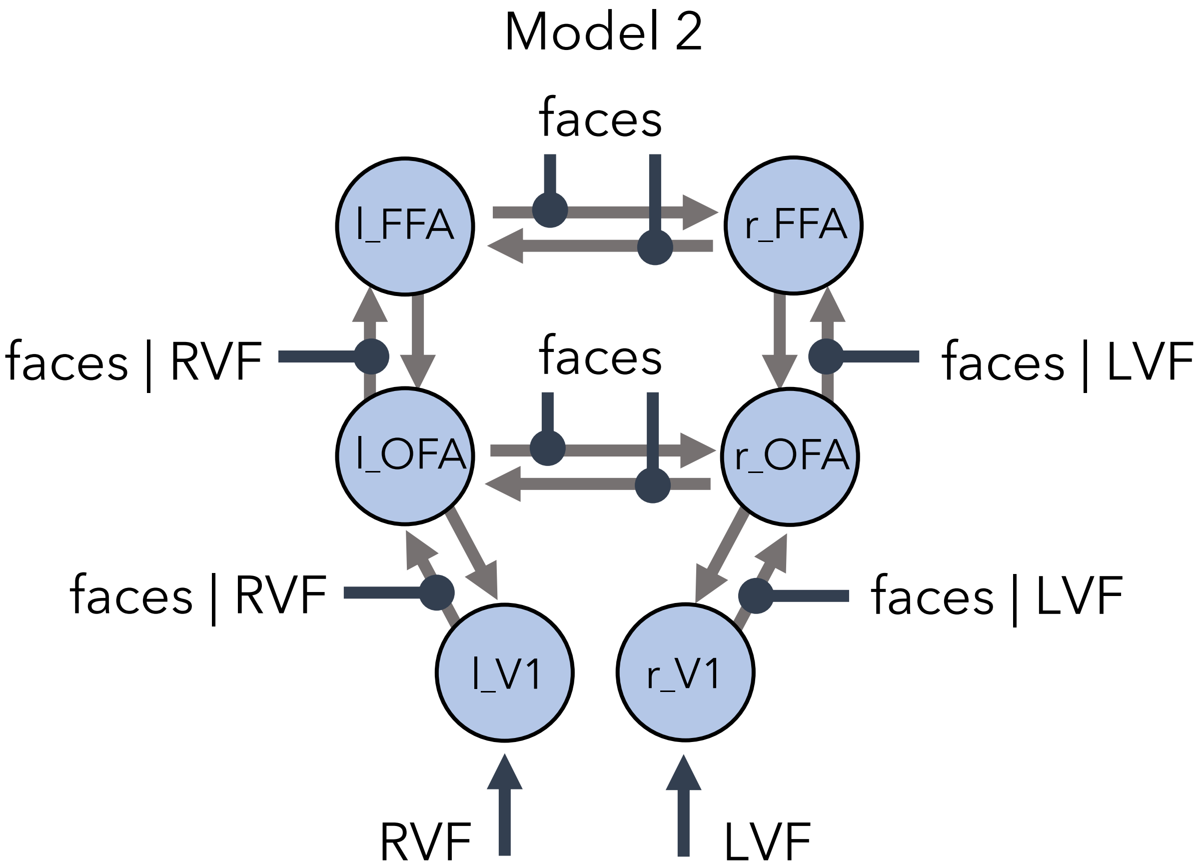
Different mechanisms in the face perception network:



DYNAMIC CAUSAL MODELING

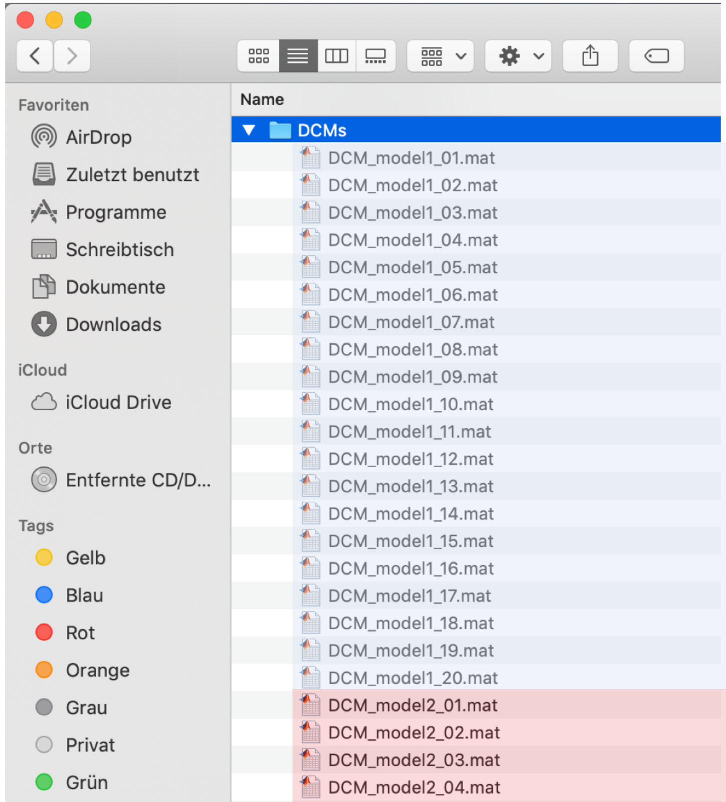
Different mechanisms in the face perception network:





ORGANIZATION OF DATA

Data is stored in "**DCM_model*.mat**" files:



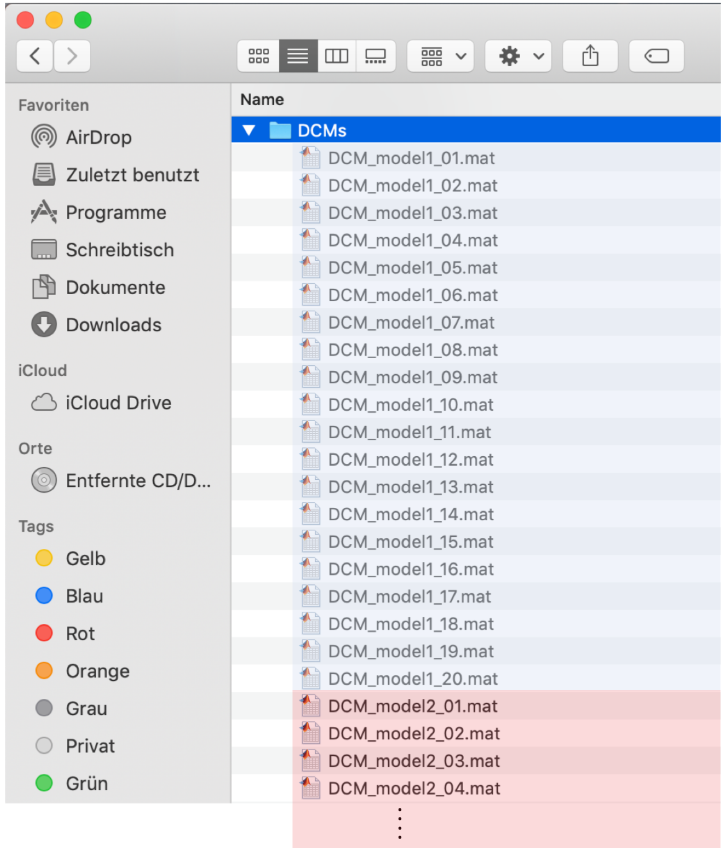
⋮

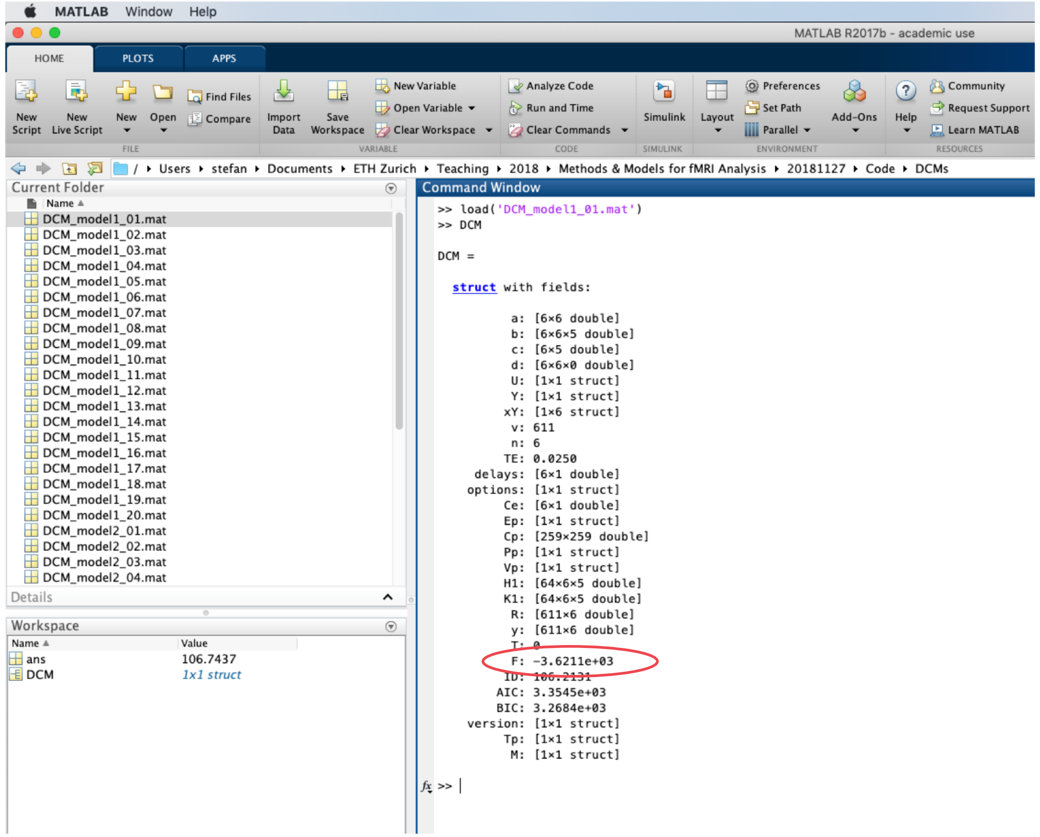
Model 1 for 20 subjects

Model 2 for 20 subjects

Organization of Data

Data is stored in "**DCM_model*.mat**" files:





Try to set up a Fixed-Effects BMS analysis

Reminder: **Bayes factors** and **Group Bayes Factors**:

$$BF_{ij} = \frac{p(y|m_i)}{p(y|m_j)} \longrightarrow \log(BF_{ij}) = \log(p(y|m_i)) - \log\left(p(y|m_j)\right)$$

$$GBF_{ij} = \prod_k BF_{ij}^{(k)} = \prod_k \frac{p(y_k|m_i)}{p(y_k|m_j)} \longrightarrow \log(GBF_{ij}) = \sum_k \log\left(BF_{ij}^{(k)}\right)$$

Kass & Raftery, 1995, *J. Am. Stat. Assoc.*

Try to set up a Fixed-Effects BMS analysis

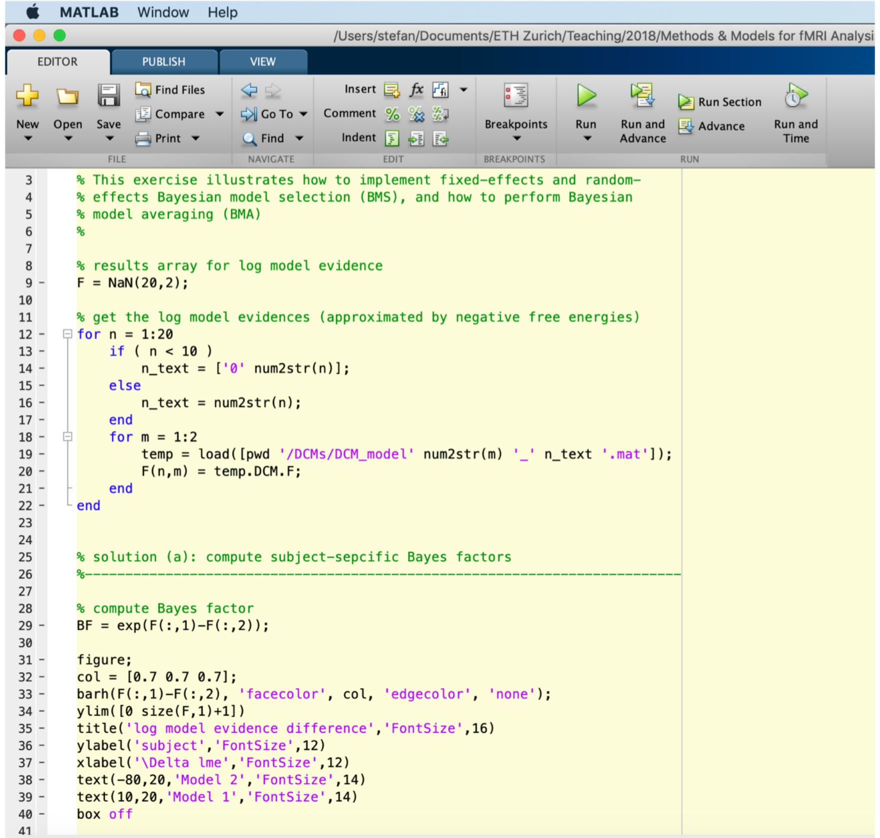
Implement the following two analyses:

(1) Compute the individual log Bayes factors and plot the results. You should write your own MATLAB code for this. The negative free energies serve as our metric of model goodness and are stored in the "**DCM.F**" field.

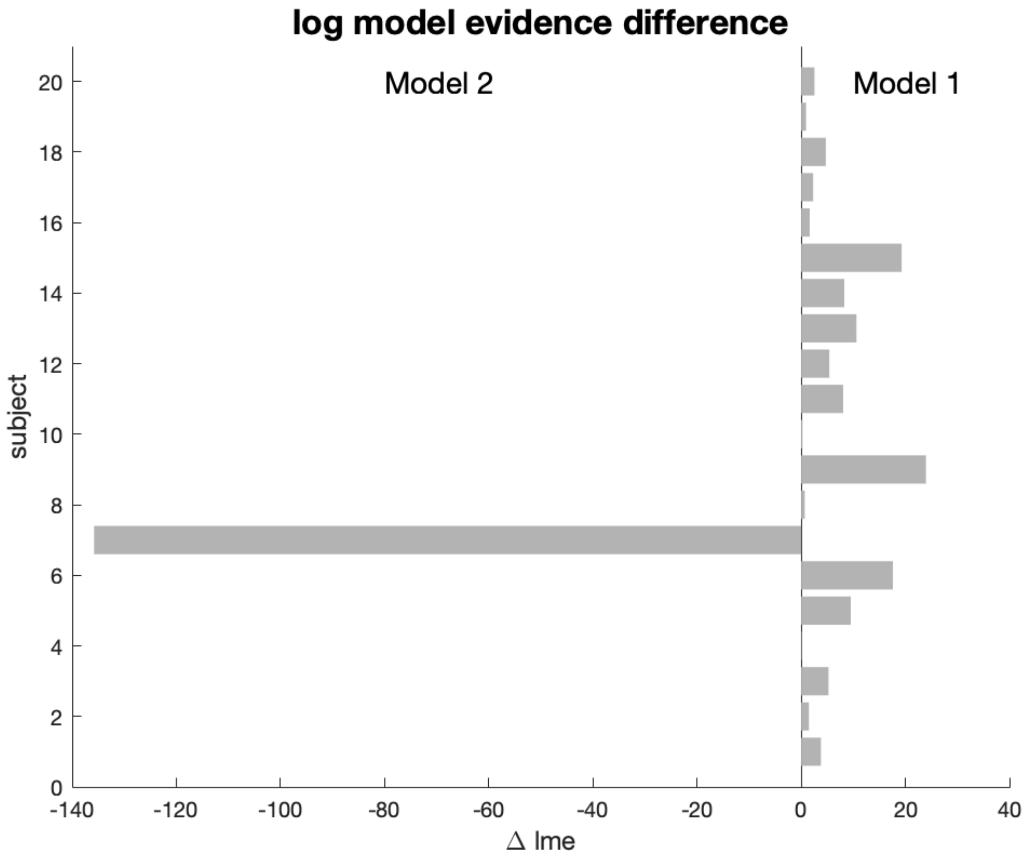
(2) Perform fixed-effects Bayesian model selection by computing the Group Bayes factor and/or the posterior model probabilities. You can either use the SPM GUI or write your own code. What do you notice when comparing the results from (1) and (2)?

TRY TO SET UP A FIXED-EFFECTS BMS ANALYSIS

(1) Implementation of individual log Group Bayes factor:

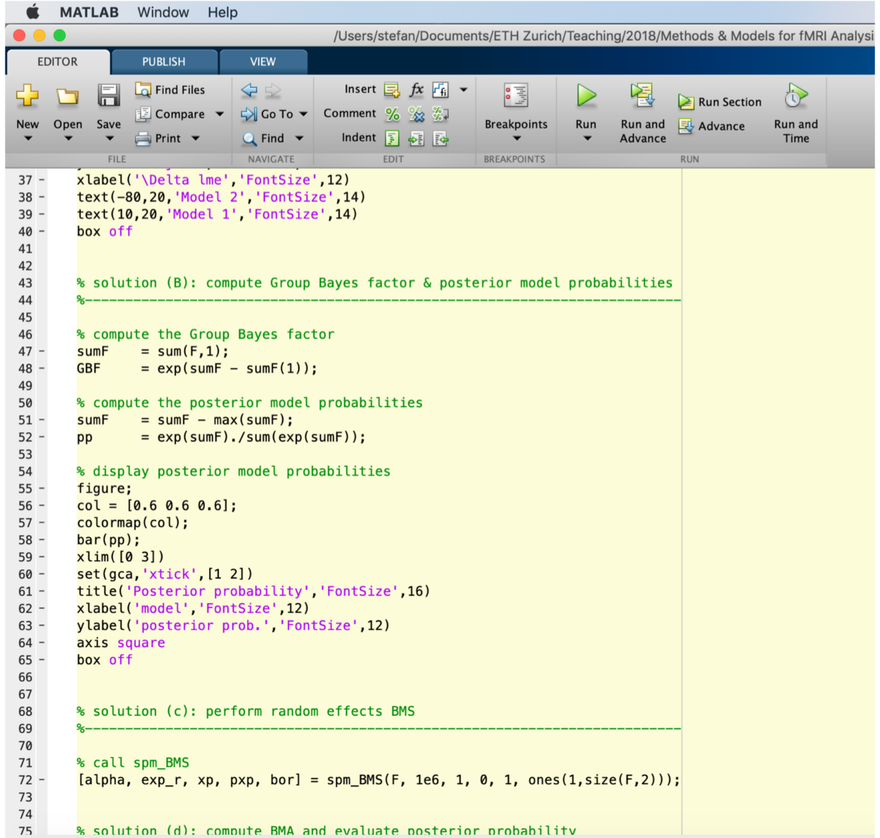


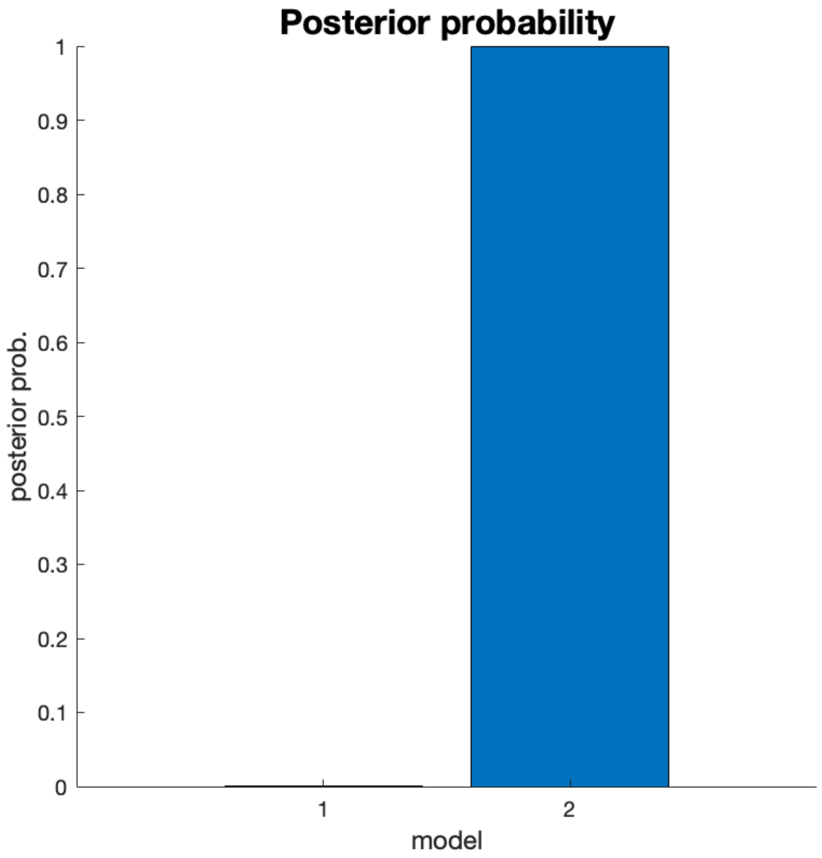

```matlab
% This exercise illustrates how to implement fixed-effects and random-
% effects Bayesian model selection (BMS), and how to perform Bayesian
% model averaging (BMA)
%

% results array for log model evidence
F = NaN(20,2);

% get the log model evidences (approximated by negative free energies)
for n = 1:20
    if ( n < 10 )
        n_text = ['0' num2str(n)];
    else
        n_text = num2str(n);
    end
    for m = 1:2
        temp = load([pwd '/DCMs/DCM_model' num2str(m) '_' n_text '.mat']);
        F(n,m) = temp.DCM.F;
    end
end


% solution (a): compute subject-sepcific Bayes factors
%-----------------------------------------------------------------------------

% compute Bayes factor
BF = exp(F(:,1)-F(:,2));

figure;
col = [0.7 0.7 0.7];
barh(F(:,1)-F(:,2), 'facecolor', col, 'edgecolor', 'none');
ylim([0 size(F,1)+1])
title('log model evidence difference','FontSize',16)
ylabel('subject','FontSize',12)
xlabel('\Delta lme','FontSize',12)
text(-80,20,'Model 2','FontSize',14)
text(10,20,'Model 1','FontSize',14)
box off
```



Try to set up a Fixed-Effects BMS analysis

(2) Implement fixed-effects Bayesian model selection:





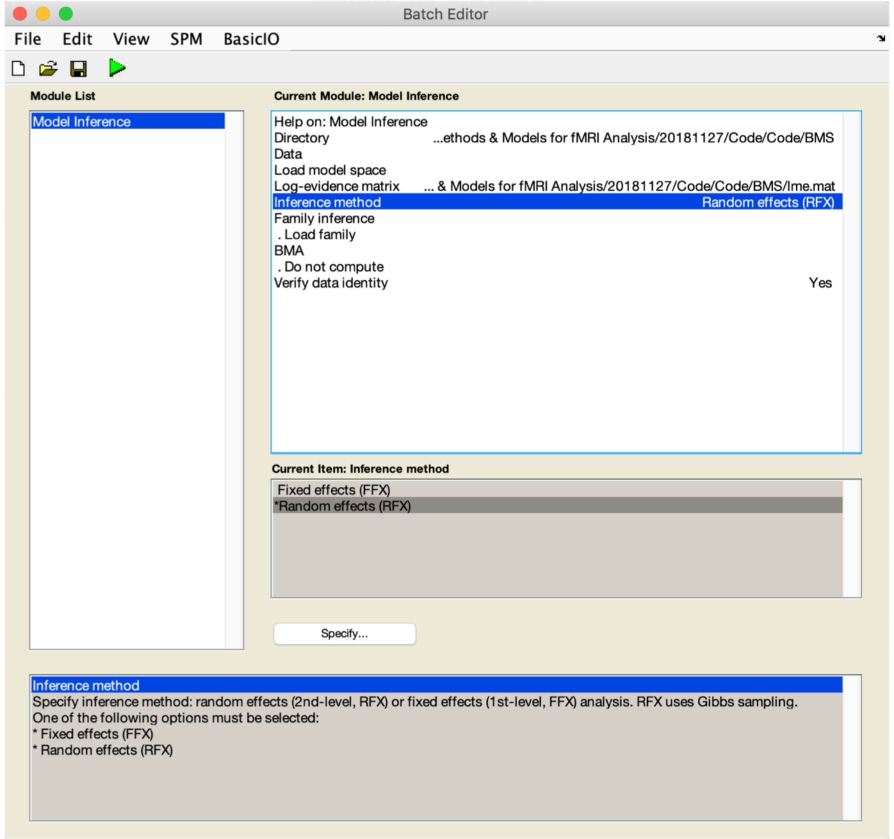
Try to set up a Random-Effects BMS analysis

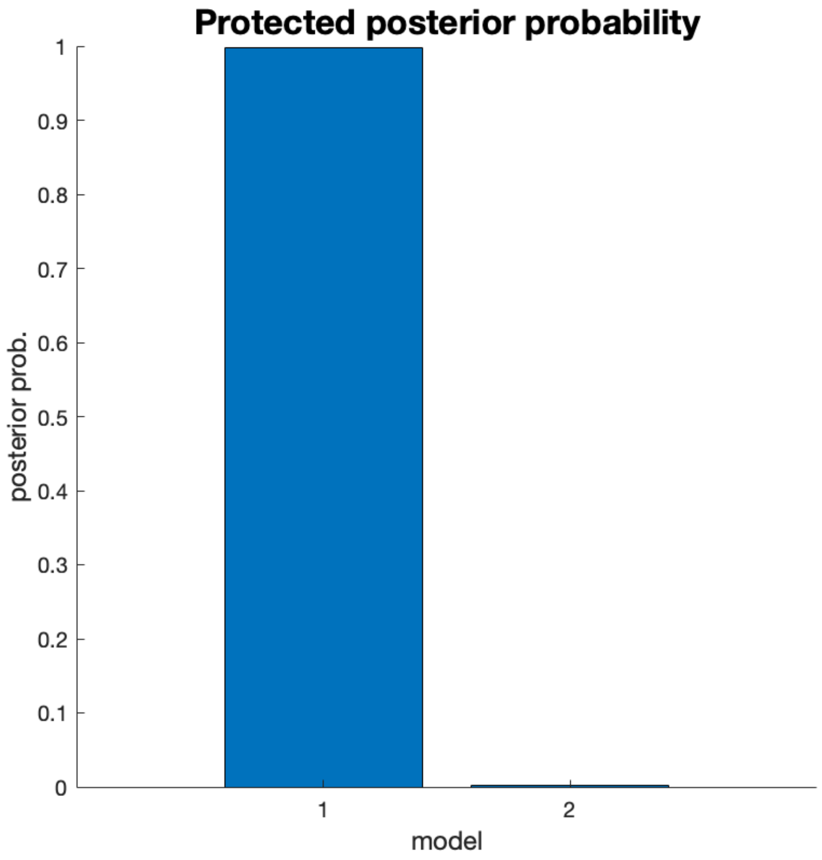
Implement the following analysis:

(1) Perform random-effects Bayesian model selection by computing the protected exceedance probability. You can either use the SPM GUI or write your own code (*Hint: when writing your own code, use to function "spm_BMS.m"*). What do you notice when comparing the results with the FFX-BMS results from before?

TRY TO SET UP A RANDOM-EFFECTS BMS ANALYSIS

(1) Implement random-effects Bayesian model selection:





Try to set up a Random-Effects BMA analysis

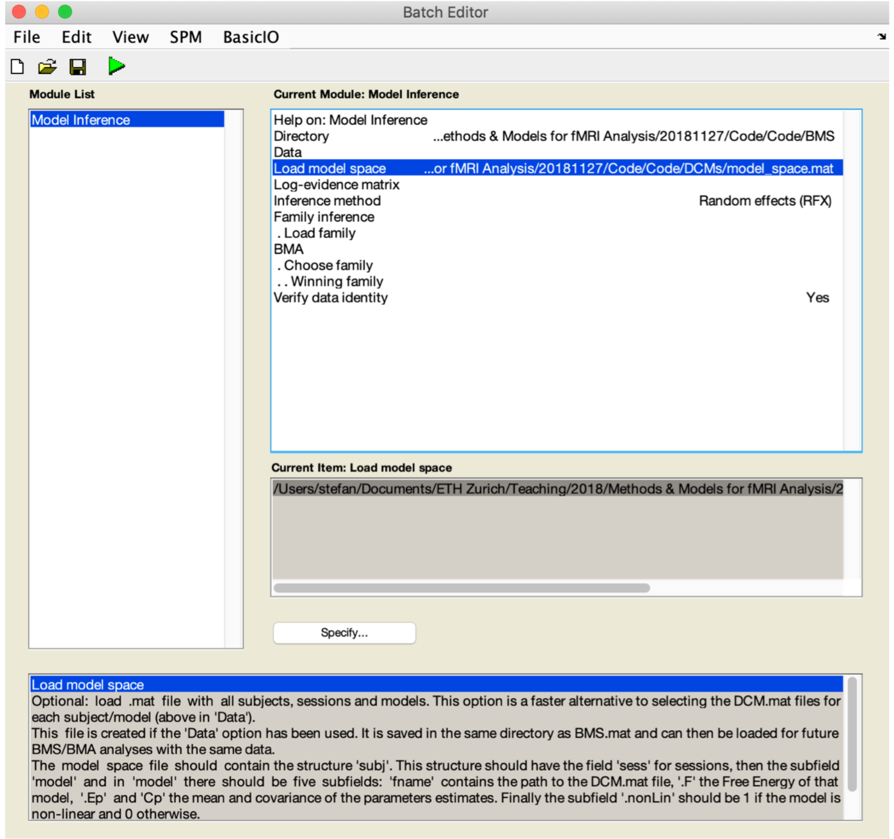
Implement the following analysis:

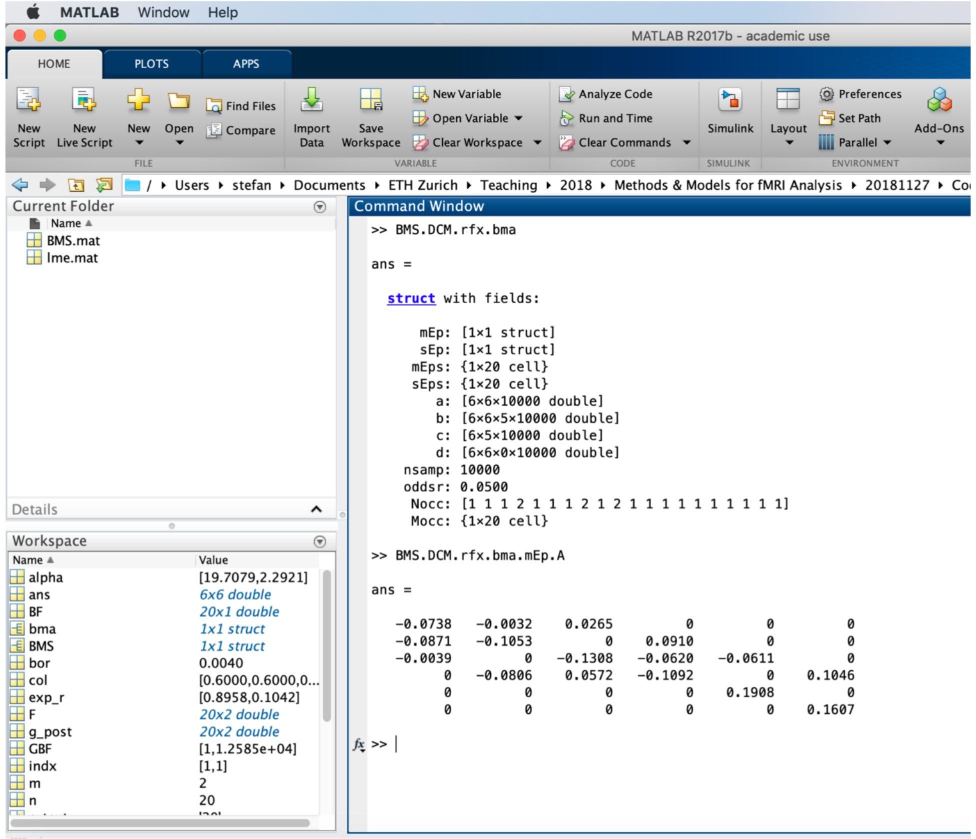
(1) Perform random-effects Bayesian model averaging and compute the BMA posterior over model parameters. You can either use the SPM GUI or write your own code (*Hint: when writing your own code, use the functions "spm_BMS_gibbs.m" and "spm_dcm_bma.m"*).

(2) Bonus: Test for which of the parameters, the connection strength of an "average" subject differs from zero. For this, use the group-level BMA results (*which are stored in mEp and sEp*) from step (1) to compute, for each connection, the posterior probability that the connection strength differs from zero *(Hint: the function "spm_Ncdf.m" will be helpful for that purpose).*

TRY TO SET UP A RANDOM-EFFECTS BMA ANALYSIS

(1) Implement random-effects Bayesian model averaging:

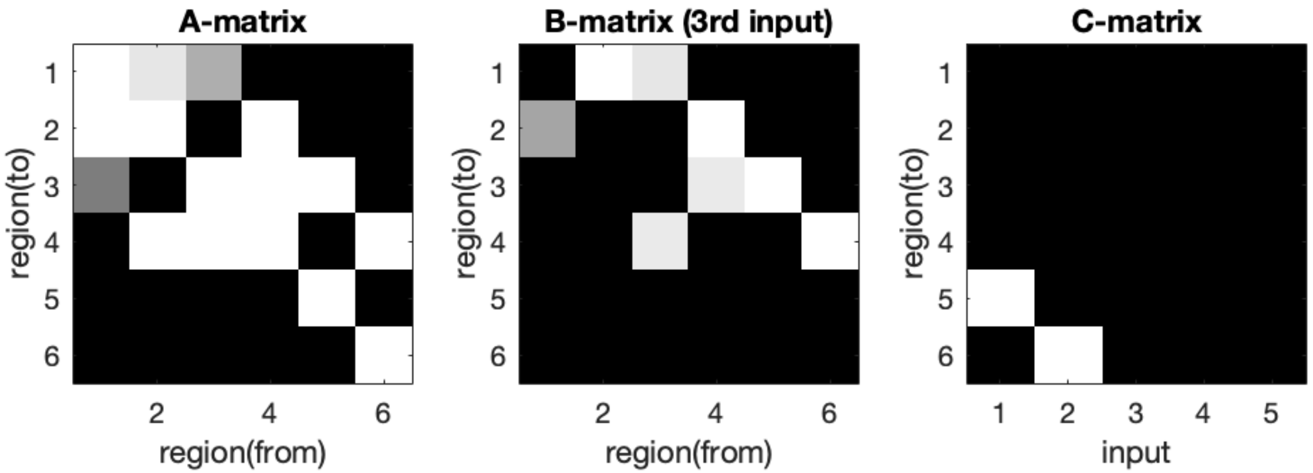




Try to set up a Random-Effects BMA analysis

(2) Evaluate the posterior probability of each connection:

```
bma = spm_dcm_bma(g_post,indx,subj,nsamp,odds_ratio);

% get posterior mean and covariance
Ep  = bma.mEp;
Cp  = spm_unvec((spm_vec(bma.sEp)).^2,bma.sEp);

% get prior means
model1 = load([pwd '/DCMs/DCM_model1_01.mat']);
model2 = load([pwd '/DCMs/DCM_model2_01.mat']);
DCM.a = model1.DCM.a | model2.DCM.a;
DCM.b = model1.DCM.b | model2.DCM.b;
DCM.c = model1.DCM.c | model2.DCM.c;
DCM.d = zeros(size(DCM.a,1),size(DCM.a,2),0);
[pE,~,~] = spm_dcm_fmri_priors(DCM.a,DCM.b,DCM.c,DCM.d);
T   = full(spm_vec(pE));

% evaluate the posterior probabilities of model parameters
Pp  = spm_unvec(1 - spm_Ncdf(T,abs(spm_vec(Ep)),spm_vec(Cp)),Ep);

% plot the posterior probabilities for A-matrix
figure,
subplot(1,3,1)
imagesc(Pp.A)
colormap('gray')
axis square
xlabel('region(from)')
ylabel('region(to)')
title('A-matrix')
caxis([0 1])

% plot the posterior probabilities for B-matrix (3rd input)
subplot(1,3,2)
imagesc(Pp.B(:,:,3))
colormap('gray')
axis square
title('B-matrix (3rd input)')
xlabel('region(from)')
ylabel('region(to)')
caxis([0 1])
```



Questions

Regarding Lecture or Practical Session

?

Thank you for your Attention !

Stefan Frässle, PhD
Translational Neuromodeling Unit (TNU)
University of Zurich & ETH Zurich

Email: stefanf@biomed.ee.ethz.ch
Phone: +41 44 634 91 14
@stefan_fraessle