# RESTING STATE ANALYSIS with TIMESERIES (TC) and FC

FOR EACH PARTICIPANT

Use dcm2niigui to convert dicom to nifti\_gz

Use Melodic on rsMRI timeseries (default parameters)

Outputs:

ICA components

filtered\_func\_data.nii.gz

transforms (reg/)

*Generate aal in native space (must use nearest neighbour)*

Use Applyxfm, using reg transforms with aal 1mm3 as input: ~/aal/aal.nii.gz

/usr/local/fsl/bin/flirt -in /users/mortenk/aal/aal.nii.gz -applyxfm -init /users/mortenk/data/s0007/s0007.ica/reg/standard2example\_func.mat -out /users/mortenk/data/s0007/s0007.ica/aal\_s0007 -paddingsize 0.0 -interp nearestneighbour -ref /users/mortenk/data/s0007/s0007.ica/example\_func.nii.gz

mkdir aal

mv aal\_s0007.nii.gz aal/

*Create masks of each region*

cd aal

~/perl/make4Dmasks.pl aal\_s0007.nii.gz

#!/usr/bin/perl

# creates 116 binary masks from aal file provided by first argument

# include cerebellum regions 91-116

die "usage: make4Dmasks.pl aal\n" if @ARGV <1;

$aal=$ARGV[0];

for ($t=1; $t<117; $t++){

print("fslmaths $aal -thr $t -uthr $t -bin $t\n");

system("fslmaths $aal -thr $t -uthr $t -bin $t");

}

*Create timecourses from filtered\_func\_data for each region*

cd ..

~/perl/maketimecourses\_rev.pl filtered\_func\_data tc

#!/usr/bin/perl

# creates 90 timecourses from the 90 aal binary masks on the filtered data

# NOTE:

# execute from melodic main directory

# .. assumes that aal binary masks are in subdir aal/

# and named 1...90.nii.gz

die "usage: maketimecourses\_rev.pl datafile tcdir\n" if @ARGV <2;

$data=$ARGV[0];

$tcdir=$ARGV[1];

# make timecourse directory

system("mkdir $tcdir");

for ($t=1; $t<91; $t++){

$arg="fslmeants -i $data -m aal/$t -o ".$tcdir."/$t.txt --transpose";

print($arg."\n");

system($arg);

}

*Concatenate this into one vector with 90x140*

cd tc

~/perl/makevector.pl

mv total.txt total\_s0007.txt

This has been automatized in processall.pl

#!/usr/bin/perl

# matching experiment

# script to process AAL timecourses for all 30 sessions for participants

# needs to have an aal dir, with AAL in the right space, using make4Dmasks.pl

@sess=("Movie","Scram","Noise");

$s=1;

for ($r=0; $r<3; $r++){

for ($t=1; $t<31; $t++){

$dataset="Sub".$s."\_".$sess[$r]."\_session".$t;

$tc="tc\_".$sess[$r]."\_".$t;

#process the timecourses

$arg="~/Documents/MATLAB/perl/maketimecourses\_rev.pl $dataset $tc";

print $arg."\n";

system($arg);

# make timecourse vectors

$arg="cd $tc; ~/Documents/MATLAB/perl/makevector.pl; mv total.txt ../total\_".$tc.".txt";

print $arg."\n";

system($arg);

}

}

Open **Matlab**, (example for processing single participant with AAL116):

>> m= importdata('total\_s0007.txt');

>> fc\_s0007= LR\_version\_symm116(make\_fcz116(m));

>> imagesc(fc\_s0007)

>> save('s0007.mat', 'fc\_s0007', 'm')

Alternatively

>> tc\_s0007= importdata('total\_s0007.txt');

>> tcsymm\_s0007=LR\_tc\_version\_symm116(tc\_s0007);

>> fc\_s0007= make\_fcz116(tcsymm\_s0007));

>> imagesc(fc\_s0007)

>> save('s0007.mat', 'fc\_s0007', 'm')

To reduce to AAL76 (only cortical areas) use:

>> excl76=[19:21,36:39, 46:71, 78:81, 96:98];

>> SC76=SC116;

>> SC76(excl76,:)=[];

>> SC76(:,excl76)=[];

>> figure;imagesc(SC76)

To reduce to AAL90 (only cortical and subcortical areas) use:

>> excl90=[46:71];

>> SC90=SC116;

>> SC90(excl90,:)=[];

>> SC90(:,excl90)=[];

>> figure;imagesc(SC90)

Make timecourses symmetrical from non-symmetrical AAL116:

tc76symm\_s0003=make\_tc\_symm76(tc\_s0003);

tc76symm\_s0004=make\_tc\_symm76(tc\_s0004);

tc76symm\_s0005=make\_tc\_symm76(tc\_s0005);

tc76symm\_s0006=make\_tc\_symm76(tc\_s0006);

tc76symm\_s0007=make\_tc\_symm76(tc\_s0007);

tc76symm\_s0011=make\_tc\_symm76(tc\_s0011);

tc76symm\_s0016=make\_tc\_symm76(tc\_s0016);

tc76symm\_s0017=make\_tc\_symm76(tc\_s0017);

tc76symm\_s0018=make\_tc\_symm76(tc\_s0018);

tc76symm\_s0020=make\_tc\_symm76(tc\_s0020);

tc76symm\_s0021=make\_tc\_symm76(tc\_s0021);

tc76symm\_s0026=make\_tc\_symm76(tc\_s0026);

tc76symm\_s0027=make\_tc\_symm76(tc\_s0027);

tc76symm\_s0028=make\_tc\_symm76(tc\_s0028);

tc76symm\_s0029=make\_tc\_symm76(tc\_s0029);

tc76symm\_s0030=make\_tc\_symm76(tc\_s0030);

tc90symm\_s0003=make\_tc\_symm90(tc\_s0003);

tc90symm\_s0004=make\_tc\_symm90(tc\_s0004);

tc90symm\_s0005=make\_tc\_symm90(tc\_s0005);

tc90symm\_s0006=make\_tc\_symm90(tc\_s0006);

tc90symm\_s0007=make\_tc\_symm90(tc\_s0007);

tc90symm\_s0011=make\_tc\_symm90(tc\_s0011);

tc90symm\_s0016=make\_tc\_symm90(tc\_s0016);

tc90symm\_s0017=make\_tc\_symm90(tc\_s0017);

tc90symm\_s0018=make\_tc\_symm90(tc\_s0018);

tc90symm\_s0020=make\_tc\_symm90(tc\_s0020);

tc90symm\_s0021=make\_tc\_symm90(tc\_s0021);

tc90symm\_s0026=make\_tc\_symm90(tc\_s0026);

tc90symm\_s0027=make\_tc\_symm90(tc\_s0027);

tc90symm\_s0028=make\_tc\_symm90(tc\_s0028);

tc90symm\_s0029=make\_tc\_symm90(tc\_s0029);

tc90symm\_s0030=make\_tc\_symm90(tc\_s0030);

Create average FC76 and FC90

fc76\_s0003=make\_fcz76(tc76symm\_s0003);

fc76\_s0004=make\_fcz76(tc76symm\_s0004);

fc76\_s0005=make\_fcz76(tc76symm\_s0005);

fc76\_s0006=make\_fcz76(tc76symm\_s0006);

fc76\_s0007=make\_fcz76(tc76symm\_s0007);

fc76\_s0011=make\_fcz76(tc76symm\_s0011);

fc76\_s0016=make\_fcz76(tc76symm\_s0016);

fc76\_s0017=make\_fcz76(tc76symm\_s0017);

fc76\_s0018=make\_fcz76(tc76symm\_s0018);

fc76\_s0020=make\_fcz76(tc76symm\_s0020);

fc76\_s0021=make\_fcz76(tc76symm\_s0021);

fc76\_s0026=make\_fcz76(tc76symm\_s0026);

fc76\_s0027=make\_fcz76(tc76symm\_s0027);

fc76\_s0028=make\_fcz76(tc76symm\_s0028);

fc76\_s0029=make\_fcz76(tc76symm\_s0029);

fc76\_s0030=make\_fcz76(tc76symm\_s0030);

fc76\_3=cat(3, fc76\_s0003, fc76\_s0004,fc76\_s0005,fc76\_s0006,

fc76\_s0007,fc76\_s0011, fc76\_s0016,fc76\_s0017, fc76\_s0018,

fc76\_s0020,fc76\_s0021,fc76\_s0026,fc76\_s0027, fc76\_s0028,

fc76\_s0029,fc76\_s0030);

fc76\_mean=mean(fc76\_3,3);

imagesc(fc76\_mean)

fc90\_s0003=make\_fcz90(tc90symm\_s0003);

fc90\_s0004=make\_fcz90(tc90symm\_s0004);

fc90\_s0005=make\_fcz90(tc90symm\_s0005);

fc90\_s0006=make\_fcz90(tc90symm\_s0006);

fc90\_s0007=make\_fcz90(tc90symm\_s0007);

fc90\_s0011=make\_fcz90(tc90symm\_s0011);

fc90\_s0016=make\_fcz90(tc90symm\_s0016);

fc90\_s0017=make\_fcz90(tc90symm\_s0017);

fc90\_s0018=make\_fcz90(tc90symm\_s0018);

fc90\_s0020=make\_fcz90(tc90symm\_s0020);

fc90\_s0021=make\_fcz90(tc90symm\_s0022);

fc90\_s0021=make\_fcz90(tc90symm\_s0021);

fc90\_s0026=make\_fcz90(tc90symm\_s0026);

fc90\_s0027=make\_fcz90(tc90symm\_s0027);

fc90\_s0028=make\_fcz90(tc90symm\_s0028);

fc90\_s0029=make\_fcz90(tc90symm\_s0029);

fc90\_s0030=make\_fcz90(tc90symm\_s0030);

fc90\_3=cat(3, fc90\_s0003, fc90\_s0004,fc90\_s0005,fc90\_s0006,

fc90\_s0007,fc90\_s0011, fc90\_s0016,fc90\_s0017, fc90\_s0018,

fc90\_s0020,fc90\_s0021,fc90\_s0026,fc90\_s0027, fc90\_s0028,

fc90\_s0029,fc90\_s0030);

fc90\_mean=mean(fc90\_3,3);

imagesc(fc90\_mean)

function [new]=LR\_tc\_version\_symm116(C)

% returns a symmetrical LR version of AAL timecourse matrix (116 x n)

if (size(C,1)==116)

odd=[1:2:116];

even=sort([2:2:116],'descend');

new=zeros(116,size(C,2));

new(1:58,:)=C(odd,:);

new(59:116,:)=C(even,:);

else

error ('Matrix is not a AAL 116 timecourse file');

end;

end

function [fcz]=make\_fcz(old)

% function to create z score version of AAL correlation matrix

% and use the Fisher transform to generate z score matrix

% nb: diagonal is set to zero

fc=zeros(90,90);

%make pairwise correlations

for x=1:89,

for y=x+1:90,

[fc(x,y),p]=corr(old(x,:)',old(y,:)');

end;

end;

% create other half of matrix

fc\_half=fc;

for x=1:89,

for y=x+1:90,

fc(y,x)=fc(x,y);

end;

end;

% Fisher transform to z

fcz=.5.\*log((1+fc)./(1-fc));

MAKING AVERAGE MATRIX OVER ALL PARTICIPANTS (16 in this example):

>> fc3=cat(3, fc\_s0003, fc\_s0004,fc\_s0005,fc\_s0006,fc\_s0007,fc\_s0011, fc\_s0016,fc\_s0017, fc\_s0018,fc\_s0020,fc\_s0021,fc\_s0026,fc\_s0027, fc\_s0028,fc\_s0029,fc\_s0030);

>> fcm=mean(fc3,3);

>> imagesc(fcm)

>> save('fc16','fc3','fcm', 'fc\_s0003', 'fc\_s0004','fc\_s0005','fc\_s0006','fc\_s0007', 'fc\_s0011','fc\_s0016','fc\_s0017','fc\_s0018','fc\_s0020','fc\_s0021','fc\_s0026', 'fc\_s0027','fc\_s0028','fc\_s0029','fc\_s0030');

# 