

# PESTICA for AFNI v1.3

an overview of physio noise and PESTICA tutorial

PESTICA=Physiologic EStimation by Temporal ICA

Comments/questions/requests/bugs:

contact Erik Beall – [ebeall@gmail.com](mailto:ebeall@gmail.com)

Also see <http://www.nitrc.org/projects/pestica>

For use with pestica\_afni v1\_3.zip

# Overview of PESTICA

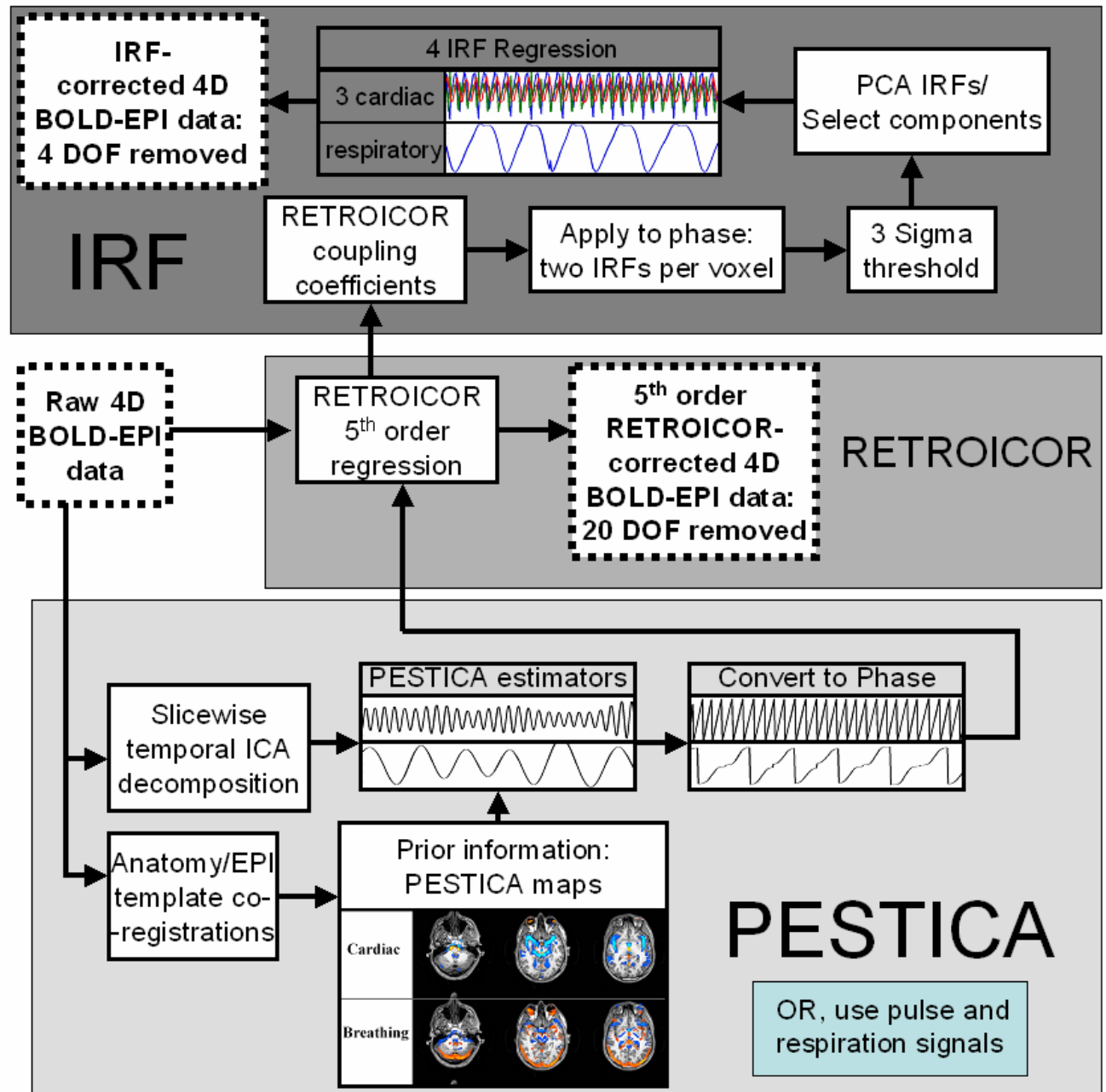
- Contains scripts and templates for
  - PESTICA cardiac and respiratory estimation
  - Physiologic noise correction
    - RETROICOR and IRF-RETROICOR, an improved noise regression method
      - *IRFRET removes less degrees of freedom (DOF)*
  - Quality assurance tool
- NOTE: requires AFNI, MATLAB and AFNI\_matlab
- *But first, an overview of physiologic noise in BOLD MRI*

# Overview

Step 3: IRF-RETROICOR

Step 2: RETROICOR

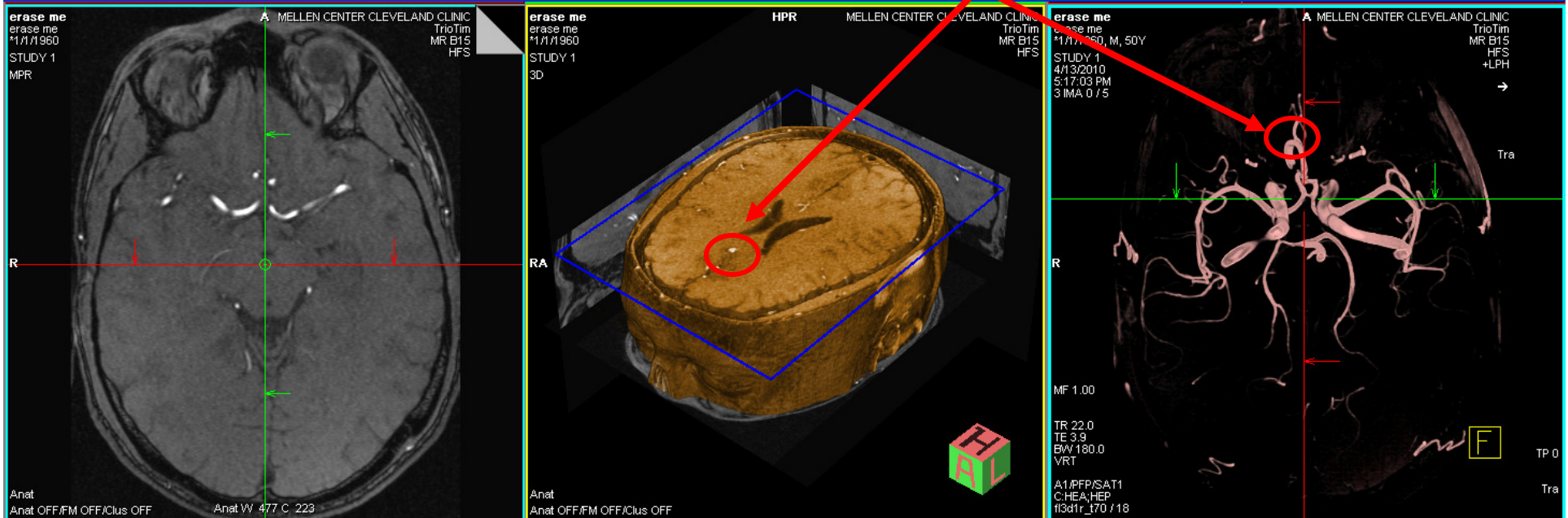
Step 1: PESTICA



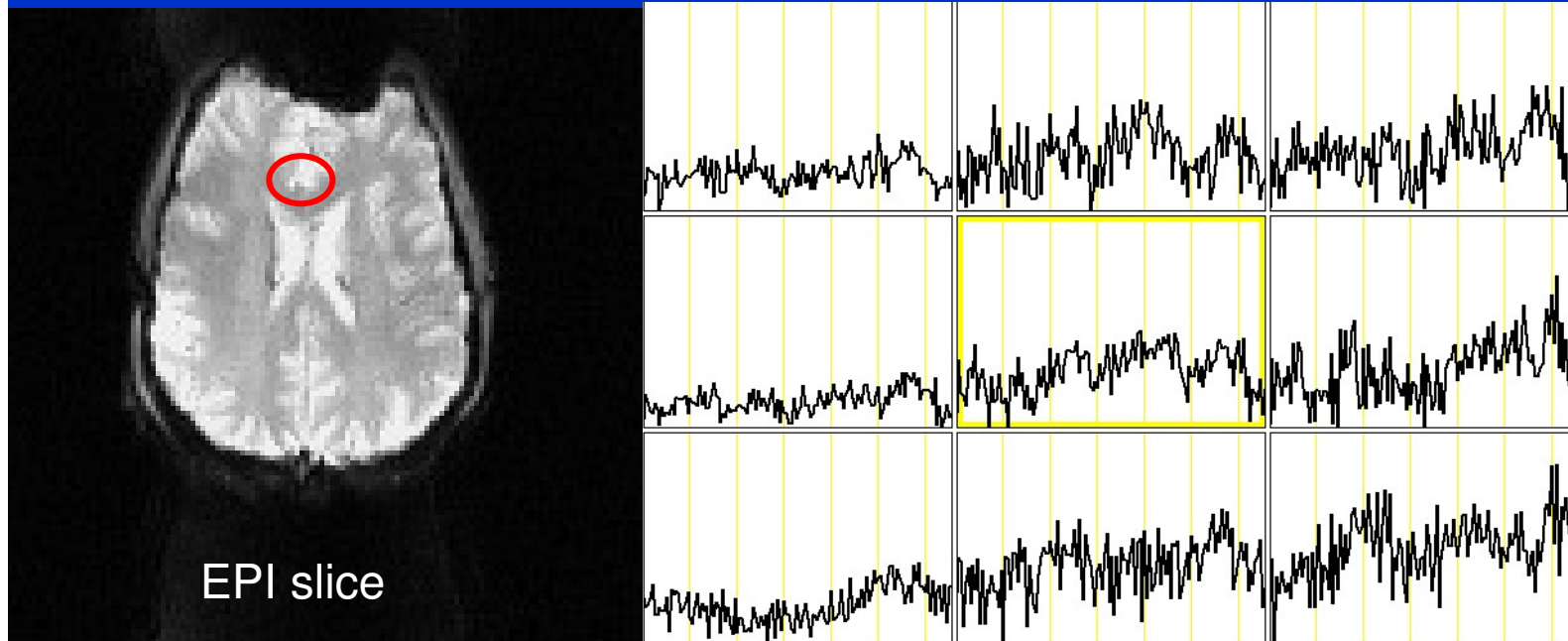
# Physiologic noise in BOLD MR

- Non-neuronal noise from heart/breathing
  - reduces specificity and significance in fMRI
  - spatially varying bias in connectivity data
- Correction 1: temporally filter the data
  - Depends on sampling  $>$  Nyquist rate of noise
  - Rare to have whole-brain sampling high enough for higher harmonics of cardiac noise and respiration, so this doesn't work
- Correction 2: regress out noise
  - Works well, but depends on having a model of the noise
  - We will get this using PESTICA...

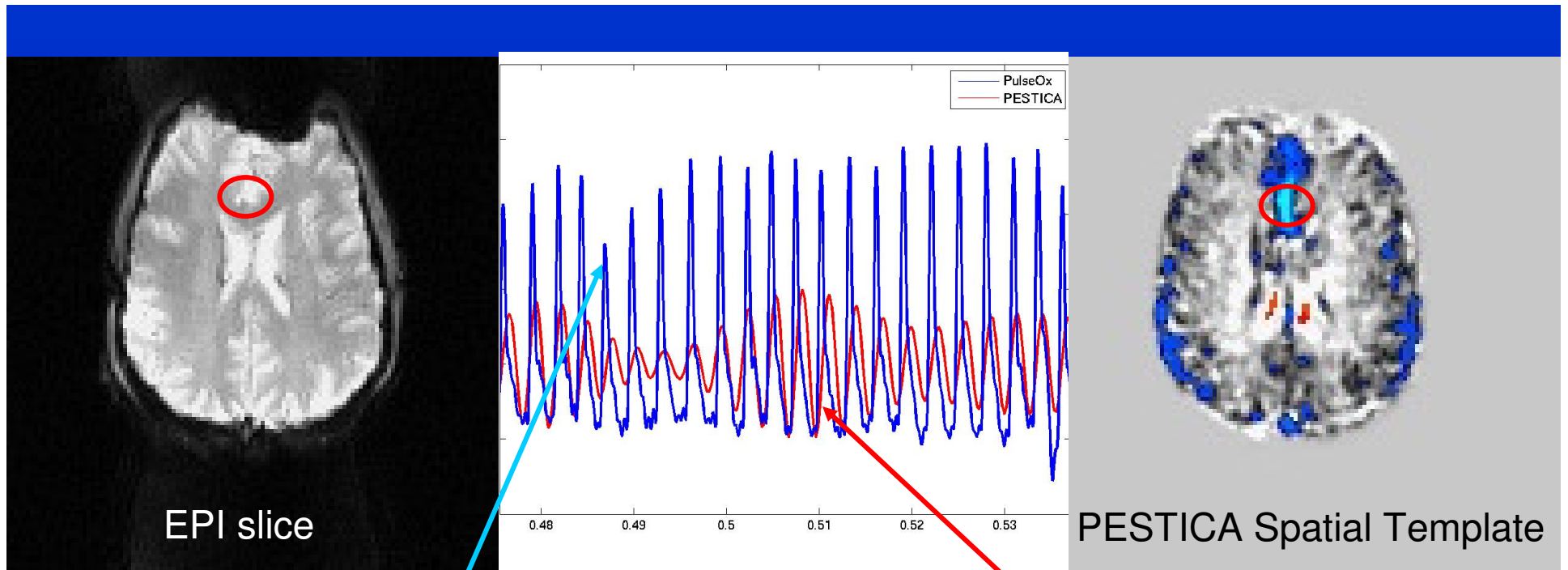
## Cardiac: arterial locations on an MRA, note **anterior cerebral artery (ACA)**



- During every heartbeat systole, blood flow and pressure increases briefly.
- In fast imaging, artifactual signal is phase-locked with the cardiac cycle.



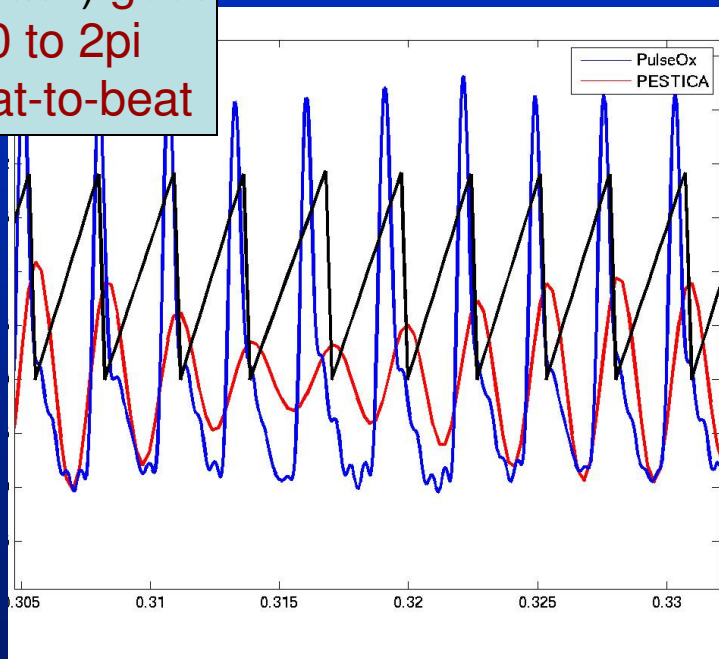
- ACA in echoplanar (EPI) timeseries
- Cardiac noise near ACA doubles or triples variance in those voxels (AFNI plot of 9 voxels centered on ACA voxel)
- Need to measure pulse if we want to account for it...



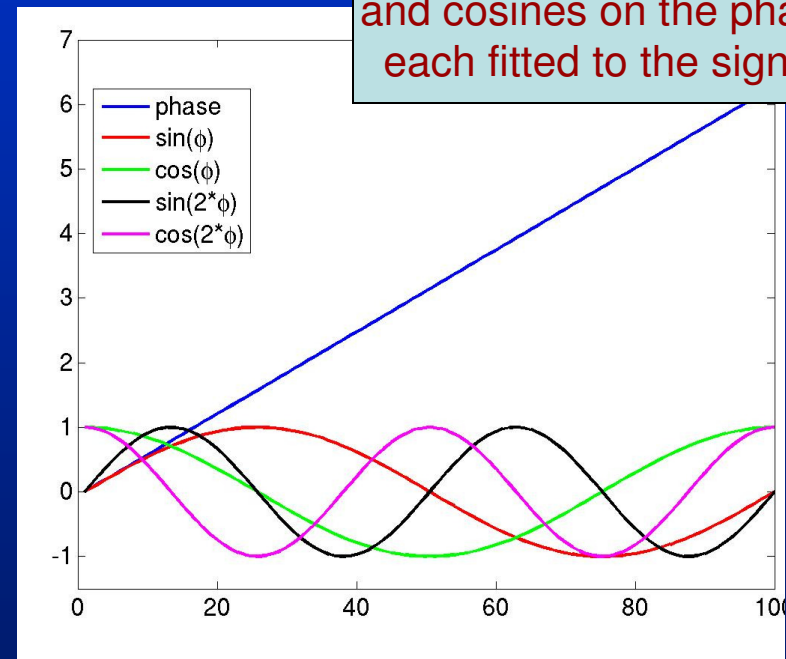
- Spatial template from PESTICA used to track cardiac
- Pulse Oximeter signal displayed with PESTICA-derived pseudo pulse ox signal
  - Same periodicity - See **Lowe08, Wu08 Wu09, Beall10.**
  - PESTICA gives us the periodicity, we must still model the beat-to-beat noise ...

- RETROspective Image-based CORrection
- If noise signature is same beat-to-beat, use model based on the relative phase within a cycle: Fourier series of phase.

Phase (black) goes from 0 to  $2\pi$  from beat-to-beat



Fourier Series Expansion is M orders of sines and cosines on the phase, each fitted to the signal.

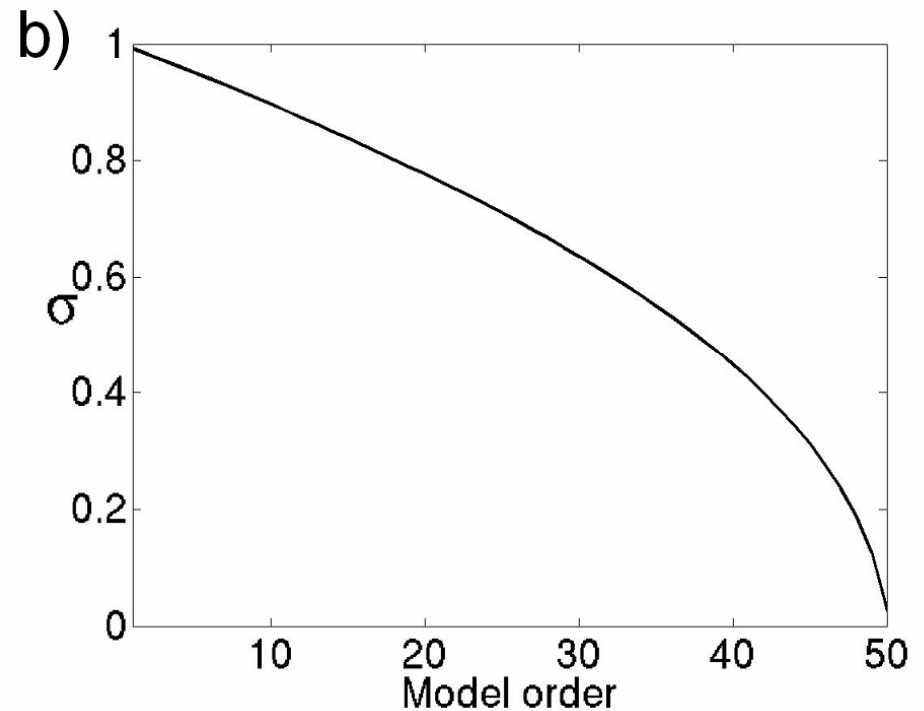
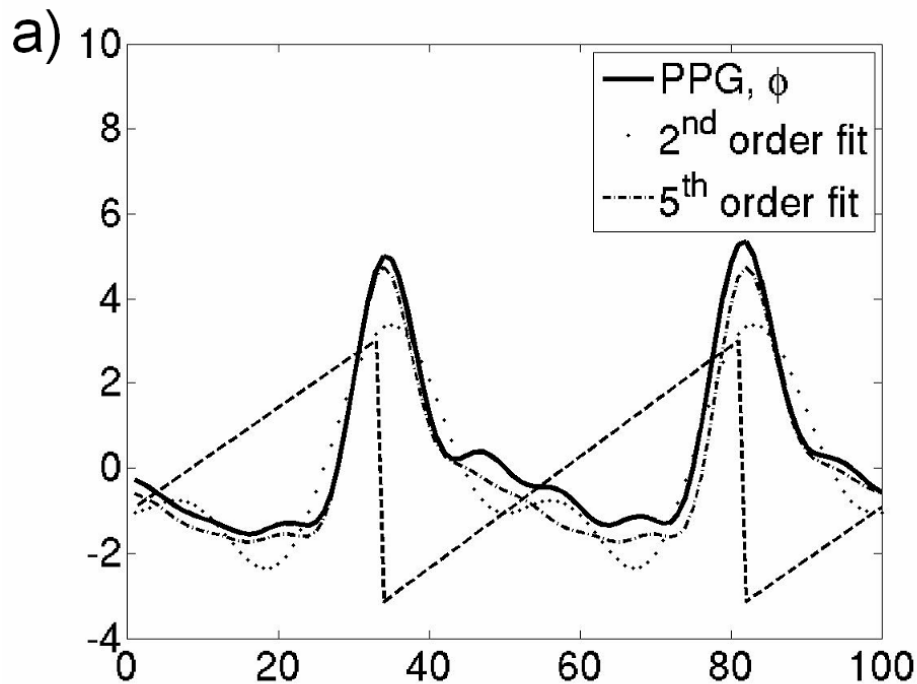


- Do the same for respiratory noise: convert respiratory signal to phase, then 2 to 5 orders of Fourier series on it. This is **RETROICOR**.



# Higher order terms

- RETROICOR model is general and effective.
- However, to get *all* the noise at higher field strengths, need 3<sup>rd</sup>-5<sup>th</sup> order series.
  - RETROICOR **Glover00** at 1.5T recommended 2<sup>nd</sup> order, **Harvey08** recommends 4<sup>th</sup> order cardiac, 3<sup>rd</sup> order respiration but cautions about overfitting data.
  - We observe 5<sup>th</sup> order for both retains significant coupling, but removes a lot of non-noise variance.
- That is a lot of regressors: 5<sup>th</sup> order cardiac+5<sup>th</sup> order respiratory=20 regressors



- Increased model order increases accuracy of fit (left fig)
- As model order increased, standard deviation decreases, **even when we're regressing random noise which has nothing to do with our dataset!**
  - Eventually there is nothing left... (right fig is random phase and data with 100 timepoints, 50<sup>th</sup> order=100 regressors)
- This means we're ***unavoidably damaging our data at a level dependent on the number of regressions***

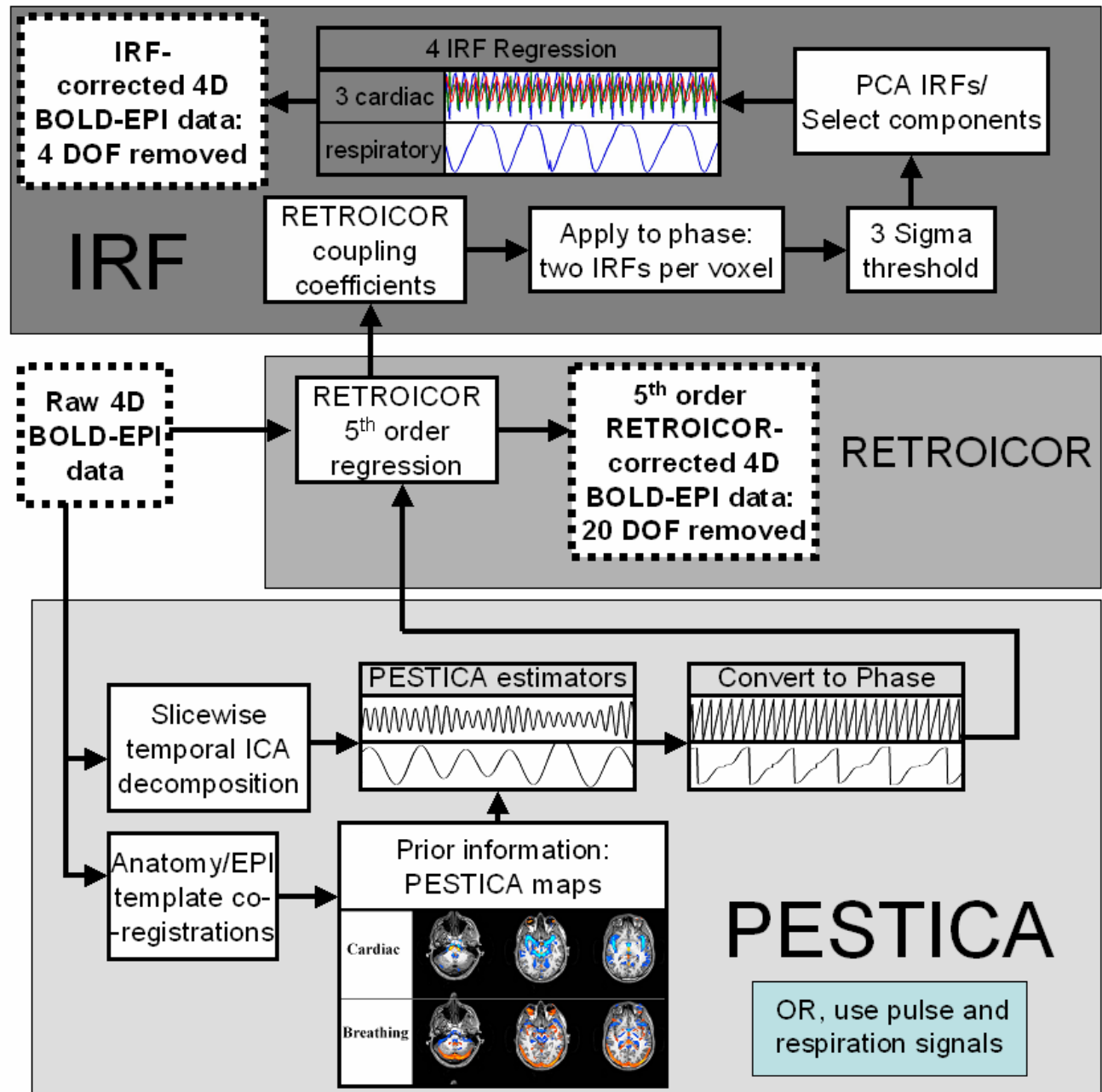
# Impulse Response Functions (IRF)

- Each voxel produces a different shape noise response in the fitting process, we want to select only those fits that explain signal over and above background noise
  - Take fitted RETROICOR coefficients, but pass to Principal Component Analysis
  - Keep only those that explain most of variance
    - use Monte Carlo to set 3sigma threshold
  - End up with conservative set of noise response shapes: IRFs

3: IRF-RETROICOR

2: RETROICOR

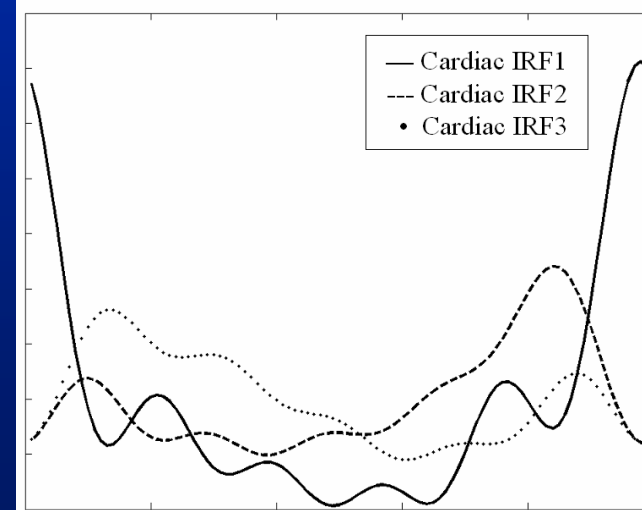
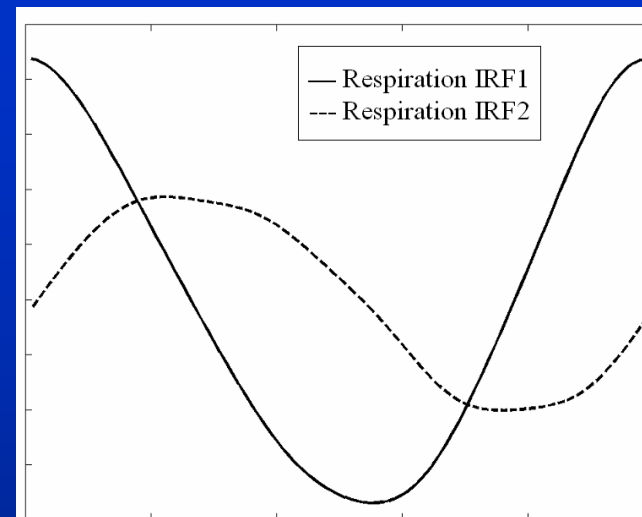
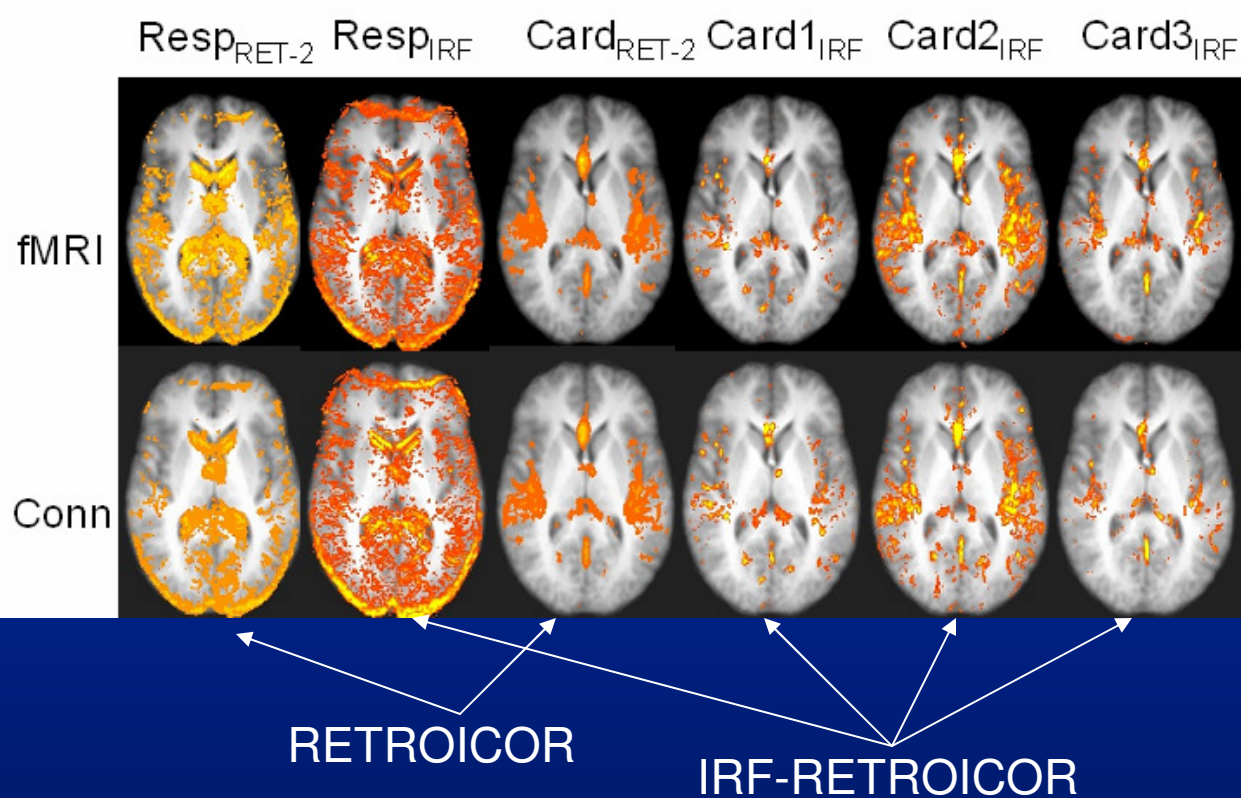
1: PESTICA



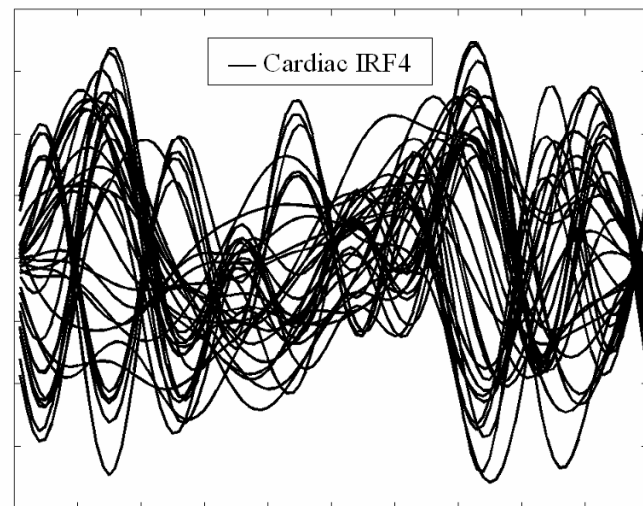
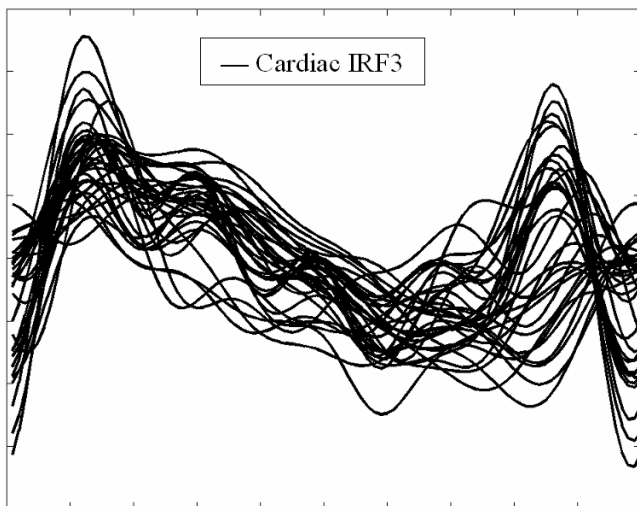
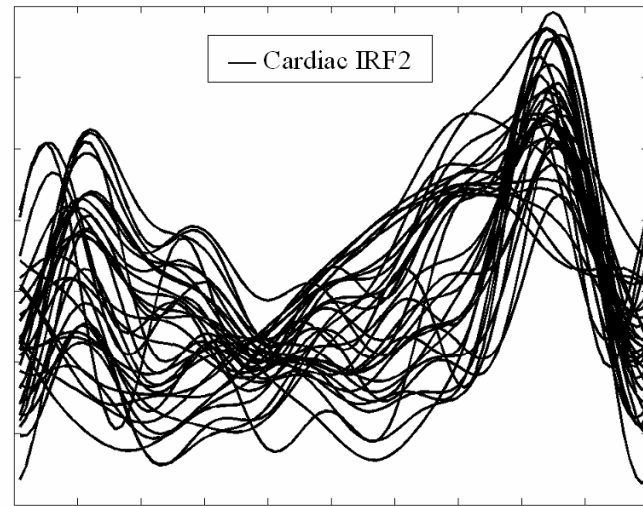
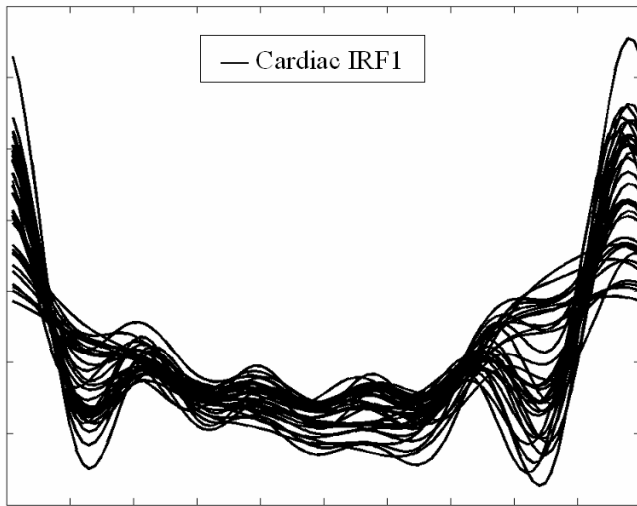
# IRFRET shapes

- If you feed in physiologic signals from the wrong subject, or just plain noise, you get arbitrary/inconsistent IRF shapes and totally wrong coupling maps
  - Therefore, the consistency of IRF shapes/maps is useful for validation
- Number of voxels significantly coupled also useful as validation
  - However, both of these are dependent on acquisition and may change with protocol changes

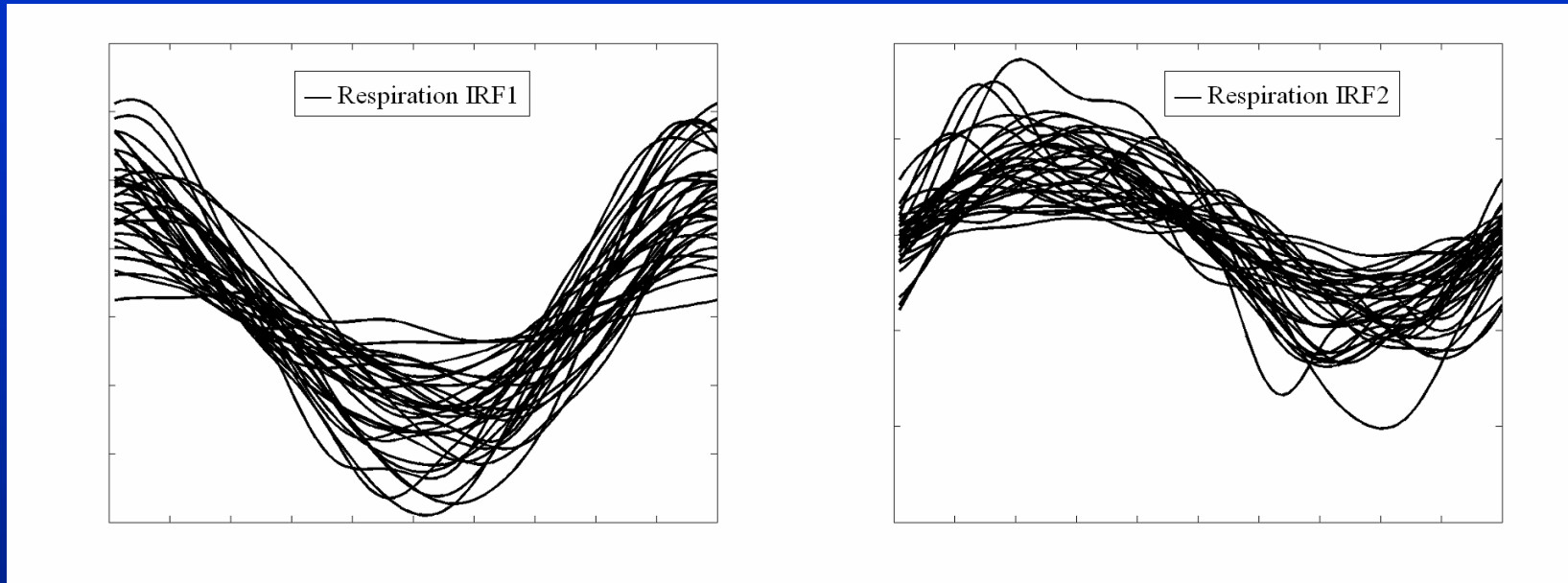
# First, IRFs and coupling maps



# Multiple subject cardiac IRFs:

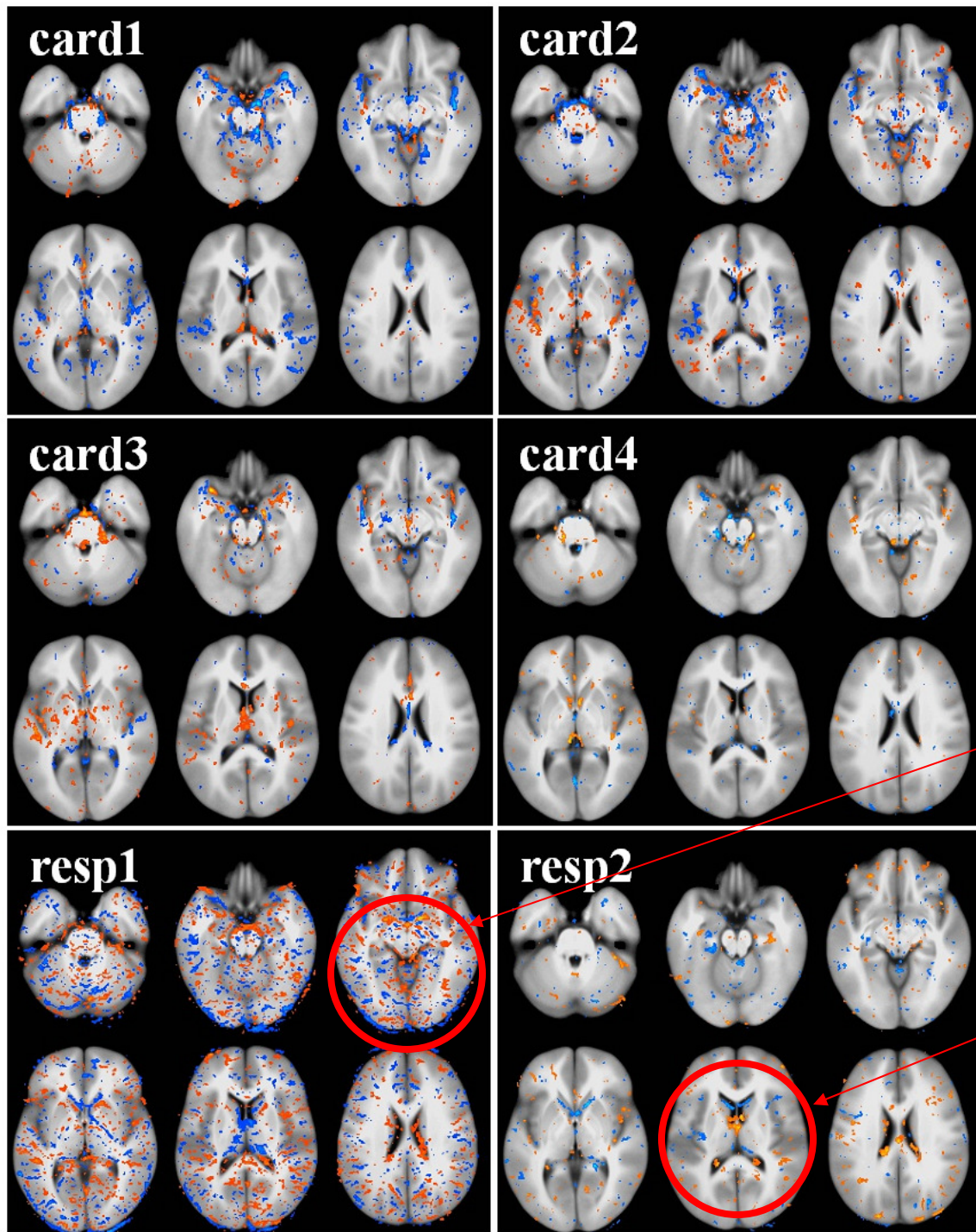


# Respiratory IRFS



- Striking similarity across 34 independent subjects (same results in both block-paradigm motor fMRI and in resting connectivity scans) for first 3 cardiac and first 2 respiratory IRFS, all data analyzed separately.
- When random phase is input to algorithm, we get garbage out. Very little coupling and arbitrary IRFs.





# Physical consequences

- No clear physical picture for cardiac, but respiration may make sense
- First IRF is chest expansion-induced field offset, second IRF may be CSF flow secondary to respiratory motion of brainstem

# Now, how to run PESTICA...

- Prep: setup PESTICA for your data (once)
  - » Always source the setup file in later use
- Run All stages in order (can specify stage)
  - *run\_pestica.sh <datafile>*
- Step 1: run temporal ICA on every slice
- Step 2: run PESTICA estimation algorithm
- Step 3: filter estimators (interactive)
- Step 4,5: run IRF-RETROICOR, optimize
- Step 6: run QA script

# Preparation of PESTICA

- Un-tar PESTICA distribution
  - `tar xvfz pestica.tar.gz`
  - `cd pestica`
- Edit *bash* script inside PESTICA distribution
  - Edit `setup_pestica.sh`, choose 1) data directory, 2) voxel volume and 3) orientation (contact me if you use *tcs*h or another shell)
- Run setup ONLY once from pestica directory
  - `./setup_pestica.sh`
    - Sets up links to the appropriate orientation geometry, check with viewer against your data, creates a file “.firsttime” so it doesn’t relink files during future runs of this setup script.
      - » If you later need to setup with different orientation, remove .firsttime and re-run again from inside pestica directory
- Put setup into your startup script
  - Edit `.bashrc` or similar startup script in your home dir, add:
  - `source /home/user/pestica_directory/setup_pestica.sh`
    - Or `source` setup file manually every time you need PESTICA

```
fmri@pub2:/mnt/netScratch/fmri/pestica2mm - Shell - Konsole
Session Edit View Bookmarks Settings Help

[fmri@pub2 ~]$ cd /mnt/netScratch/fmri/pestica2mm
[fmri@pub2 pestica2mm]$ ls
apply_PESTICA1.m          prepare_ICA_decomp.m
apply_PESTICA2.m         prepare_PESTICA1.m
assemble_slices_to_timeseries.m  README
disassemble_timeseries_to_slices.m register_ep2d_stddev_to_MNI.m
eeglab                   register_epi2epi_PESTICA1.m
example_runfile.m       setup_pestica.sh
get_correlation_threshold.m tfilter_fft.m
optimize_pmu.m          view_and_correct_estim_for.m
pestica_volumes
[fmri@pub2 pestica2mm]$ emacs setup_pestica.sh

[1]+  Stopped                  emacs setup_pestica.sh
[fmri@pub2 pestica2mm]$ bg
[1]+  emacs setup_pestica.sh &
[fmri@pub2 pestica2mm]$ ./mnt/autofs/netHome/scratch/fmri/pestica2mm/setup_pestica.sh
[fmri@pub2 pestica2mm]$ env | grep PESTICA
PESTICA_ORIENT=RAI
PESTICA_DIR=/mnt/autofs/netHome/scratch/fmri/pestica2mm
PESTICA_VOL=16mm3
[fmri@pub2 pestica2mm]$ ls
apply_PESTICA1.m          MNI_T1.hdr          resp_mean_mni.hdr
apply_PESTICA2.m         MNI_T1.img          resp_mean_mni.img
assemble_slices_to_timeseries.m optimize_pmu.m       setup_pestica.sh
card_mean_mni.hdr        pestica_volumes     stddev_avg_mni.hdr
card_mean_mni.img        prepare_ICA_decomp.m stddev_avg_mni.img
disassemble_timeseries_to_slices.m prepare_PESTICA1.m  tfilter_fft.m
eeglab                   README              view_and_correct_estim
example_runfile.m       register_ep2d_stddev_to_MNI.m
get_correlation_threshold.m register_epi2epi_PESTICA1.m
[fmri@pub2 pestica2mm]$
```

1. Choose directory for pestica
2. Edit setup\_pestica.sh
3. Source the setup file for first time (first time MUST be in directory)
4. Check environment and files (links are made only during the first time setup\_pestica.sh is run, subsequent runs only setup environment)

```
emacs@localhost.localdomain
File Edit Options Buffers Tools Insert Help

# edit the hard-coded base directory containing the matlab code and the averaged vol
export PESTICA_DIR="/mnt/autofs/netHome/scratch/fmri/pestica2mm"

# edit your matrix size (approximate, decide which is closer to your voxel volume, 16mm^3 or 64mm^3)
# one set was created using 2x2x4mm voxels (16mm^3), the other with 4x4x4 voxels (64mm^3)
export PESTICA_VOL="16mm3"
#export PESTICA_VOL="64mm3"

# edit the hard-coded orientation depending on your scanner's reconstruction orientation
export PESTICA_ORIENT="RAI"

--- setup_pestica.sh (Shell-script[bash])--L1--Top-----
Minibuffer window is not active
```

- 1) Edit directory, pick best
- 2) matrix size (16 or 64mm<sup>3</sup>),
- and 3) choose orientation

# Data setup

- “cd” to your data directory
- Data must be in one of: ANALYZE, NIFTI, NIFTI\_PAIR, NIFTI\_GZ or AFNI BRIK
  - Make sure EPI data is in 3D+time format
  - Note, PESTICA uses the AFNI\_matlab distribution for reading/writing
  - First run will convert a non-AFNI BRIK format input file to AFNI BRIK format

# run\_pestica.sh -h

Usage: run\_pestica.sh -b ep2d\_pace\_132vols (for AFNI BRIK format)

OR : run\_pestica.sh -b ep2d\_pace\_132vols.hdr

OR : run\_pestica.sh -b ep2d\_pace\_132vols.nii

OR : run\_pestica.sh -b ep2d\_pace\_132vols.nii.gz

Opt : run\_pestica.sh -b ep2d\_pace\_132vols -r -c ep2d\_pace\_132vols.moco -v 4 -s "3 4 5 6"

## Flags:

**-b = base 3D+time EPI dataset you will run ICA and physio estimation upon (REQUIRED)**

**-r = flag to run original RETROICOR algorithm as well as IRF-RETROICOR**

**-d = number of volumes to drop at start (default is 0, if your data is not equilibrated, use up to 4)**

**NOTE: you MUST not have un-equilibrated volumes at the start. Most scanners take "dummy" volumes, where the ADCs are turned off but the RF and gradients are running as normal, for the first ~3 seconds (modulo TR), but in some scanners this is not so and you can see contrast change from the 1st to 2nd volumes. Look at your data!**

**-c = 3D+time EPI dataset you will run any corrections on (e.g. run ICA on uncorrected data, but apply physiologic noise correction to motion-corrected data) - assumes timing is identical.**

**This is safe to use with the -d flag; PESTICA will detect number of vols. This file must be in AFNI BRIK format.**

**Example: run\_pestica.sh -b ep2d\_pace\_132vols -r -c ep2d\_pace\_132vols.moco -d 4**

**will create estimators that are 4 volumes shorter than ep2d\_pace\_132vols, so ep2d\_pace\_132vols.moco must already be 4 vols shorter. In this example, it could have been created like so:**

**3dvolreg -prefix ep2d.moco -base "ep2d[0]" -zpad 8 -heptic "ep2d[4..\$]"**

**NOTE: this is not recommended - PESTICA estimators seem better if the ICA is also done on the moco'ed data.**

**-s = "1 2 3 4 5 6" individual stages to run**

**Stage allows you to run or re-run only parts of PESTICA**

**please give the stages (in increasing order) that you want to run**

**e.g. : run\_pestica.sh -b ep2d\_pace\_132vols -s "5 6"**

**otherwise, this script will run all stages in order**

**If not in BRIK format, this script will make a copy into AFNI .BRIK format for PESTICA**

# Use PESTICA on your data...

- Run “*run\_pestica.sh <datafile>*”
  - Scripts will create a new subdir “<data>\_pestica/” and all working data will be stored here for later use
- First use of script on data will create a masked brain named <dataset>.brain+orig
  - If you like, you can use your own masked brain by copying or renaming your mask to this filename
- Can run individual stages
  - *run\_pestica.sh <datafile> -s “3 4 5 6”*
  - ICA stage (1) and second-pass correction stage (5) will take considerable time (newer MATLAB is faster)
  - *If you don't specify stage, it (should) run all in order.*

# Optional arguments

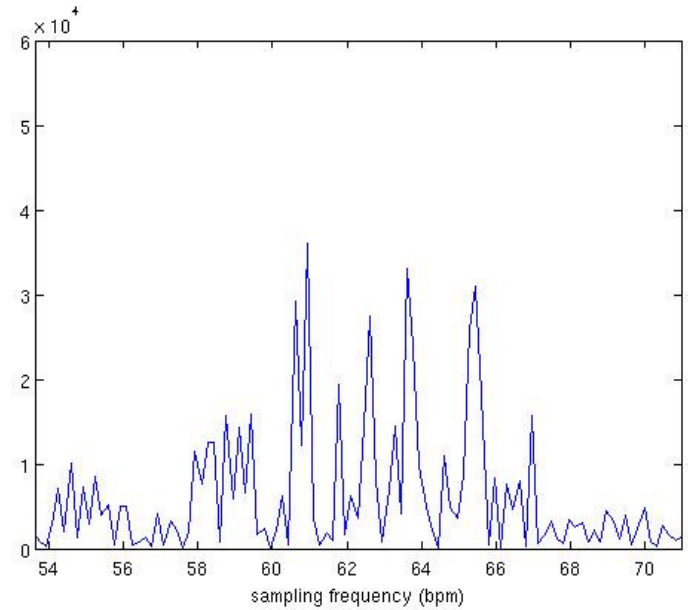
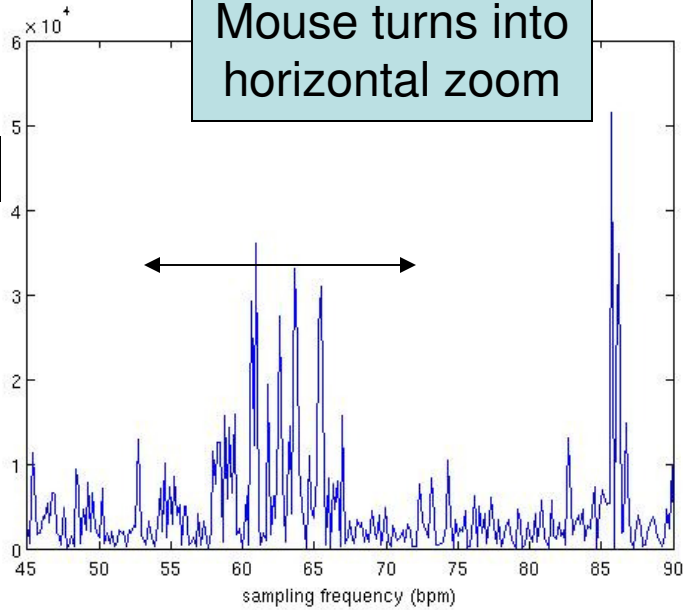
- “-r” flag: run original RETROICOR
- “-d” opt: drop X volumes at start of datafile
  - All subsequent work will those volumes
- “-c” opt: option to apply corrections (stage 4-5) on different datafile than used in ICA
- “-s” opt: specify which stages to run
  - Useful if a stage failed and must be re-run
  - OR, if you want to try different filtering, then start at stage 3: `run_pestica.sh <data> -s “3 4 5 6”` and see if coupling is improved in QA plots



# Apply temporal filter to estimators

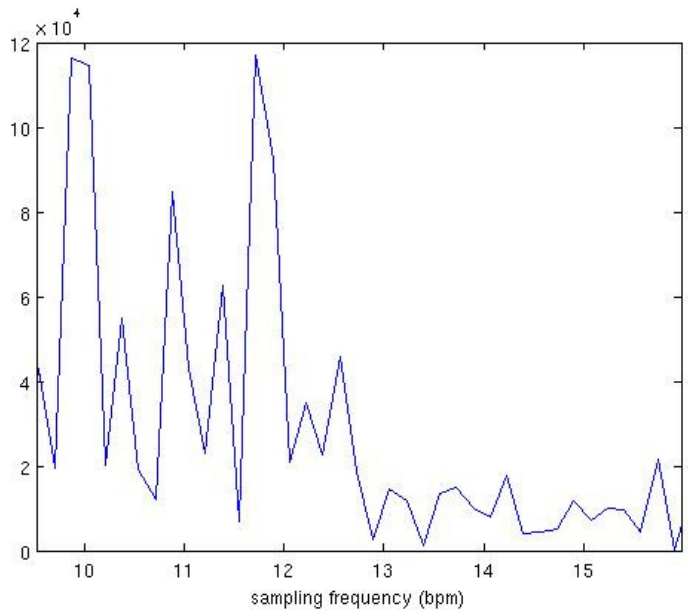
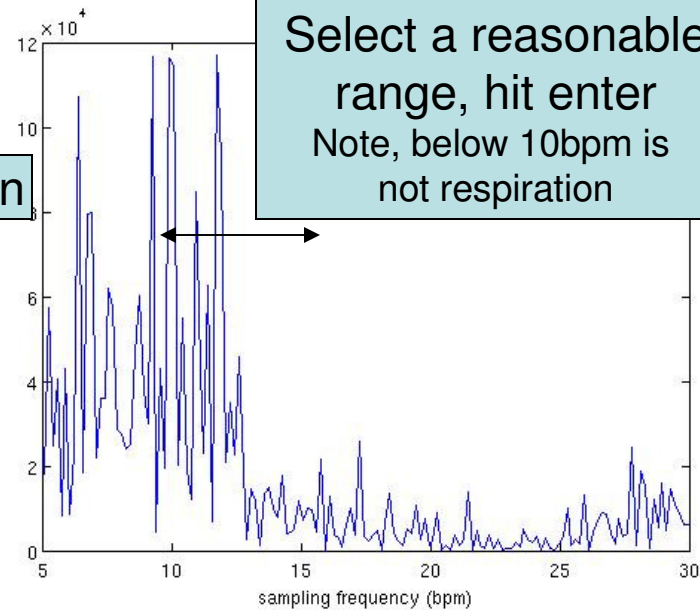
- Stage 3 of `run_pestica.sh` will pop up interactive window for cardiac and respiratory estimators – the filter window must be selected interactively
- Creates MATLAB plot showing Fourier transform of cardiac estimator, zoomed to appropriate region
  - Initial range is set to 45-90 beats per minute, can select further, look for bunch of peaks in this range. In a large set of subjects, we found the cardiac peak freq to range from 48 to 85bpm with 5-10bpm width per subject.
  - Zoom with mouse over the peaks you suspect are cardiac.
  - If you don't have much confidence, feel free to leave zoom region large (already zoomed to 45-90bpm, which is a reasonable filter)
  - Hit enter in MATLAB command line, move to next (respiration)
  - Respiration freq range initially set to 5-30, breaths per minute, but typically between 10 and 25.

Cardiac



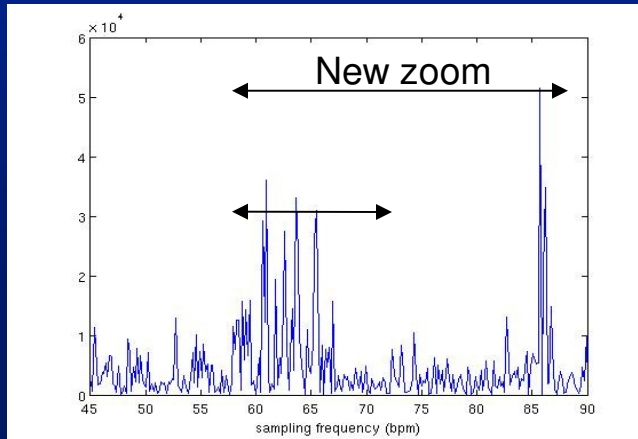
If you are ever in doubt, go wider!

Respiration



# My notes on frequency selection

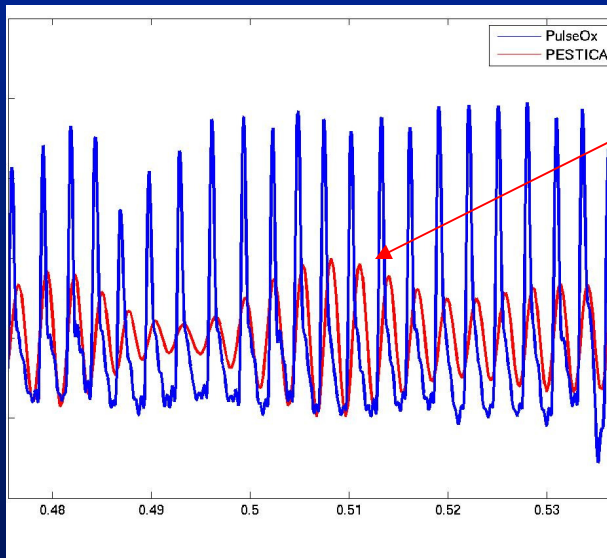
- In pulse monitoring data, we never saw anyone outside the 48-85 and 10-24 breaths or beats per min range – this plot is intentionally wider.
- Look for “chunk” of peaks that’s about 5-10bpm wide – its rarely but occasionally narrower than this, sometimes wider though...



Compared to previous page, I would re-run filtering and later stages with “run\_pestica.sh -s ‘3 4 5 6’ ” and include the 85 bpm peak to see if I get a better result. In this subject, it did not improve things, leading me to conclude that the 87bpm peak is not cardiac. You can always re-run stages 3-5 again (caveat, stage 5, IRF-RET, takes 1-4 hours).

# IRFRET

- Stage 5 does a second-pass of IRFRET after cleaning up the cardiac estimator. This cleaning process takes the most time – it is adjusting the dither of each cardiac peak to optimize it.
  - Temporal filtering adds dither in peak, which is substantial for cardiac. This fixes it, and results in output that is closer matched to a pulse oximeter



Note dither in peak locations – sometimes the peaks line up, sometimes they are shifted slightly, which will hurt fitting.

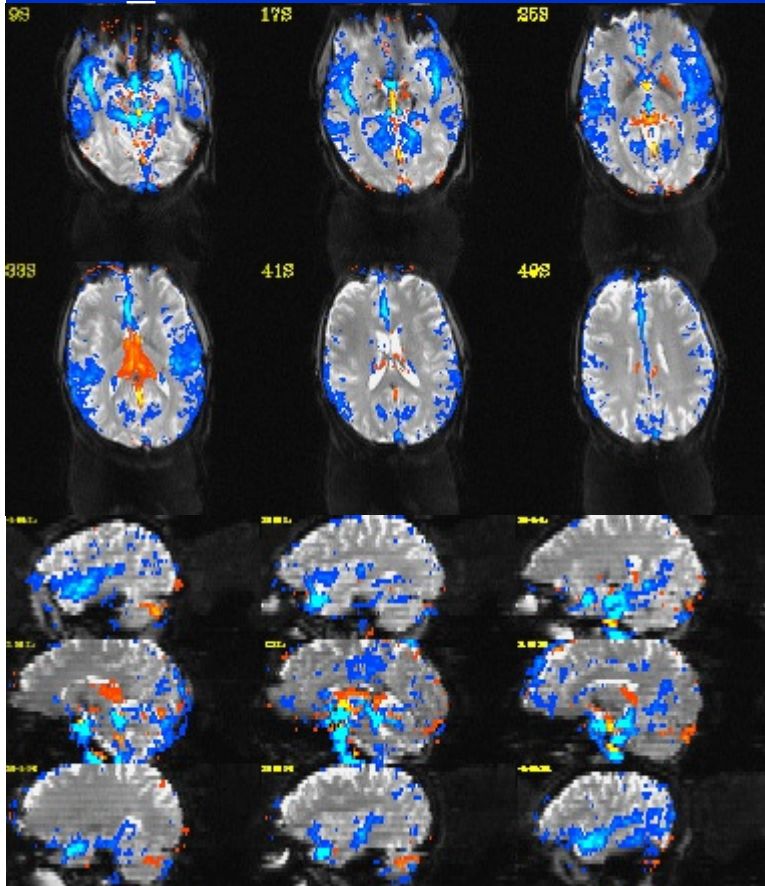
This dither is due to harmonics in cardiac process that filtering removed and thermal noise added in to signal. Cleanup script fixes this – validated against pulse ox data.

# QA

- Stage 6: `run_pestica.sh <data> -s 6`
- MATLAB figures showing histogram of breaths/heartbeats per minute, IRFs and coupling maps overlain on data
  - IRFs are data acquisition-specific
- Jpegs saved in `<data>_pestica` subdir include number of voxels significantly coupled to heart and breathing processes
  - Also acq-specific, but >2000 voxels seems good for our acquisition
  - We will try to build a database of voxel count cutoffs for QA purposes for several different acquisitions
- Jpegs show template coregistrations (templates are always same, just look at them for coreg verification and to compare with coupling maps for spatial structure) and coupling maps
  - In coupling maps, look for arteries/ventricles in cardiac, anterior/posterior edges in respiration... Compare with template overlays
  - Images are not very cleanly indicative of physiologic noisy regions, but this is typical, and these are the same as what you'd get with pulse ox/respiration belt. Mostly just make sure you're not getting flat or weird maps.

# Check template coregistration

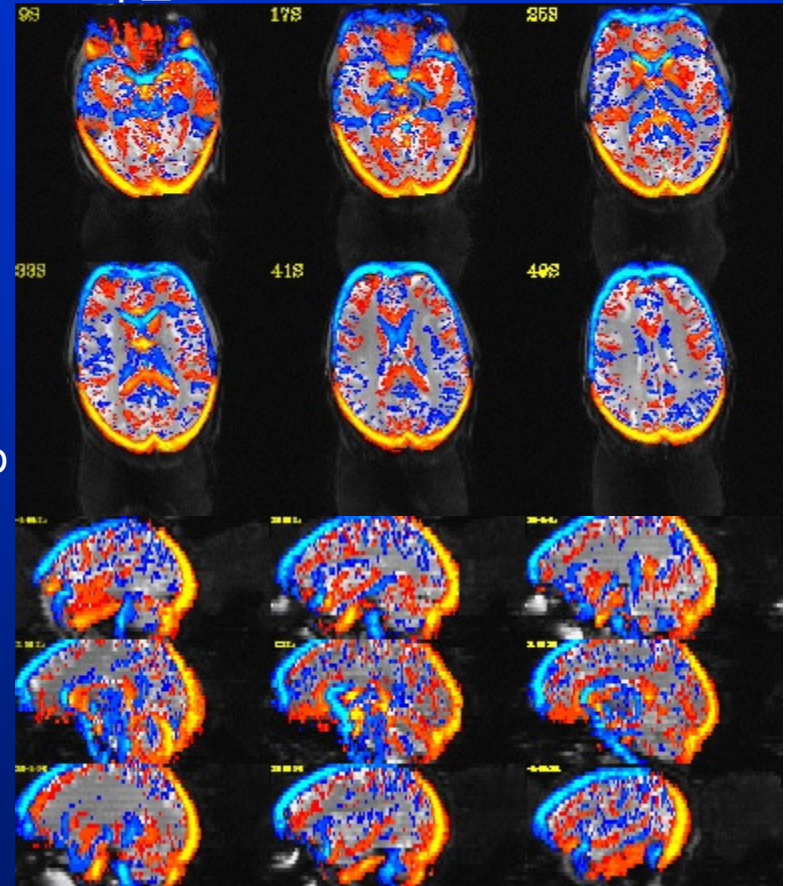
card\_PESTICA.nii overlain on EPI data



Note cardiac coupling lines up over arteries, ventricles,

respiration lines up over anterior/posterior edges of brain structures

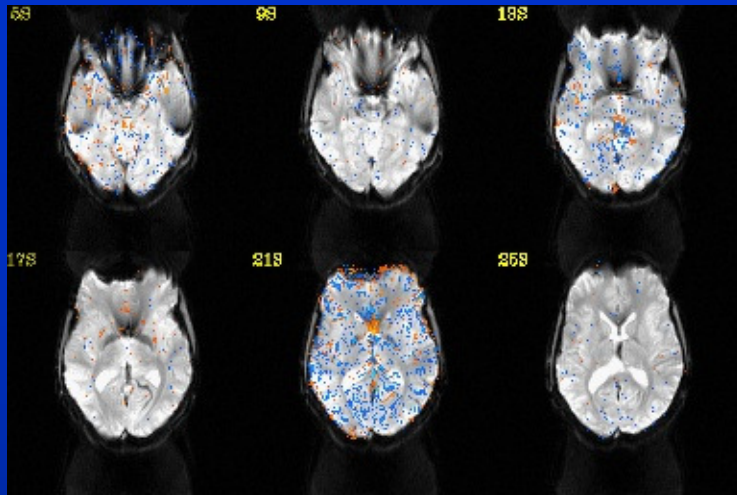
resp\_PESTICA.nii overlain



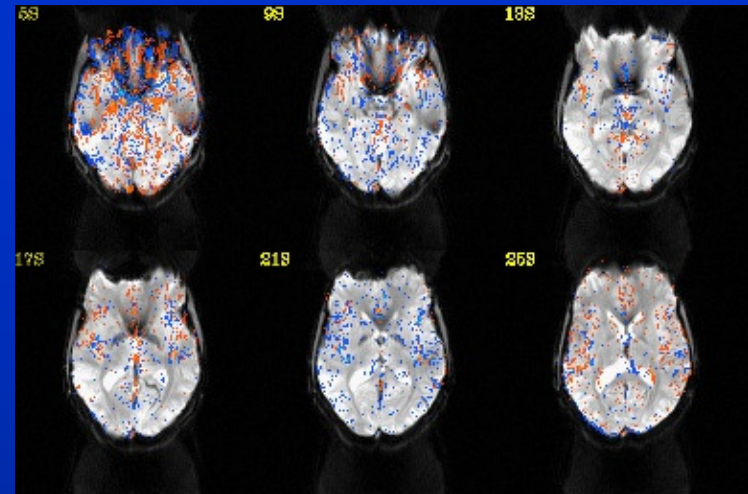
Use “afni -dset pestica/\* .nii” and it will pick up files in current directory as well, set overlay threshold to 5, and turn off autoRange and set range to 30. For a nicer colorscale, set color panes to 12 (instead of \*\*) and you should get similar visual appearance as shown here. If coreg doesn't work, it will be obvious.

# Cardiac coupling maps

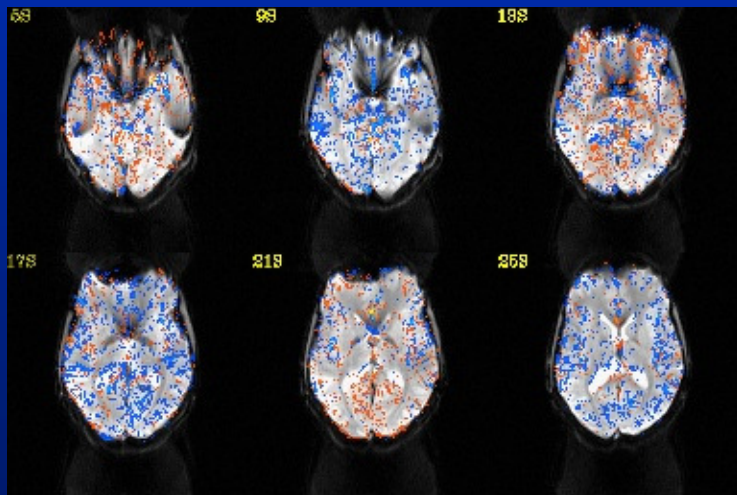
IRF1



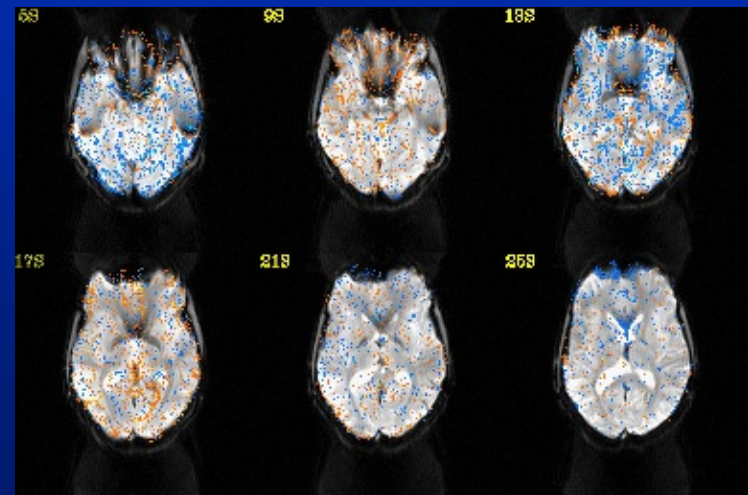
IRF3



IRF2

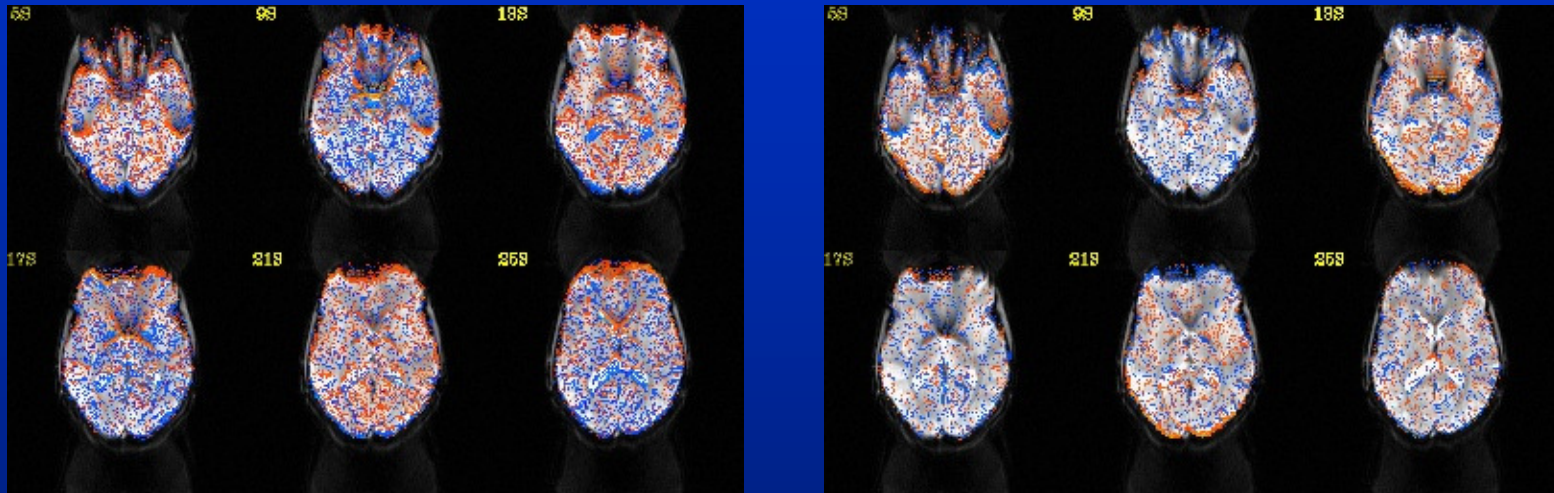


IRF4



Note artery in front of corpus callosum in some IRFs, ventricles, CSF spaces and some MCA arteries. Also note strong every other slice – due to slice acquisition being interleaved in this case. Your data may or may not have this.

# Respiratory coupling



Note anterior/posterior edges of structures. This is because phase-encoding direction is anterior/posterior. This is stronger on Siemens Trios and Verios than Allegras and several non-Siemens scanners. The respiratory artifact is primarily a small field shift that is mostly seen as image shifting in the phase-encode direction by a portion of a voxel. Hence the coupling shows up along image intensity boundaries perpendicular to the A/P axis.