

In [6]:

```
addpath('/rds/general/user/es2814/home/WORK/work/code/c3nl_fusion')

/rds/general/user/es2814/home/WORK/work/data/12TASKS
```

## Mine brain metrics

In [14]:

```
% start by mining dFC
modelDir = '~/work/data/12TASKS';
cd '~/work/data/12TASKS')
for r = {'DM_MD.nii', 'CT_200.nii', 'intersectionROI.nii'}
cmd = sprintf('%s "maxNumCompThreads(1);rehash toolboxcache;run.epi2dFC($PBS_ARRAY_INDEX,%s,%s);exit"',...
'matlab -nodisplay -nodesktop -nosplash -r', '~/work/data/12TASKS', sprintf(
'%s', r{1}));
jid = c3nl.pbatch(struct('mem',5,'cpu',1,'array','1-60','time','00:60:00','workdir',modelDir,'cmd',{cmd}));
end
```

In [10]:

```
% Then mine voxel-wise activation
modelDir = '~/work/data/12TASKS'
Bepi = c3nl.select('pth',[modelDir,filesep,'POST'],'name','*.nii.gz');
roiSet = {'DM_MD.nii', 'CT_200.nii', 'intersectionROI.nii'}
for r=roiSet
[L,G] = apply.interpolator(sprintf("%s/ROI/%s",modelDir,r{1}),Bepi{1});
[X,Y,S] = mine.Boldbetas(L>0,[],modelDir,0);
D = new.folder([], [modelDir, '/ML/Data'], {'BAvx'});
[~,roiname] = fileparts(roiSet);
save(sprintf('%s/%s.mat',D.BAvx,roiname), 'X', 'Y', 'S', '-v7.3');
end
```

In [12]:

```
% And ROI mean activation
roiSet = '/home/es2814/work/data/12TASKS/ROI/DM_MD.nii'
Bepi = c3nl.select('pth',[modelDir,filesep,'POST'],'name','*.nii.gz');
for r=roiSet
[L,G] = apply.interpolator(sprintf("%s/ROI/%s",modelDir,r{1}),Bepi{1}, 'reslice');
[X,Y,S,BA] = mine.Boldbetas(convert.label2Dummy(L),atlas.toTable(L,G),modelDir,0);
D = new.folder([], [modelDir, '/ML/Data'], {'BA'});
[~,roiname] = fileparts(roiSet);
save(sprintf('%s/%s.mat',D.BA,roiname), 'X', 'Y', 'S', 'BA', '-v7.3');
end
```

In [3]:

```
% Finally mine PPI
modelDir = '~/work/data/12TASKS';
for r = {'DM_MD.nii', 'CT_200.nii', 'intersectionROI.nii'} % 'CT_200.nii', 'intersectionROI.nii'
cmd = sprintf('%s "maxNumCompThreads(1);rehash toolboxcache;run.epi2ppi($PBS_ARRAY_INDEX,%s,%s);exit"',...
'matlab -nodisplay -nodesktop -nosplash -r', '' '~/work/data/12TASKS'', sprintf(
'' '%s'', r{1}));
jid = c3nl.pbatch(struct('mem',5,'cpu',1,'array','1-60','time','00:60:00','workdir',modelDir,'cmd',{cmd}));
end
```

submitting

## Harvest brain metrics into a database

In [20]:

```
for r = {'DM_MD', 'CT_200.nii', 'intersectionROI.nii'}
D = new.folder([], [modelDir, '/ML/Data'], {'PPI'});
fn = c3nl.select('pth', [modelDir, '/PPI'], 'name', sprintf('%s*', r{1}), 'output', 'char');
L = load.vol(c3nl.select('pth', [modelDir, '/ROI'], 'name', sprintf('%s*.nii', roiname), 'output', 'char'));
L = atlas.reLabel(L);
n = numel(unique(L))-1;
[X,Xz,Y,S,PPI]= mine.PPIbetas(fn,n);
save(sprintf('%s/%s.mat', D.PPI, roiname), 'X', 'Y', 'S', 'PPI', '-v7.3');
end
```

In [ ]:

```
modelDir = '~/work/data/12TASKS';
for r = {'DM_MD', 'CT_200.nii', 'intersectionROI.nii'} % 'CT_200.nii', 'intersectionROI.nii'
D = new.folder([], [modelDir, '/ML/Data'], {'dFC'});
fn = c3nl.select('pth', [modelDir, '/dFC'], 'name', sprintf('%s*', r{1}), 'output', 'char');
L = load.vol(c3nl.select('pth', [modelDir, '/ROI'], 'name', sprintf('%s*.nii', r{1}), 'output', 'char'));
L = atlas.reLabel(L);
n = numel(unique(L))-1;
[X,Y,S,dFC]= mine.dFCbetas(fn,n);
save(sprintf('%s/%s.mat', D.dFC, r{1}), 'X', 'Y', 'S', 'dFC', '-v7.3');
end
```

the MATLAB function has been cancelled

## Model classification 12way models One-vs-all (FIR & voxel -> stack)

In [ ]:

```

%% Model voxel-wise and FIR FC in a 12-way stacked model
N = 100; % 100 permutations
mc = 1; % 1 repetition (as each model takes approx 20 hours to finish)
in = '/home/es2814/work/data/12TASKS/ML/Data';
out = '/home/es2814/work/data/12TASKS/ML';
roiSet = {'DM_MD', 'CT_200', 'intersectionROI'}; % 'CT_200', 'intersectionROI'
%m = {'BAVx', 'BA', 'PPI', 'dFC'}; % which metrics
m = {'BAVx', 'BA', 'PPI', 'dFC'}; % which metrics

for ii=1:numel(roiSet);
    roiname = roiSet{ii};
    D=new.folder([],in,m);
    job.cmd = {};
    for jj=1:numel(m)
        dataPath{jj} = sprintf('%s/%s.mat',D.(m{jj}),roiname);
    end
    cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCompT
hreads(1);run.wrapStackedMCModels($PBS_ARRAY_INDEX,%i,%i,'%s',{'%s'},'%s',
{'%s'})';exit"',N,mc,['STACK_FIR_VX',roiname],strjoin(dataPath,'',''),out,str
join(m,'',''));
    job.cmd = {};
    job.cmd{1} = struct('cmd',{cmd},...
                        'jobname','12WayModels',...
                        'workdir','/home/es2814/WORK/work/data/12TASKS',...
                        'output','/home/es2814/WORK/work/data/12TASKS',...
                        'mem',15,...
                        'array',sprintf('1-%i',N),...
                        'cpu',1,'time','45:50:00');
    jid = c3nl.pbatch(job.cmd{1});
end

```

the MATLAB function has been cancelled

## Model classification 12way models One-vs-all (All metrics -> stack)

In [1]:

```

%% Model voxel-wise and FIR FC in a 12-way stacked model
N = 100; % 100 permutations
mc = 1; % 1 repetition (as each model takes approx 20 hours to finish)
in = '/home/es2814/work/data/12TASKS/ML/Data';
out = '/home/es2814/work/data/12TASKS/ML';
roiSet = {'DM_MD', 'CT_200', 'intersectionROI'}; % 'CT_200', 'intersectionROI'
%m = {'BAVx', 'BA', 'PPI', 'dFC'}; % which metrics

m = {'BAVx', 'BA', 'PPI', 'dFC'}; % which metrics

for ii=1:numel(roiSet);
    roiname = roiSet{ii};
    D=new.folder([],in,m);
    job.cmd = {};
    for jj=1:numel(m)
        dataPath{jj} = sprintf('%s/%s.mat',D.(m{jj}),roiname);
    end
    cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCompT
hreads(1);run.wrapStackedMCModels($PBS_ARRAY_INDEX,%i,%i,'%s',{'%s'},'%s',
{'%s'});exit"',N,mc,['STACK_Full_',roiname],strjoin(dataPath,'',''),out,strj
oin(m,'',''));
    job.cmd = {};
    job.cmd{1} = struct('cmd',{cmd},...
                        'jobname','12WayModels',...
                        'workdir','/home/es2814/work/data/12TASKS',...
                        'output','/home/es2814/work/data/12TASKS',...
                        'mem',15,...
                        'array',sprintf('1-%i',N),...
                        'cpu',1,'time','75:50:00');
    jid = c3nl.pbatch(job.cmd{1});
end

```

submitting  
submitting  
submitting

## Model classification 12way models One-vs-one

In [25]:

```
%% Model all pairwise tasks in one-vs-one scheme
N = 100; % 1000 permutations
mc = 1; % 1 repetition (as each model takes approx 20 hours to finish)
in = '/home/es2814/work/data/12TASKS/ML/Data';
out = '/home/es2814/work/data/12TASKS/ML';
roiSet = {'DM_MD', 'CT_200', 'intersectionROI'}
m = {'BAvx', 'BA', 'PPI', 'dFC'}; % which metrics
for ii=1:numel(roiSet);
    roiname = roiSet{ii};
    fn = c3nl.select('pth',in,'name',[roiSet{ii} '*']); % dataPath
    fn = fn(c3nl.strDetect(fn,[m{1} '|' m{2}]));
    job.cmd = {};
    cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCompT
hread(1);run.wrapEcoStackBinModels($PBS_ARRAY_INDEX,%i,%i,'%s',{ '%s'},'%
s',{ '%s'});exit"',N,mc,['STACK_FIRVX_2way',roiSet{ii}],strjoin(fn,' ',''),ou
t,strjoin(m,' ',''));
    job.cmd = {};
    job.cmd{1} = struct('cmd',{cmd},...
                        'jobname','2WayModels',...
                        'workdir','/home/es2814/work/data/12TASKS',...
                        'output','/home/es2814/work/data/12TASKS',...
                        'mem',15,...
                        'array',sprintf('1-%i',N),...
                        'cpu',1,'time','22:50:00');
    jid = c3nl.pbatch(job.cmd{1});
end
```

m =

1x2 cell array

    {'BAvx'}    {'dFC'}

submitting

## Individual differences using kfold predict

In [ ]:

```
%roiSet = {'CT_200','L2_200_inter','L2_200_Diff'};
roiSet = {'DM_MD','CT_200','intersectionROI'};%
metric = {'BAvx/','dFC/'};
for ii=1:numel(roiSet)
cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCompThrea
ds(1);run.wrapIndividualDiff($PBS_ARRAY_INDEX,'indi_diff_%s',''%s',''%s')');
exit"',roiSet{ii},'/home/es2814/work/data/12TASKS/',strjoin(metric,'',''))
job.cmd = {};

job.cmd{1} = struct('cmd',{cmd}},...
    'jobname','individualDiff',...
    'workdir','/home/es2814/work/data/12TASKS',...
    'output','/home/es2814/work/data/12TASKS',...
    'mem',16,...
    'array',sprintf('1-%i',60),...
    'cpu',1,'time','10:50:00');
jid = c3nl.pbatch(job.cmd{1});

end
%
```

In [1]:

```
%roiSet = {'CT_200','L2_200_inter','L2_200_Diff'};
cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCompThrea
ds(1);run.IndividualDiff($PBS_ARRAY_INDEX,'%s',''%s',''%s')';exit"', 'CT_200',
'/home/es2814/WORK/work/data/12TASKS/','dFC/');
job.cmd = {};
job.cmd{1} = struct('cmd',{cmd}},...
    'jobname','individualDiff',...
    'workdir','/home/es2814/WORK/work/data/12TASKS',...
    'output','/home/es2814/WORK/work/data/12TASKS',...
    'mem',16,...
    'array',sprintf('1-%i',60),...
    'cpu',1,'time','10:50:00');
jid = c3nl.pbatch(job.cmd{1});

%
```

submitting

## Compare FIR and PPI models

In [27]:

```
N = 100; % 100 permutations
mc = 1; % 1 repetition (as each model takes approx 20 hours to finish)
in = '/home/es2814/work/data/12TASKS/ML/Data';
out = '/home/es2814/work/data/12TASKS/ML';
roiSet = {'DM_MD', 'CT_200', 'intersectionROI'}; % 'CT_200', 'intersectionROI'
%m = {'BAVx', 'BA', 'PPI', 'dFC'}; % which metrics

m = {'PPI', 'dFC'}; % which metrics

for ii=1:numel(roiSet);
    roiname = roiSet{ii};
    D=new.folder([],in,m);
    job.cmd = {};
    for jj=1:numel(m)
        dataPath{jj} = sprintf('%s/%s.mat',D.(m{jj}),roiname);
    end
    cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCompT
hreads(1);run.wrapStackedMCModels($PBS_ARRAY_INDEX,%i,%i,'%s',{'%s'},'%s',
{'%s'});exit",N,mc,['STACK_FIR_PPI',roiname],strjoin(dataPath,' ',''),out,st
rjoin(m,' ','')));
    job.cmd = {};
    job.cmd{1} = struct('cmd',{cmd},...
                        'jobname','12WayModels',...
                        'workdir','/home/es2814/work/data/12TASKS',...
                        'mem',15,...
                        'array',sprintf('1-%i',N),...
                        'cpu',1,'time','45:50:00');
    jid = c3nl.pbatch(job.cmd{1});
end
```

submitting  
submitting  
submitting

## Decode meta labels

In [28]:

```
%% Model all meta labels
N = 100; % 1000 permutations
in = '/home/es2814/work/data/12TASKS/ML/Data';
out = '/home/es2814/work/data/12TASKS/ML';
roiSet = {'DM_MD'}; % 'CT_200', 'intersectionROI'
m = {'BA', 'BAvx', 'PPI', 'dFC'}; % which metrics

for ii=1:numel(roiSet);
    roiname = roiSet{ii};
    D=new.folder([],in,m);
    job.cmd = {};
    for jj=1:numel(m)
        dataPath{jj} = sprintf('%s/%s.mat',D.(m{jj}),roiname);
        cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "maxNumCo
mpThreads(1);run.DecodingDifferentLabels($PBS_ARRAY_INDEX,%i,'%s','%s','%
s');exit"',N,['Response_',roiname,'_',m{jj}],dataPath{jj},out);
        job.cmd = {};
        job.cmd{1} = struct('cmd',{cmd},...
                            'jobname','12WayModels',...
                            'workdir','/home/es2814/work/data/12TASKS',...
                            'mem',15,...
                            'array',sprintf('1-%i',N),...
                            'cpu',1,'time','45:50:00');
        jid = c3nl.pbatch(job.cmd{1});
    end
end
```

submitting  
submitting  
submitting  
submitting

## Run permutation analysis for meta labels



```
N = 500; % 100 sampling
perm = 20 % number of permutation
% factorID
in = '/home/es2814/work/data/12TASKS/ML/Data';
out = '/home/es2814/work/data/12TASKS/ML';
roiSet = {'DM_MD'};%,'CT_200', 'intersectionROI'}
%m = {'BAvx','BA','PPI','dFC'};% which metrics

m = {'BAvx','BA','PPI','dFC'};% which metrics , 'dFC'

for jj=1:numel(m) % metric 4
    for k=1:4% factor id 4
        for ii=1:numel(roiSet);% roi set 1
            roiname = roiSet{ii};
            D=new.folder([],in,m);
            job.cmd = {};
            dataPath = sprintf('%s/%s.mat',D.(m{jj}),roiname);
            cmd = sprintf('matlab -nodisplay -nodesktop -nosplash -nojvm -r "max
NumCompThreads(1);run.permAsses($PBS_ARRAY_INDEX,%i,%i,'%s','%s','%s',%
i,'%s');exit"',N,perm,['Perm_VX',roiname],dataPath,out,k,m{jj});
            job.cmd = {};
            job.cmd{1} = struct('cmd',{cmd},...
                                'jobname','factor_perm',...
                                'workdir','/home/es2814/work/data/12TASKS',...
                                'mem',4,...
                                'array',sprintf('1-%i',N),...
                                'cpu',1,'time','7:50:00');
            jid = c3nl.pbatch(job.cmd{1});
        end
    end
end
```

100

[illegible]

## Collect results into table format

## 12way models One-vs-All results (FIR & Voxel-wise -> Stack)

In [36]:

```
Input = '/home/es2814/work/data/12TASKS/ML/results/';
output = '/home/es2814/work/data/12TASKS/ML/DataSets';
mkdir(output);
dn = c3nl.select('pth',Input,'name','STACK_FIR_VX*','type','d');
dn = dn(c3nl.strDetect(dn,'_MD|_200|onROI'));
DS_12way = table();
for ii=1:numel(dn)
    fn = c3nl.select('pth',dn{ii},'name','output*.mat');
    [~,name]=fileparts(dn{ii});
    for jj=1:numel(fn)
        tmp = load(fn{jj});
        DS_12way = [DS_12way;[table(repmat({name},5,1),'VariableNames',{'set'}),
tmp.output.perf(:,[1:3,5:end])]]];
    end
end
save(sprintf('%s/DS_12way.mat',output),'DS_12way');
```

## 2way models One-vs-one results (FIR & Voxel-wise -> Stack)

In [37]:

```
dn = c3nl.select('pth',Input,'name','STACK_FIRVX_2way*','type','d');
dn = dn(c3nl.strDetect(dn,'_MD|_200|onROI'));
DS_2way = table();
for ii=1:numel(dn)
    fn = c3nl.select('pth',dn{ii},'name','output*.mat');
    [~,name]=fileparts(dn{ii});
    for jj=1:numel(fn)
        tmp = load(fn{jj});
        DS_2way = [DS_2way;[table(repmat({name},height(tmp.output.perf),1),'VariableNames',{'set'}),tmp.output.perf]];
        % if mod(jj,50);fprintf('.');else;fprintf('\n');end
    end
end
save(sprintf('%s/DS_2way.mat',output),'DS_2way');
```

## meta labels classification

In [7]:

```
cd '/rdsgpfs/general/user/es2814/home/WORK/work/data/12TASKS/ML/results'
dn = c3nl.select('type','d');
dn = dn(c3nl.strDetect(dn,'Response'));
dn = dn(c3nl.strDetect(dn,'_MD|_200|onROI'));

DS.Resp = table();
DS.Psy = table();
DS.Domain = table();
DS.Input = table();
for ii=1:numel(dn)
    fn = c3nl.select('pth',dn{ii},'name','out*');
    [~,name] = fileparts(dn{ii});
    fprintf('\n%s-',name);
    name = {name;name};
    for jj=1:numel(fn)
        try
            tmp = load(fn{jj});
            tmp = tmp.output.perf;
            DS.Resp = [DS.Resp;[name,tmp.Resp]];
            DS.Psy = [DS.Psy;[name,tmp.Psy]];
            DS.Domain = [DS.Domain;[name,tmp.Domain]];
            DS.Input = [DS.Input;[name,tmp.Input]];
        end
        if ~mod(jj,50);fprintf('\n');else;fprintf('*'); end
    end
end

output = '/home/es2814/work/data/12TASKS/ML/DataSets';
save(sprintf('%s/DS_response.mat',output),'DS');
```

## meta labels permutation

In [4]:

```

cd '/rdsgpfs/general/user/es2814/home/WORK/work/data/12TASKS/ML/results'
dn = c3nl.select('type','d');
dn = dn(c3nl.strDetect(dn,'Perm'));
dn = dn(c3nl.strDetect(dn,'_200'));

DS.Psy = table();
DS.Resp = table();
DS.Domain = table();
DS.Input = table();
for ii=1:numel(dn)
    fn = c3nl.select('pth',dn{ii},'name','*out*');
    for jj=1:numel(fn)
        tmp = load(fn{jj});
        [~,name] = fileparts(fn{jj});
        name = strsplit(name,'_');
        n= height(tmp.perf);
        switch name{2}
            case '1';DS.Psy = [DS.Psy;[table(repmat(name(1),n,1),repmat(name(end),n,1),'VariableName',{'metric','iter'}),tmp.perf)];
            case '2';DS.Resp = [DS.Resp;[table(repmat(name(1),n,1),repmat(name(end),n,1),'VariableName',{'metric','iter'}),tmp.perf)];
            case '3';DS.Domain = [DS.Domain;[table(repmat(name(1),n,1),repmat(name(end),n,1),'VariableName',{'metric','iter'}),tmp.perf)];
            case '4';DS.Input = [DS.Input;[table(repmat(name(1),n,1),repmat(name(end),n,1),'VariableName',{'metric','iter'}),tmp.perf)];
        end
        if ~mod(jj,50);fprintf('\n');else;fprintf('*'); end
    end
end

output = '/home/es2814/work/data/12TASKS/ML/DataSets';
save(sprintf('%s/DS_permutation.mat',output),'DS');

```

In [ ]:

```

cd '/rdsgpfs/general/user/es2814/home/WORK/work/data/12TASKS/ML/results'
dn = c3nl.select('type','d');
dn = dn(c3nl.strDetect(dn,'STACK_FIR_PPI'));
FIR_PPI_12way = table();
for ii=1:numel(dn)
    fn = c3nl.select('pth',dn{ii},'name','output*.mat');
    [~,name]=fileparts(dn{ii});
    for jj=1:numel(fn)
        tmp = load(fn{jj});
        FIR_PPI_12way = [FIR_PPI_12way;[table(repmat({name},5,1),'VariableNames',{'set'}),tmp.output.perf(:,[1:3,5:end])]];
    end
end
output = '/home/es2814/work/data/12TASKS/ML/DataSets';
save(sprintf('%s/FIR_PPI_12way.mat',output),'FIR_PPI_12way');

```

In [ ]:

```

cd '/rdsgpfs/general/user/es2814/home/WORK/work/data/12TASKS/ML/results'
dn = c3nl.select('type','d');
dn = dn(c3nl.strDetect(dn,'/CT_|/DM_|/in'));
dn = dn(~c3nl.strDetect(dn,'.ipyn'))

IndividualDiff = table();
DS_results = table();
for ii=1:numel(dn)
    fn = c3nl.select('pth',dn{ii},'name','output*.mat');
    [~,name]=fileparts(dn{ii});
    for jj=1:numel(fn)
        tmp = load(fn{jj});
        subj = unique(tmp.out.fit.Subj);
        n = height(tmp.out.results);
        IndividualDiff = [IndividualDiff;[table(repmat({name},height(tmp.out.fit),1),'VariableNames',{'set'}),tmp.out.fit]];
        DS_results = [DS_results;[table(repmat({name},n,1),repmat({subj},n,1),'VariableNames',{'set','subj'}),tmp.out.results(:,[1,3:end])]];
    end
end
output = '/home/es2814/work/data/12TASKS/ML/DataSets';
save(sprintf('%s/IndividualDiff.mat',output),'DS_results','IndividualDiff');

```

In [3]:

```

in = '/rdsgpfs/general/user/es2814/home/WORK/work/data/12TASKS/ML/results/INDI_dFC/_CT_200';
fn = c3nl.select('pth',in,'name','output*.mat');
models = cell(numel(fn),3);

for ii=1:numel(fn)
    load(fn{ii})
    tmp = out.mdl.Trained;
    tmp0 = [];
    % merge kfold models for each task
    for k=1:12% for each task binary model
        tmp1=[];
        for jj=1:numel(tmp)% go over kfolds
            tmp1 = [tmp1;tmp{jj}.BinaryLearners{k}.Beta'];
        end
        tmp0{k} = mean(tmp1);
    end
    models{ii,1} = tmp0;
    models{ii,2} = out.results;
    models{ii,3} = out.yhat;
end

```

In [6]:

```
save('models.mat','models')
```

In [ ]: