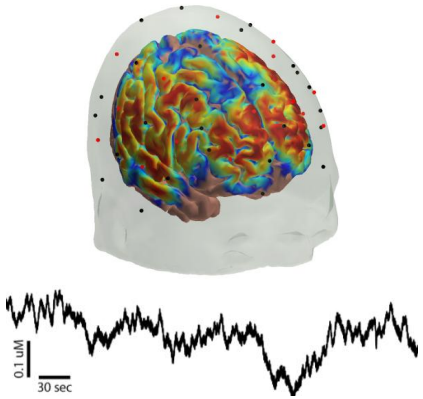
Functional Connectivity with fNIRS

Rickson Mesquita, Luis Felipe Bortoletto, Androu Abdalmalak

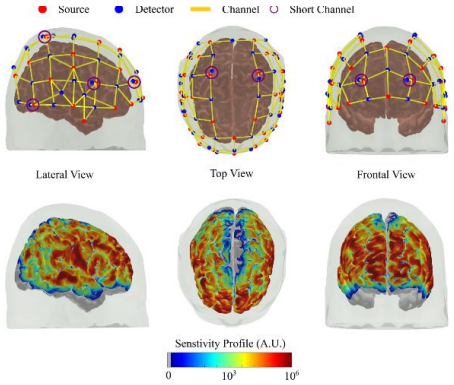
University of Birmingham, UK
University of Campinas, Brazil
Western University, Canada

Overview

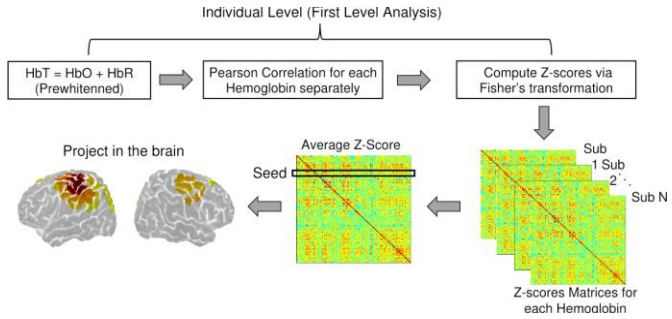
What is functional connectivity (focus on resting-state)?



Resting-state design considerations



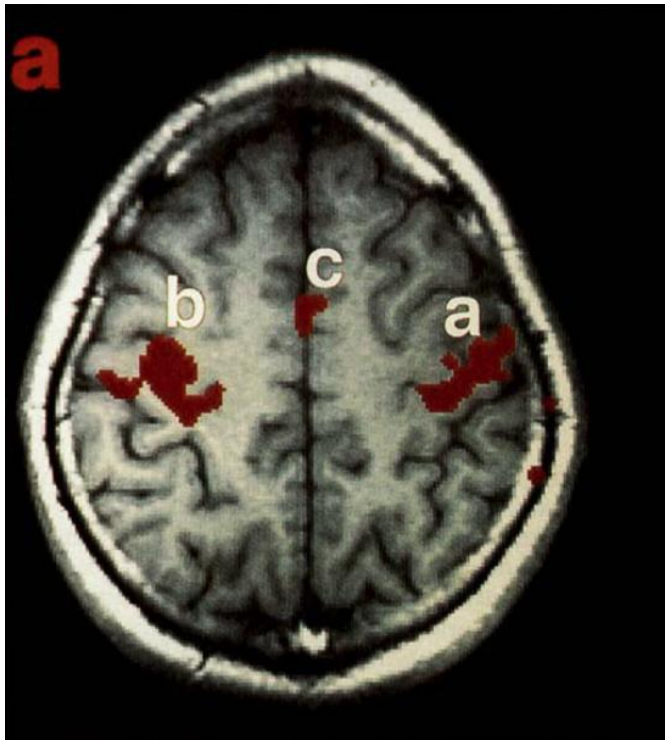
Sample preprocessing steps



What is functional connectivity?

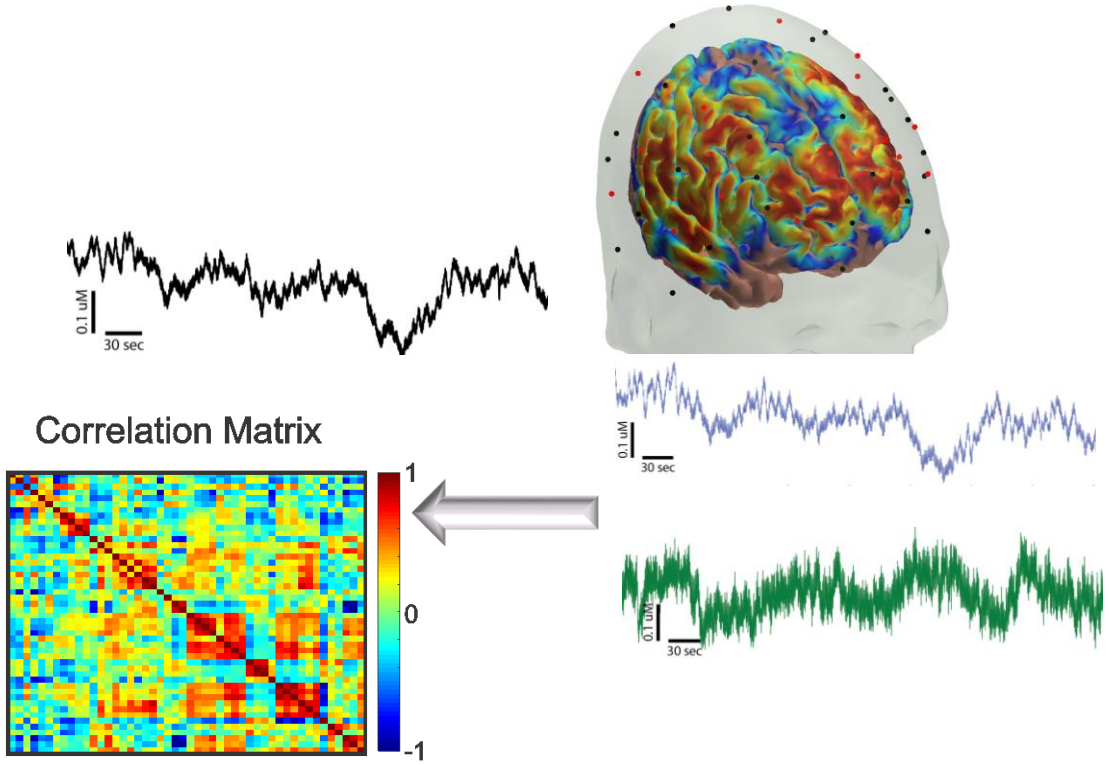
TASK-BASED ACTIVATION

Where does mean activity change with a task or condition?



FUNCTIONAL CONNECTIVITY

Which regional signals fluctuate together over time?



Localization vs connectivity

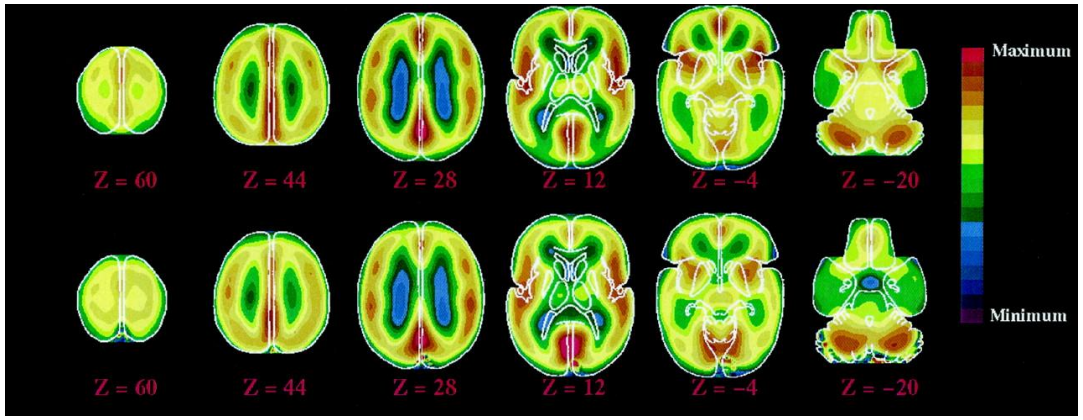
Friston (1994); Biswal et al. (1995)

What is functional connectivity?

- Functional connectivity refers to the statistical relationship between specific physiological signals in time
- Useful if we would like to study the brain without applying an ‘active’ stimulus
- Useful if the stimulus interval is > 50 seconds
- Resting state or task-based (movie watching, hyperscanning)
- We will focus on resting state functional connectivity

Why study the brain at rest?

The brain consumes ~ 20% of the total energy

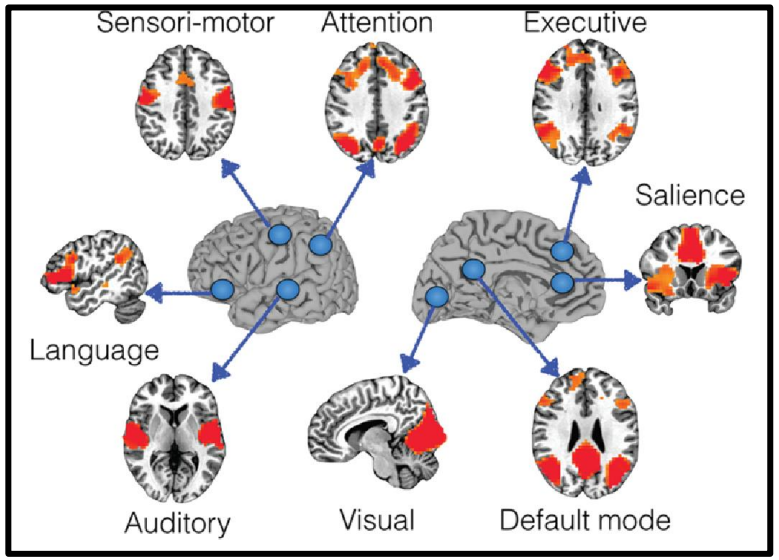


Estimated that ~60-80% of this energy is used to support cell communication

- Task evoked activity accounts for ~1%

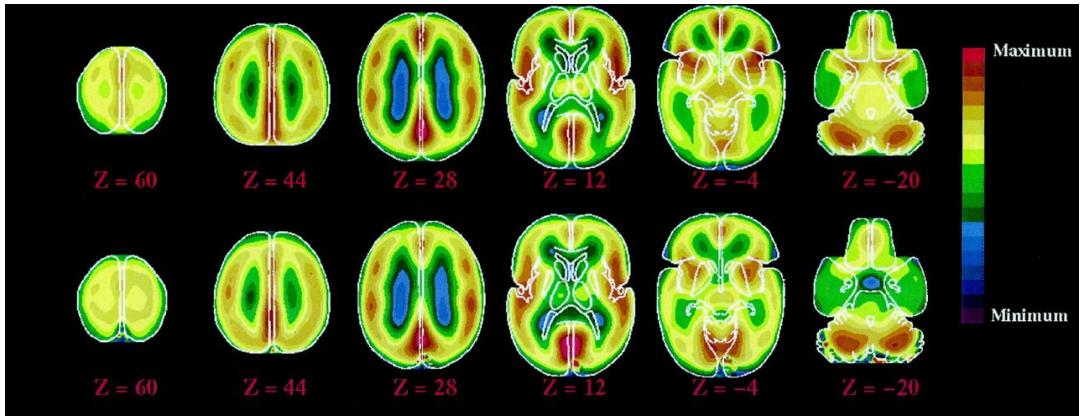
Resting-state measures spontaneous hemodynamic fluctuations in the absence of a task

- Two brain regions are functionally connected if their time series show similarities over time



Raichel et al., 2001, FSL training slides, Biswal et al. (1995); Sheline et al. (2010), Gunsnard et al., 2001

Why study the brain at rest?

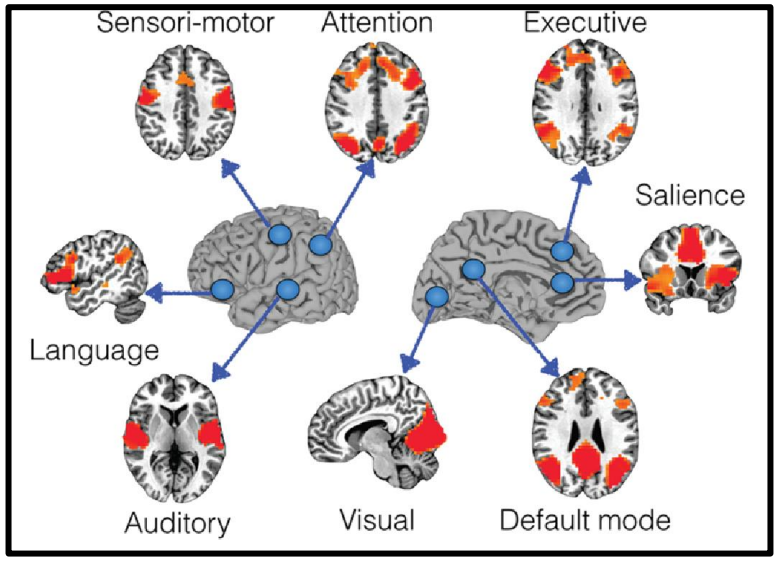


rsFC can give us important information

- Information about the intrinsic functional organization
- Clinical biomarker
- Useful when task-based paradigms are not as feasible
- Simple acquisition across diverse populations
- Easy to collaborate across centers

Disadvantages:

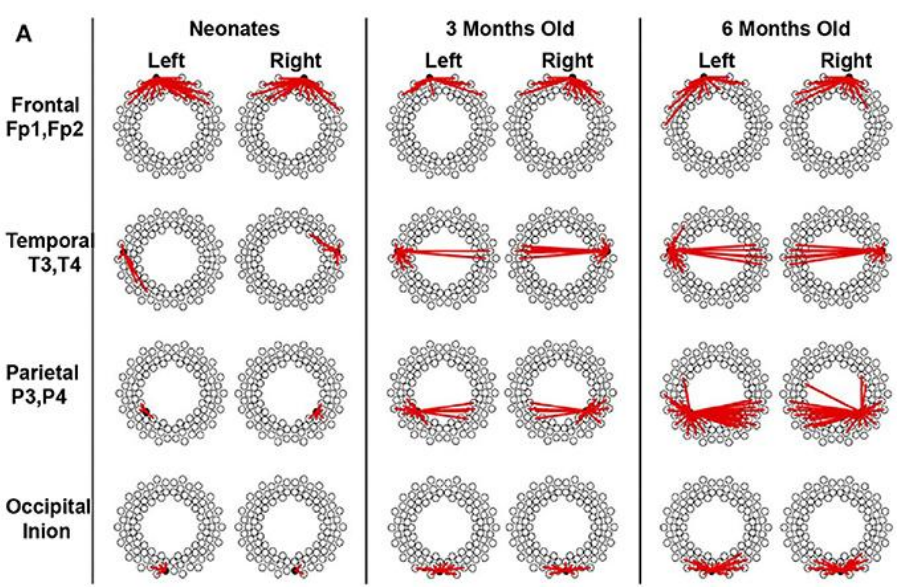
- Difficult for some groups to stay still for the duration of the study
- Requires large number of channels for extracting networks
- Preprocessing is crucial to prevent spurious correlation

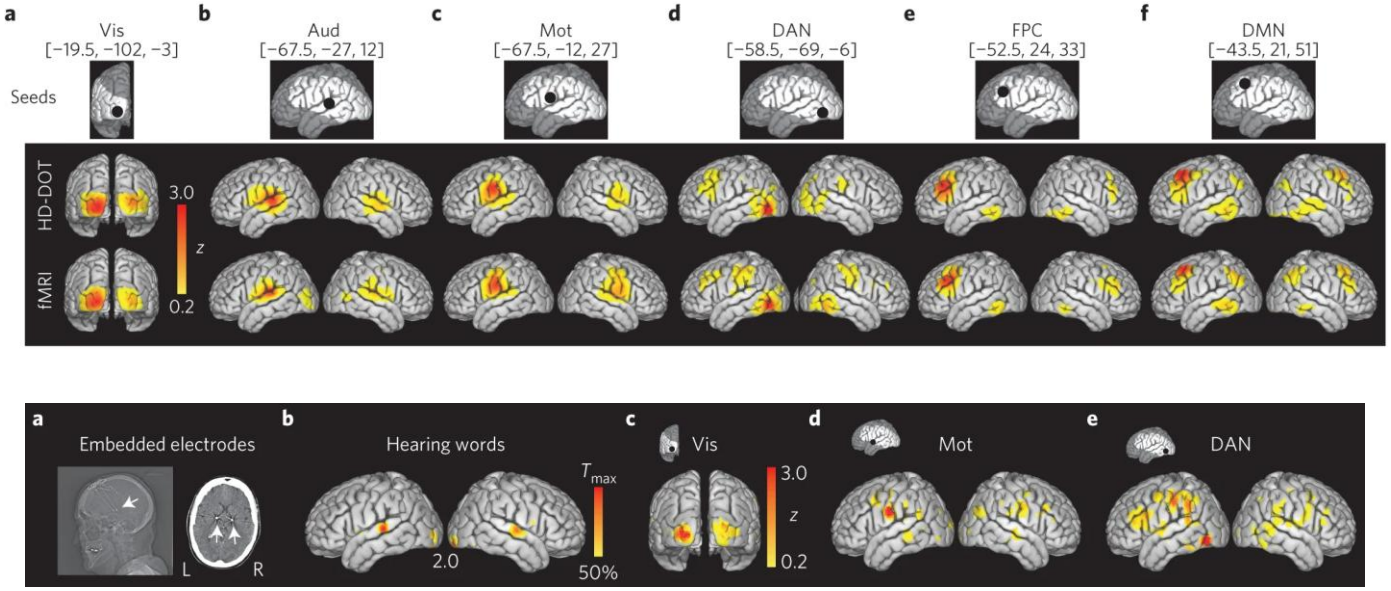


Raichel et al., 2001, FSL training slides, Biswal et al. (1995); Sheline et al. (2010), Gunsnard et al., 2001

Resting State Functional Connectivity

- Examples of resting state applications include studying brain function in healthy controls, infants, patients with brain injuries and in psychiatric disorders

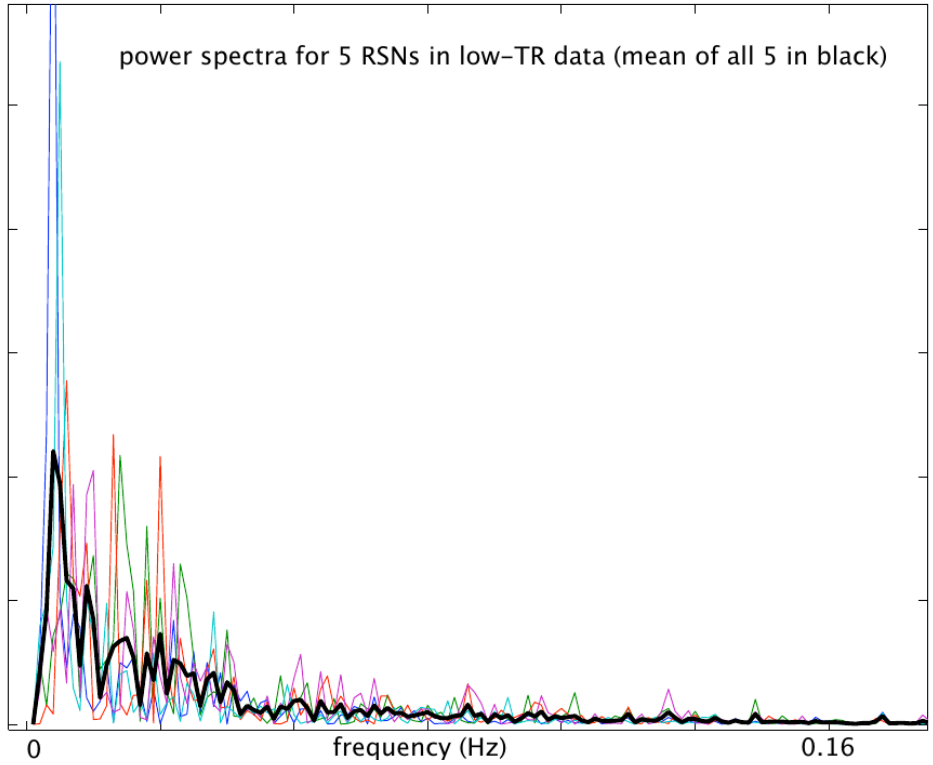


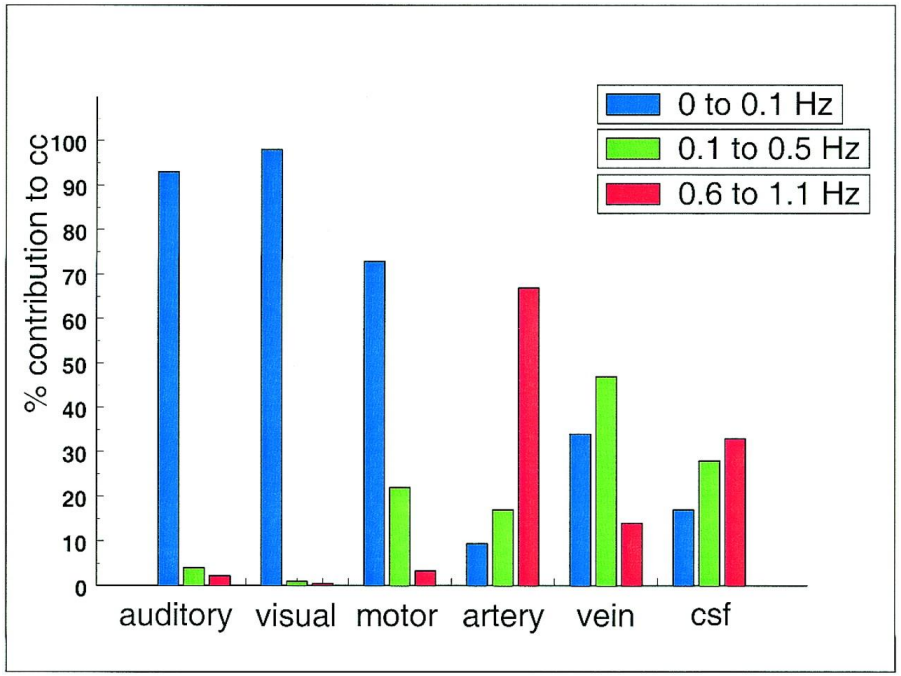


Homae et al., *The Journal of Neuroscience*, 2010
Eggebrecht et al., *Nature Photonics*, 2014

Low-frequency fluctuations dominate resting-state connectivity

Signal power is strongest at low frequencies





FSL course lectures, Cordes et al. (2001)

Resting State Functional Connectivity

Design considerations:

- Typical study length: 5-10 minutes, 10-15 minutes is better
- Wang et al., 7 minutes, but as low as 2.5 minutes
- Eyes open vs eyes closed



fMRI for newbies, Jody Culham
Abdalmalak et al., Front. In Neuroscience, 2020
Finn., *Trends in Cognitive Sciences*, 2021
Wang et al., *Scientific Reports*, 2017

How can we preprocess and analyze resting state data?

Aim of preprocessing

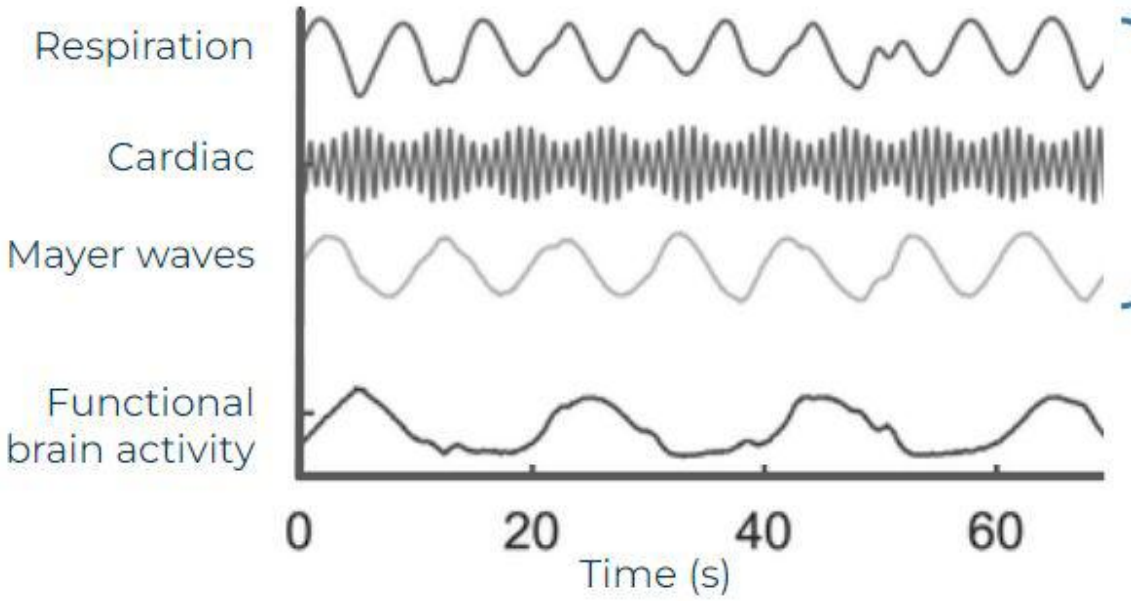
1. **Functional brain activity**
2. **Physiological noise**
3. **High frequency non-physiological noise**
4. **Low frequency non-physiological noise**
5. **Transient artifacts**
6. **Detector over/undersaturation**

Preprocessing aims to **isolate** the **functional brain activity** from the noise and artifacts

fNIRS Signal Components

1. **Functional brain activity**

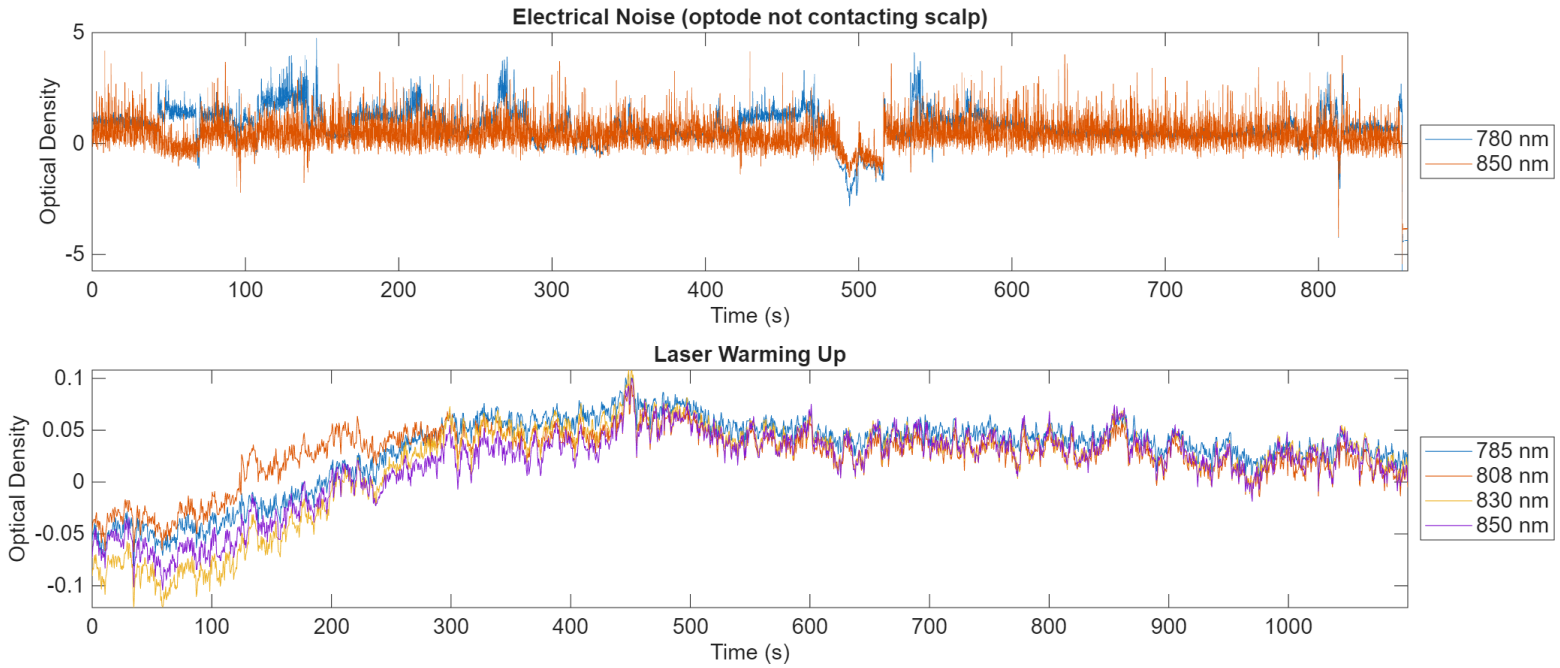
2. **Physiological noise**



fNIRS Signal Components

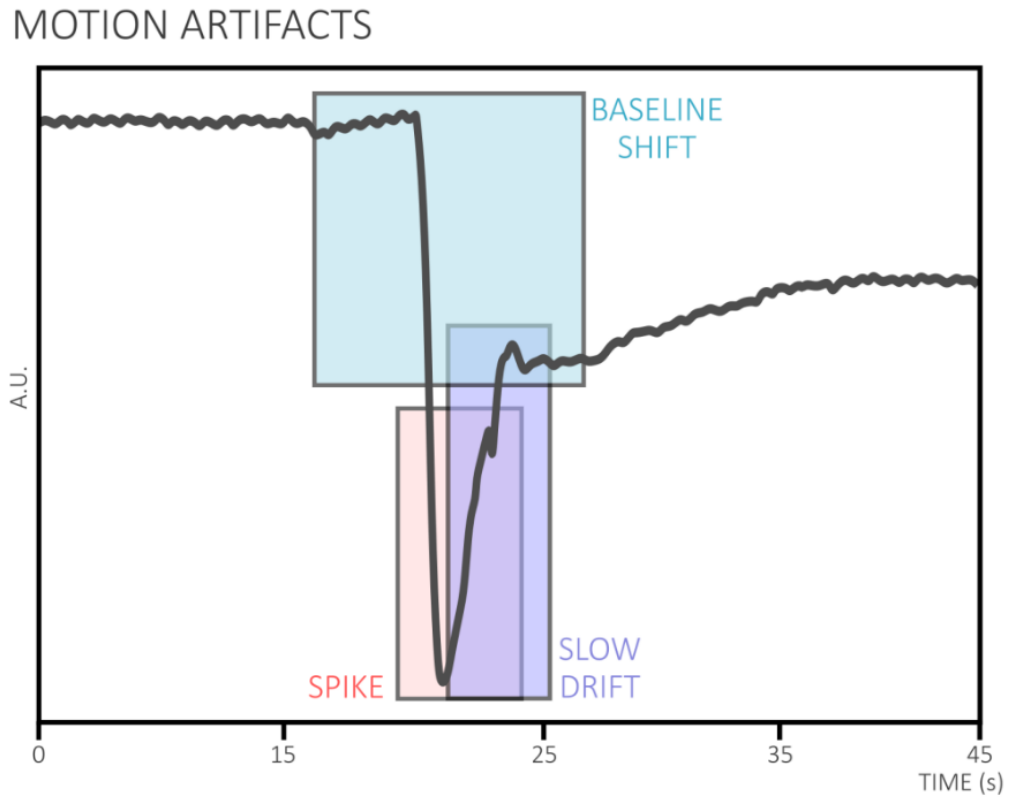
3. **High frequency non-physiological noise**

4. **Low frequency non-physiological noise**



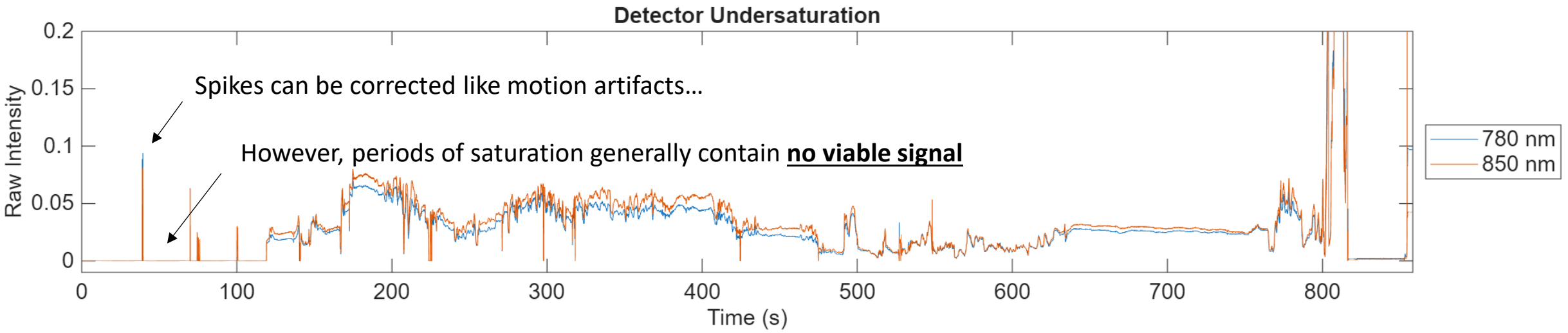
fNIRS Signal Components

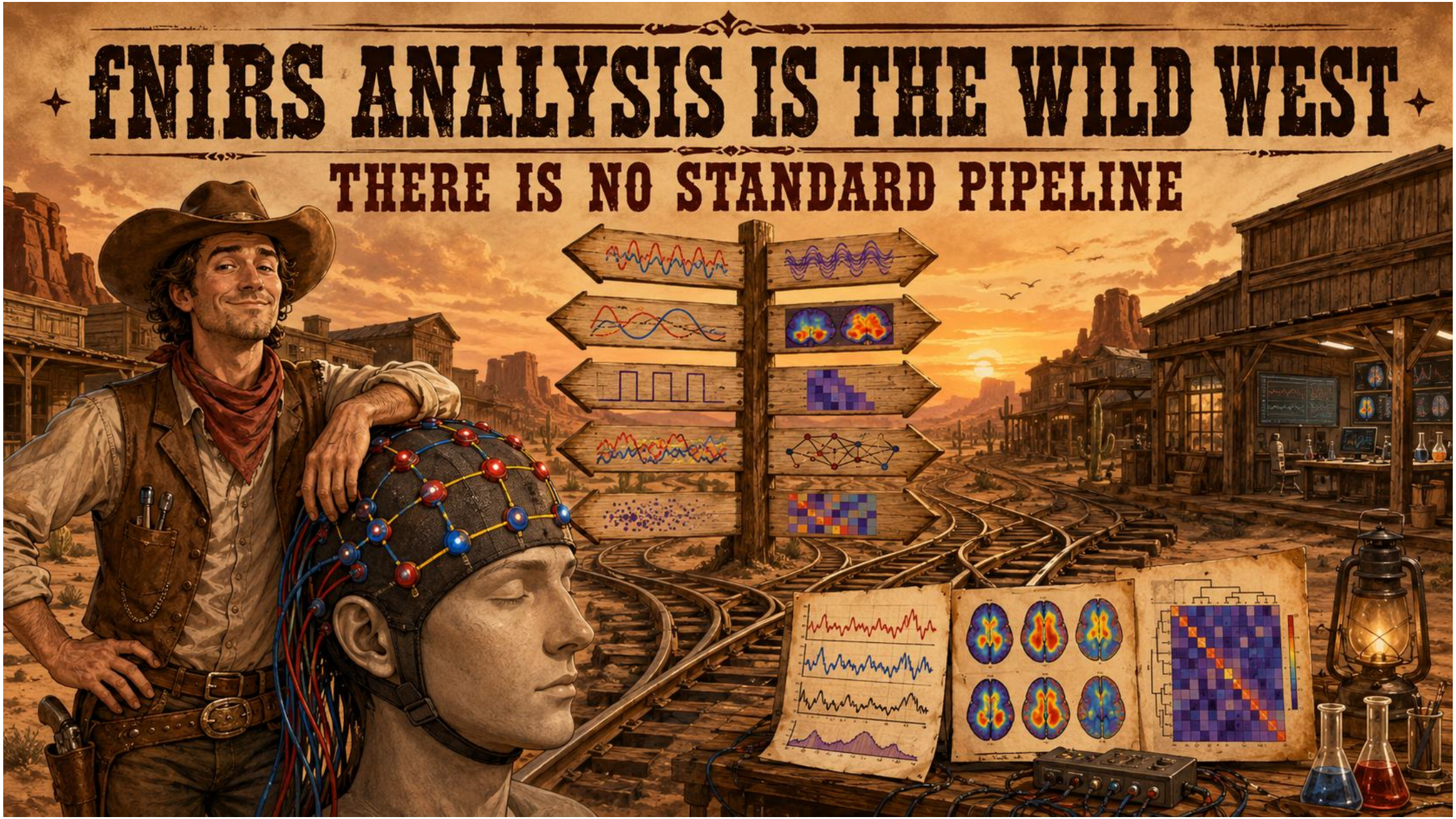
5. **Transient artifacts**



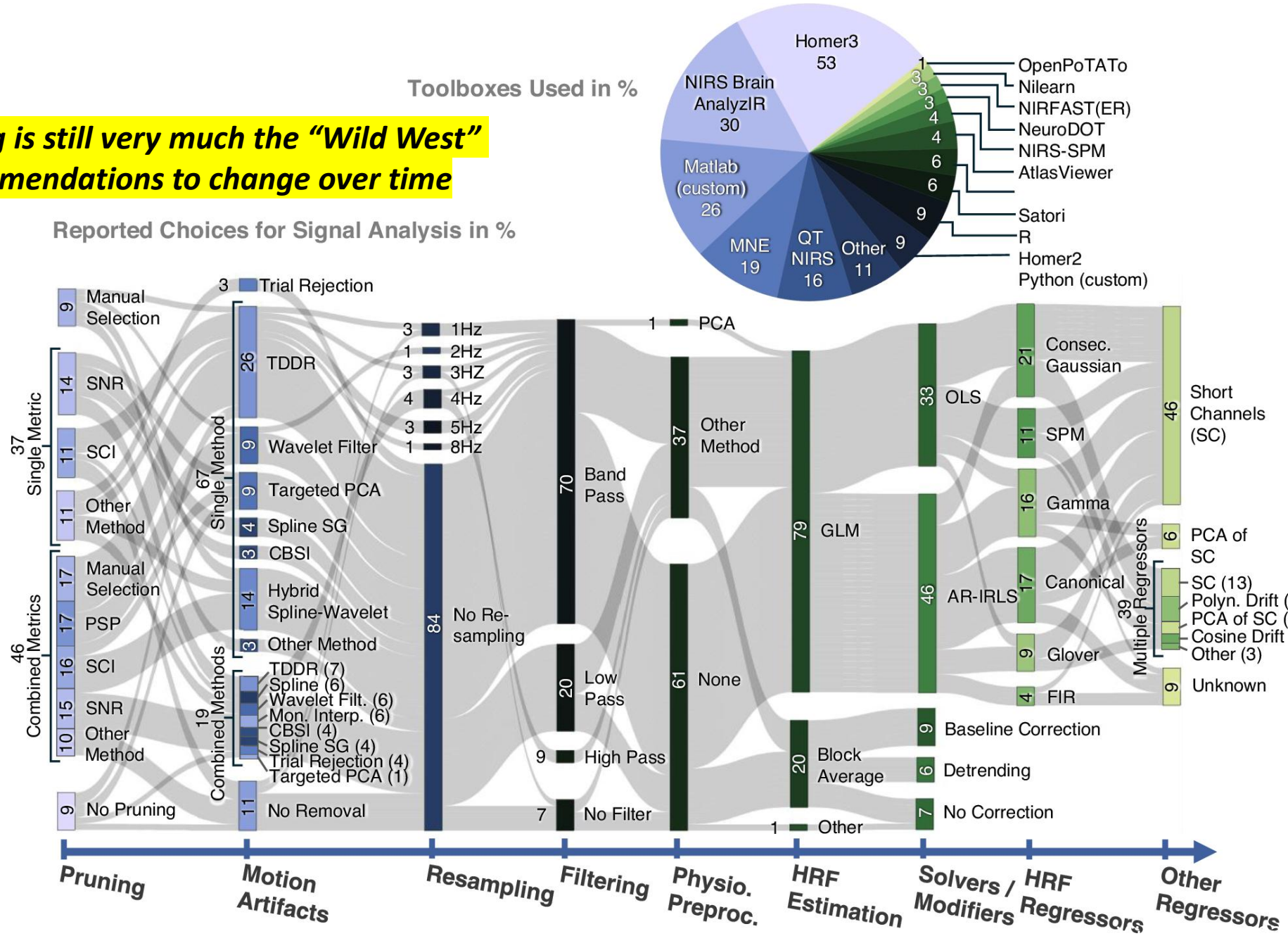
fNIRS Signal Components

6. Detector over/undersaturation




fNIRS ANALYSIS IS THE WILD WEST
THERE IS NO STANDARD PIPELINE

Preprocessing is still very much the "Wild West"
Expect recommendations to change over time



Functional Connectivity with fNIRS

Part 2: Dataset, Toolbox, and GUI

Today's Goals

What we are doing:

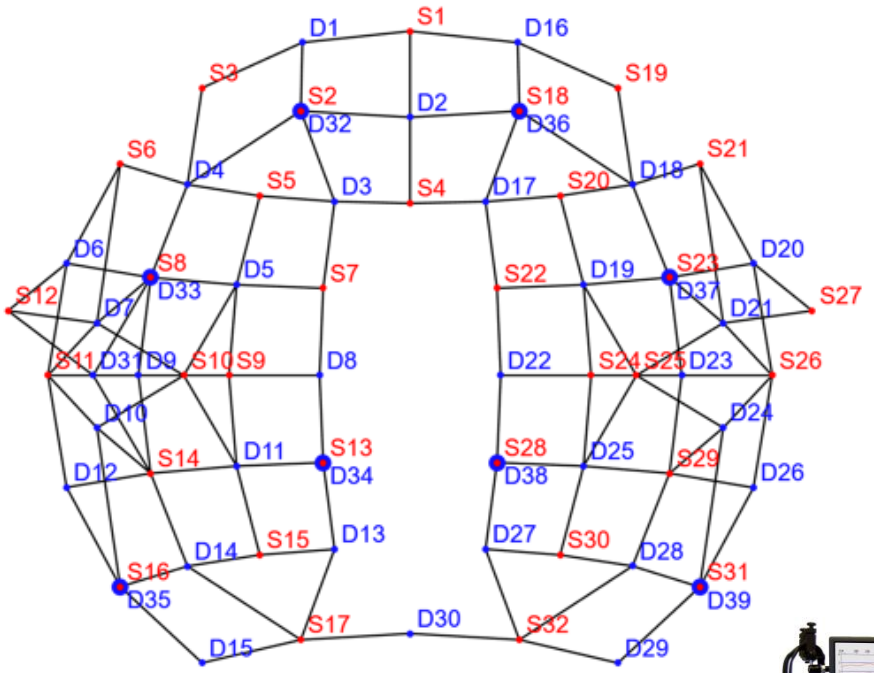
- Focus on preprocessing theory and analytic methods
- Showcasing an extensive preprocessing pipeline that highlights many options/decisions
- Walking through every preprocessing step using publicly available data and our open-source tools
- Reviewing many quality control figures along the way

What we are NOT doing:

- Suggesting that this pipeline is perfect or universally suitable
- Teaching MATLAB or NIRS Toolbox
  - We encourage you to follow along, but unfortunately there won't be time to stop and troubleshoot
- Trying to convince you to use our Toolbox/GUI over any others
  - Although we believe that it is great for new users

Demo Dataset

- N = 2
- Healthy, typically developing adults
  - Low-moderate motion



- ~13 minutes
- Task-free resting state
  - Relaxed, eyes closed

- 3.9002 Hz
- 4 wavelengths: 785, 808, 830, and 850 nm (NIRScoutXP)
- 129 channels (nearly whole head coverage)
  - 8 short channels
  - 121 long channels



fNIRS-Preprocessing Toolbox

Developed by Kevin Stubbs at Western University. MATLAB-based. Builds in NIRS Toolbox.

- **Expands on NIRS Toolbox functions**
  - E.g., MBLL can adjust PPF for age
- **Implements additional processing steps** not presently in NIRS Toolbox
  - E.g., cleanest segment selection
- Organizes outputs into standardized **BIDS derivatives** with incremental suffixes
- **Generates validation figures for <u>every single step</u>**
- **Stores the pipelines in a convenient object format** that can easily be compared/validated and shared. The full details of the steps performed are stored in every data file (enough to replicate)
- **Automates version control** and **manages the NIRS Toolbox version** so that important updates are not missed (updates are currently opt-in rather than fully automatic)
  - Does not require any knowledge of Git
- **Manages the MATLAB path to reduce that chance of conflicts** with other toolboxes
- Accessible: easy to setup and use

Our Brand New GUI

- In addition to the code-based version, we now have a basic GUI alternative
- The provided version is setup for a resting state pipeline with several parameters and optional steps
- Future versions will have the flexibility to add/remove/reorder most steps, which will also enable task-based pipelines
  - Efficiency is also a high priority
- This is an early version so there are still a few rough edges and visual bugs

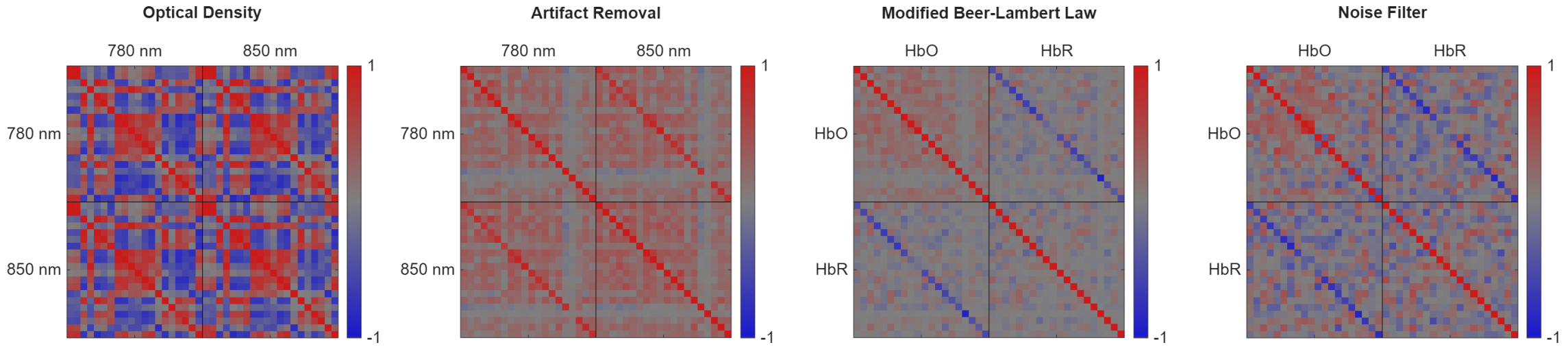


Functional Connectivity with fNIRS

Part 3: Preprocessing

Useful Figures: Correlation matrices

- Rows and columns are both each channel for each wavelength
- Expected flow:
  - **Start** – large correlations due to artifacts and physiology (artifacts can cause negative correlations)
  - **Artifacts Removed** – correlations reduced and mostly positive, still largely driven by shared physiology
  - **MBLL** – distinct HbO and HbR with anticorrelations, still largely driven by shared physiology
  - **Noised Filtered** – final patterns emerge across channels

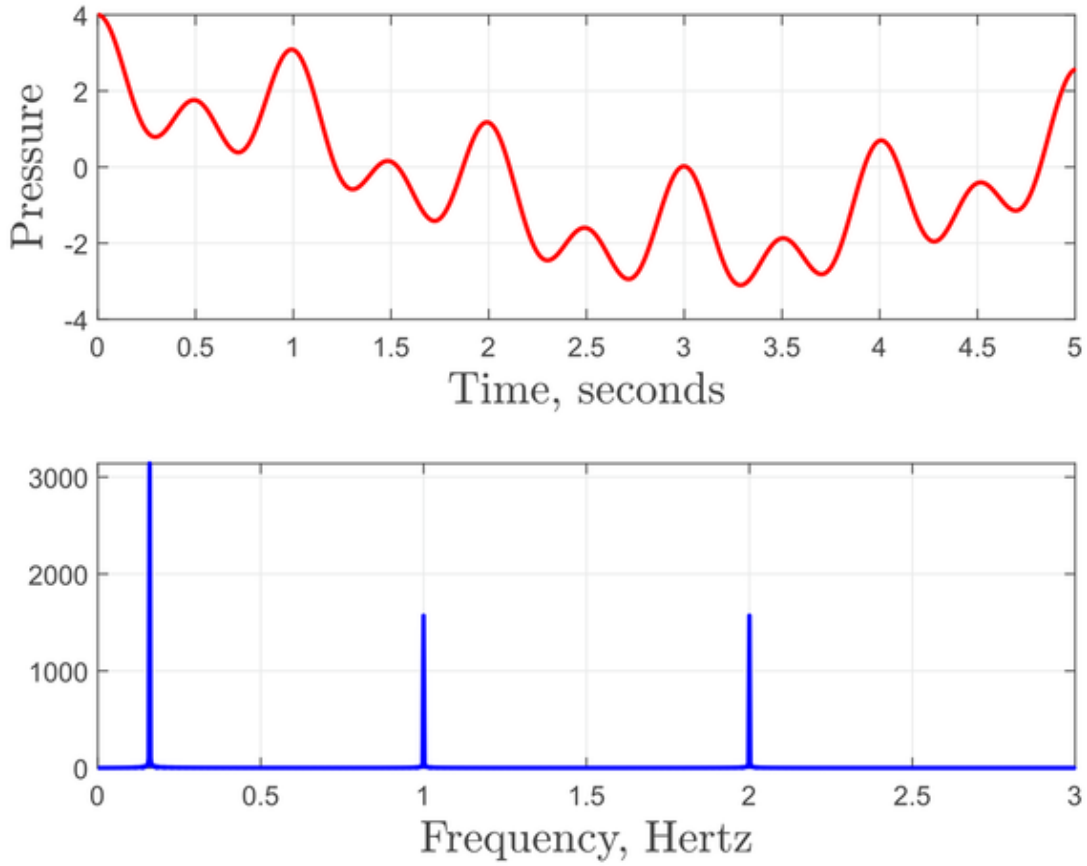


Useful Figures: Fourier transform / power spectrum

Fourier transform produces the **power spectrum** for any timecourse (raw intensities, OD, or concentrations)

Sum of 3 sine waves:

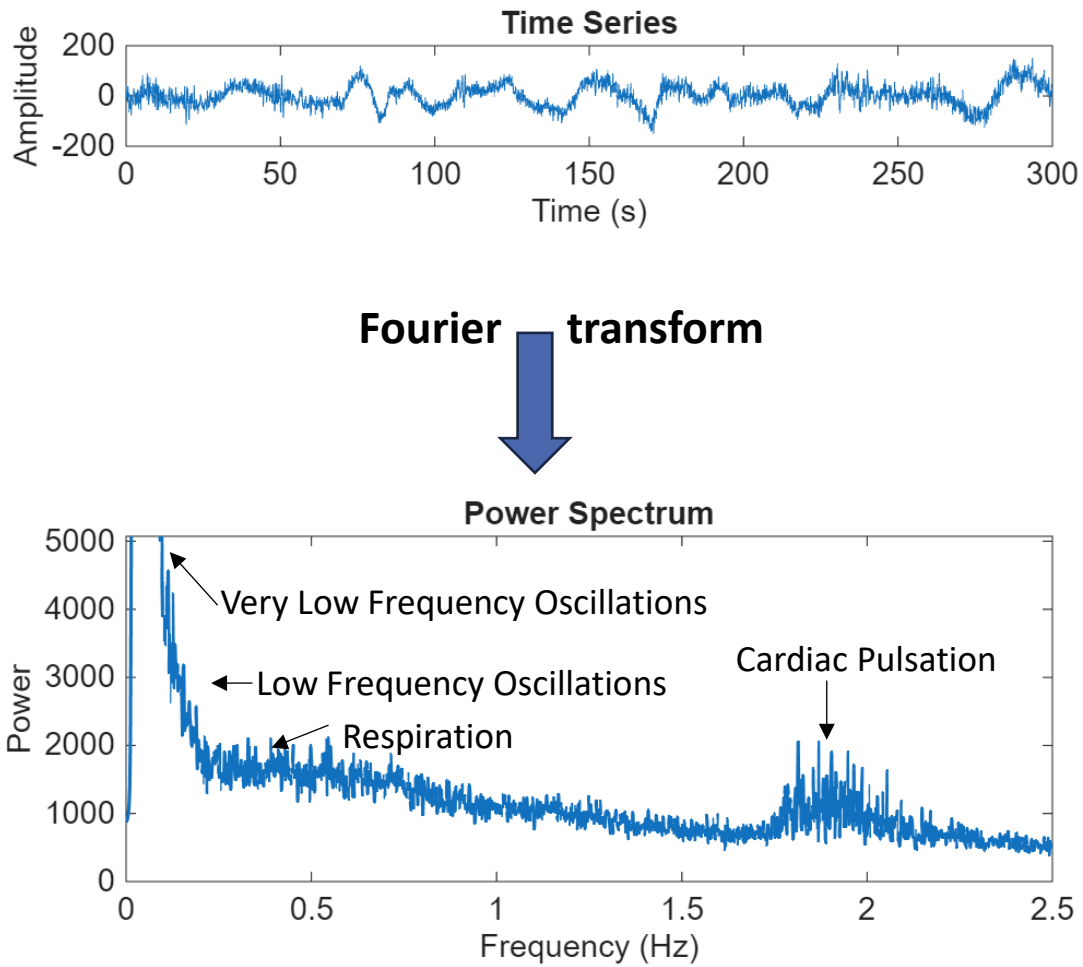
1. Small 2 Hz
2. Small 1 Hz
3. Large 0.2 Hz



Useful Figures: Fourier transform / power spectrum

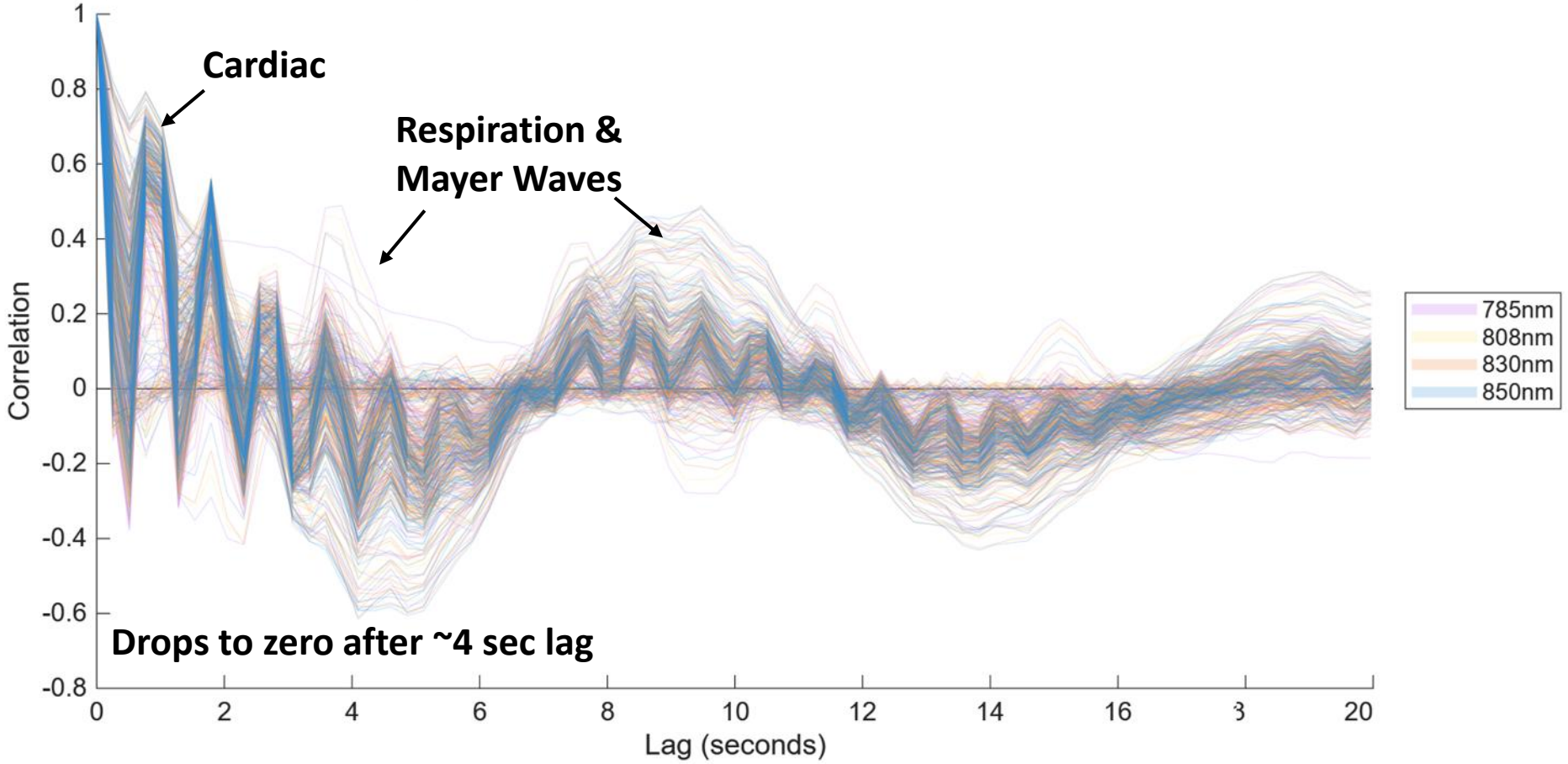
- Quantifies **frequencies up to half of the sampling rate** (*Nyquist frequency*)

- **Uses include:**
  - Assess cyclical noise, which is mostly physiology
  - Estimate the range of the heart rate



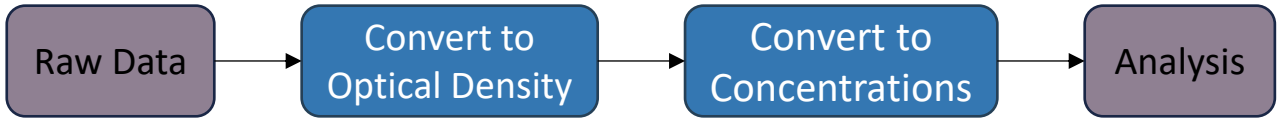
Useful Figures: Temporal Autocorrelations

- A **signal's correlation to itself at incremental lags**
- Takes several seconds to zero out because we are recording slow phenomena (hemodynamics, respiration, Mayer waves)
  - i.e., our timepoints are not independent



Aim of preprocessing

- **Two mandatory conversions**…



1. **Convert raw signals to optical densities (OD)**

   Converts arbitrary raw intensities into **meaningful, linear units**

   Simple logarithmic transformation

   Required for MBLL

2. **Convert optical density to concentrations (HbO and HbR)**

   Converts optical densities into **relative changes in hemoglobin concentrations (HbO and HbR)**

   Uses the *Modified Beer-Lambert Law* (MBLL)

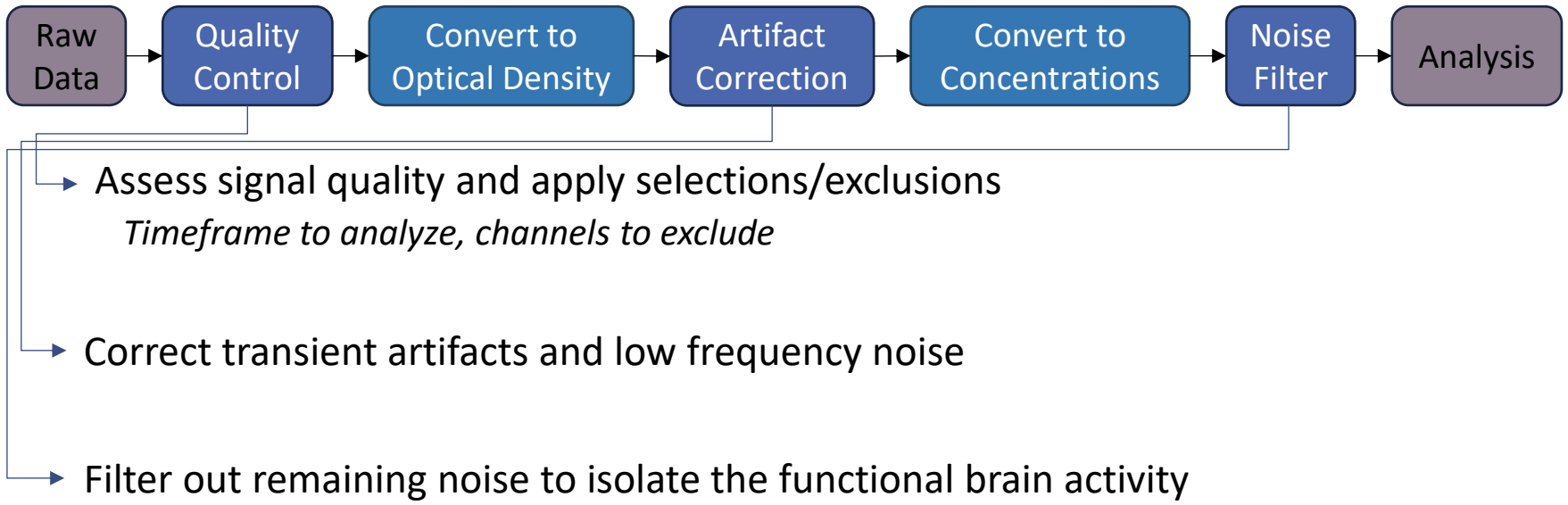
   Assumes certain spectral properties that can be adjusted to account for participant age

   Factors in the channel lengths

All other steps are slotted in between these core conversions.

Aim of preprocessing

- Typically, **three additional steps are included**:



Visual inspection at every stage is critical.

Preparing and importing your raw data

1. Select a head template based on age, which is used to scale channel lengths to head size
2. Trim any noise at the start/end that should not be processed (e.g., talking)
3. Save to BIDS format

Imported data should contain everything that will be required for preprocessing and analysis.

A	B	C	D	E	F	G	H	I	J
Include	Path_To_Raw	Subject_Number	Session_Number	Run_Number	Cardiac_Min_bpm	Cardiac_Max_bpm	Age_years	Head_Circumference_cm	iqr_override
TRUE	./sub01/sub01/2021-02-23/2021-02-23_001	1	1	1	60	100	35	56	
TRUE	./sub09/2021-02-23_002	2	1	1	60	100	35	56	
TRUE	./sub10/2021-03-01/2021-03-01_001	3	1	1	60	100	35	56	
TRUE	./sub11/2021-03-03/2021-03-03_001	4	1	1	60	100	35	56	
TRUE	./sub12/2021-03-03/2021-03-03_001	5	1	1	60	100	35	56	
TRUE	./sub13/2021-03-10/2021-03-10_001	6	1	1	60	100	35	56	
TRUE	./sub14/2021-03-10/2021-03-10_001	7	1	1	60	100	35	56	
TRUE	./sub15/2021-03-16/2021-03-16_001	8	1	1	60	100	35	56	
TRUE	./sub16/2021-03-16/2021-03-16_001	9	1	1	60	100	35	56	
TRUE	./sub17/2021-03-17/2021-03-17_001	10	1	1	60	100	35	56	

Quality control:
Scalp Coupling Index (SCI) and Peak Spectral Power (PSP)

Principals:

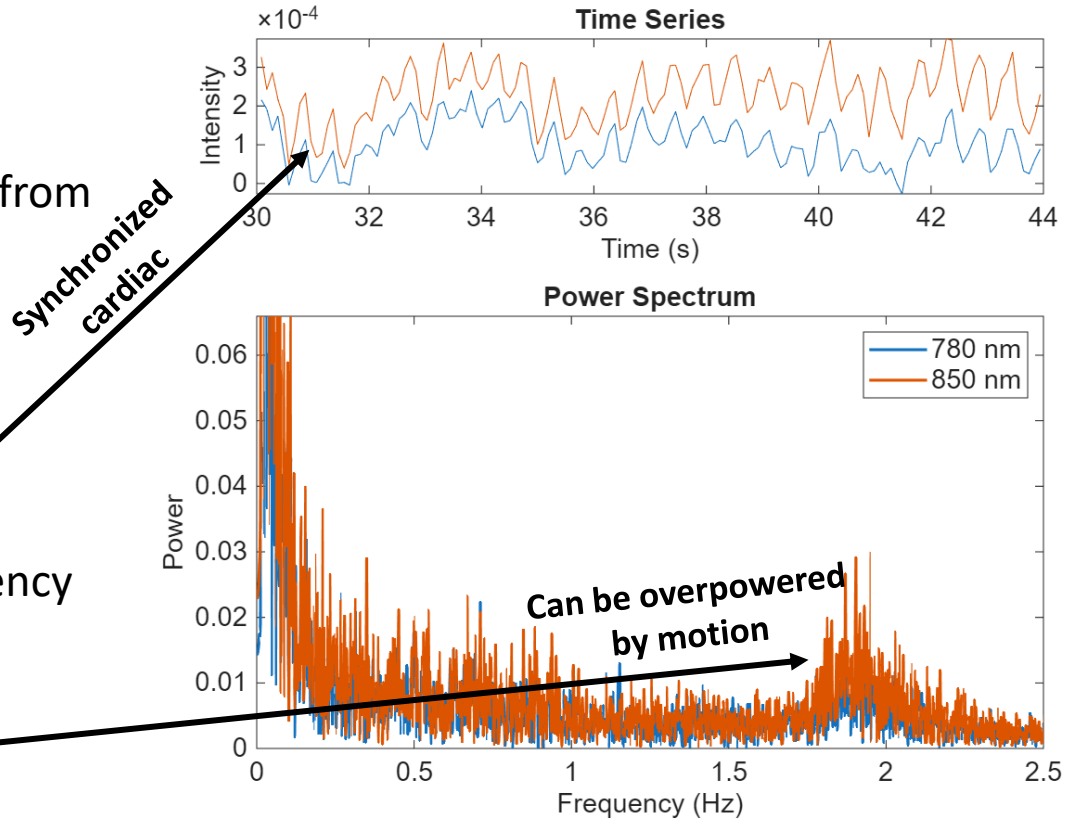
1. High-quality signals contain cardiac pulsation from at least the scalp and ideally also the brain

1. Cardiac pulsation in the wavelengths of each channel is very similar with minimal lag

   Quantified by **Scalp Coupling Index (SCI)**

2. Transient artifacts introduce large high-frequency components that temporarily overpower the cardiac component

   Quantified by **Peak Spectral Power (PSP)**



Quality control:
SCI and PSP continued

Method

1. Specify the **frequency band of the cardiac component**
2. **Bandpass** all signals around this frequency to isolate cardiac, then normalize
3. For each channel…

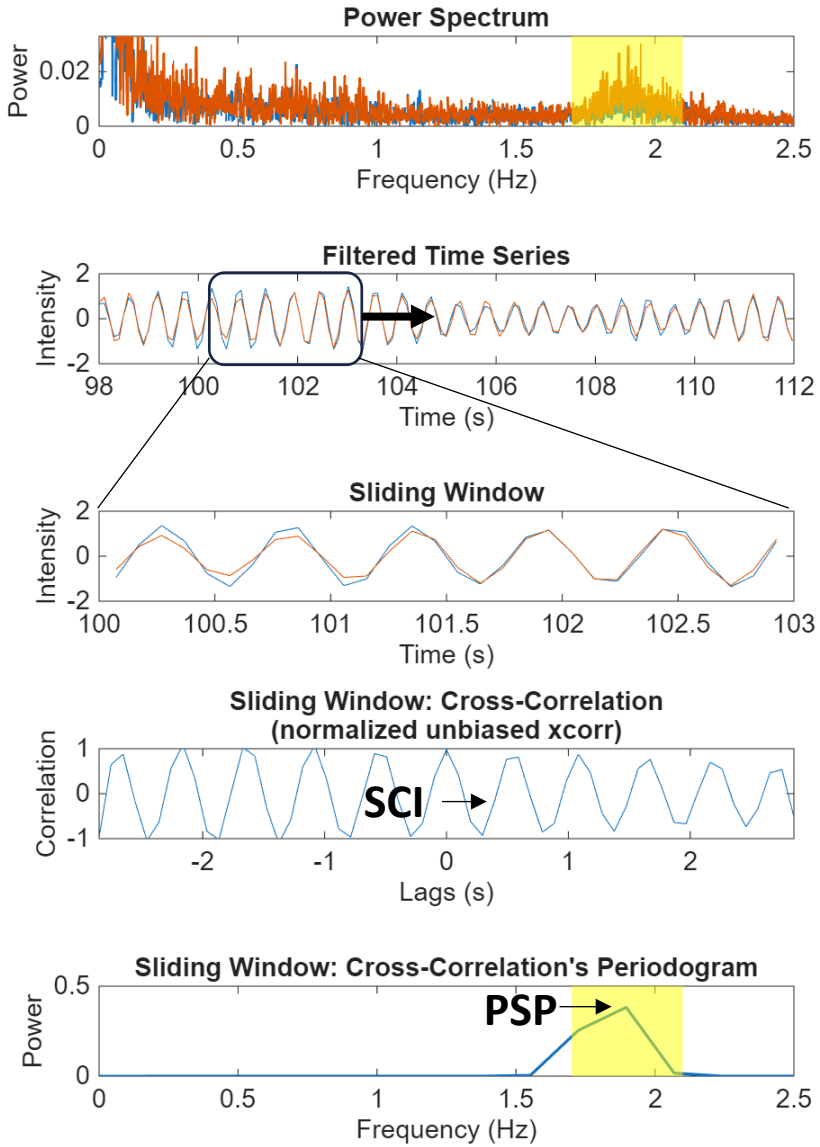
   For each timepoint…

   In a 3s window centered on the timepoint:

   SCI = correlation between wavelengths

   PSP = maximum power in the cardiac frequency band of a periodogram calculated on the cross-correlations

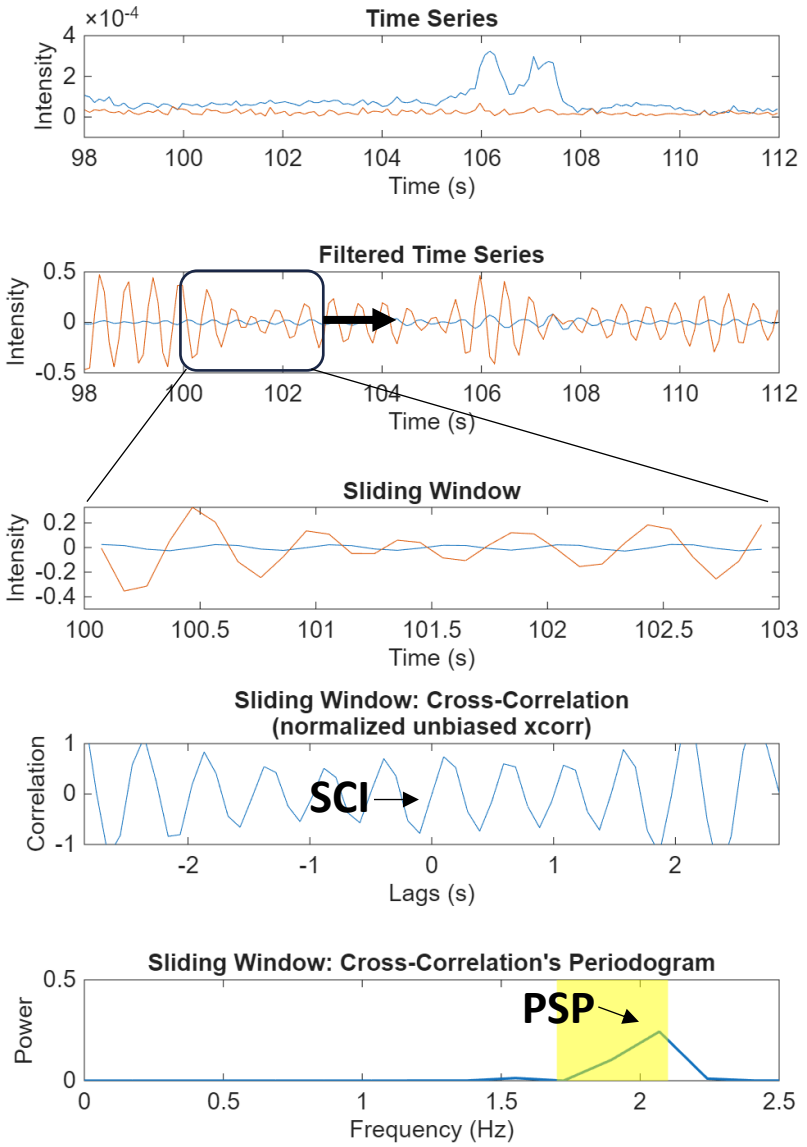
   i.e., extent to which the shifted wavelengths move in/out of phase at the expected frequency
4. **Yields SCI and PSP for each Channel x Timepoint**



Quality control:
SCI and PSP continued

Example: poor contact with scalp

- **SCI** extremely low (0.0130)
- **PSP** moderate (0.2434)



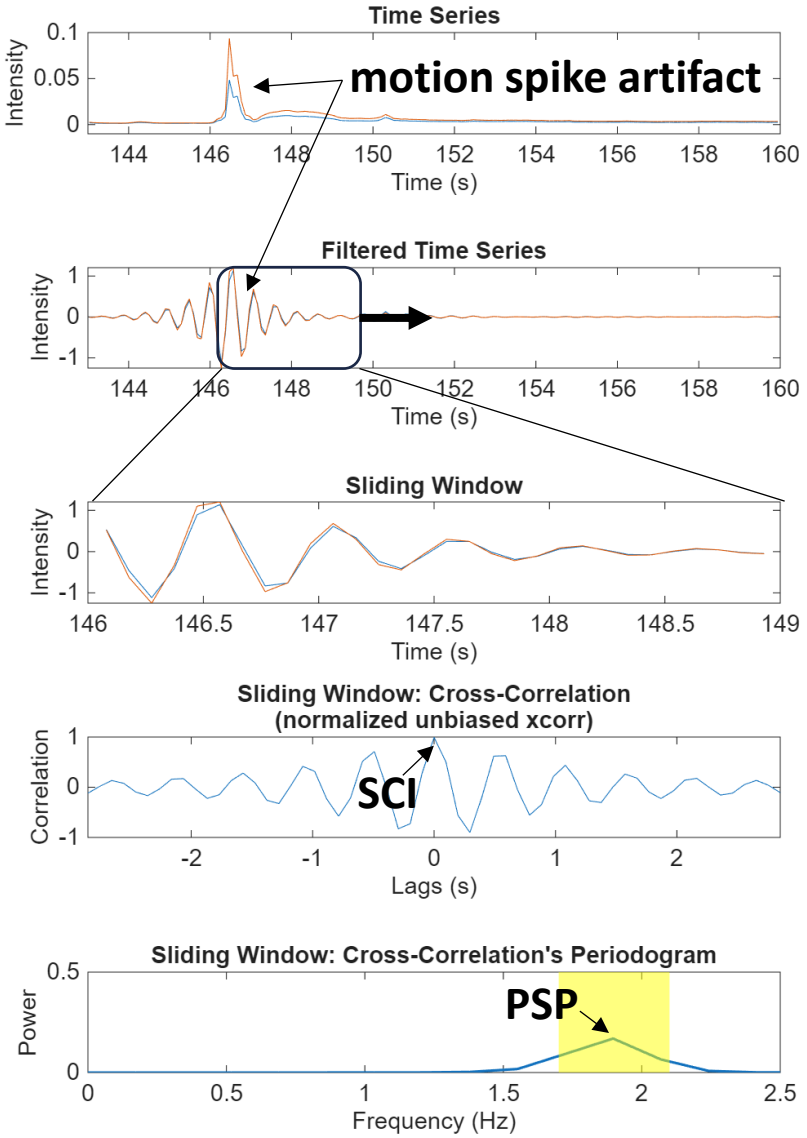
Quality control:
SCI and PSP continued

Example: poor contact with scalp

- **SCI** extremely low (0.0130)
- **PSP** moderate (0.2434)

Example: motion artifact

- **SCI** extremely high(0.9930)
- **PSP** low (0.1692)



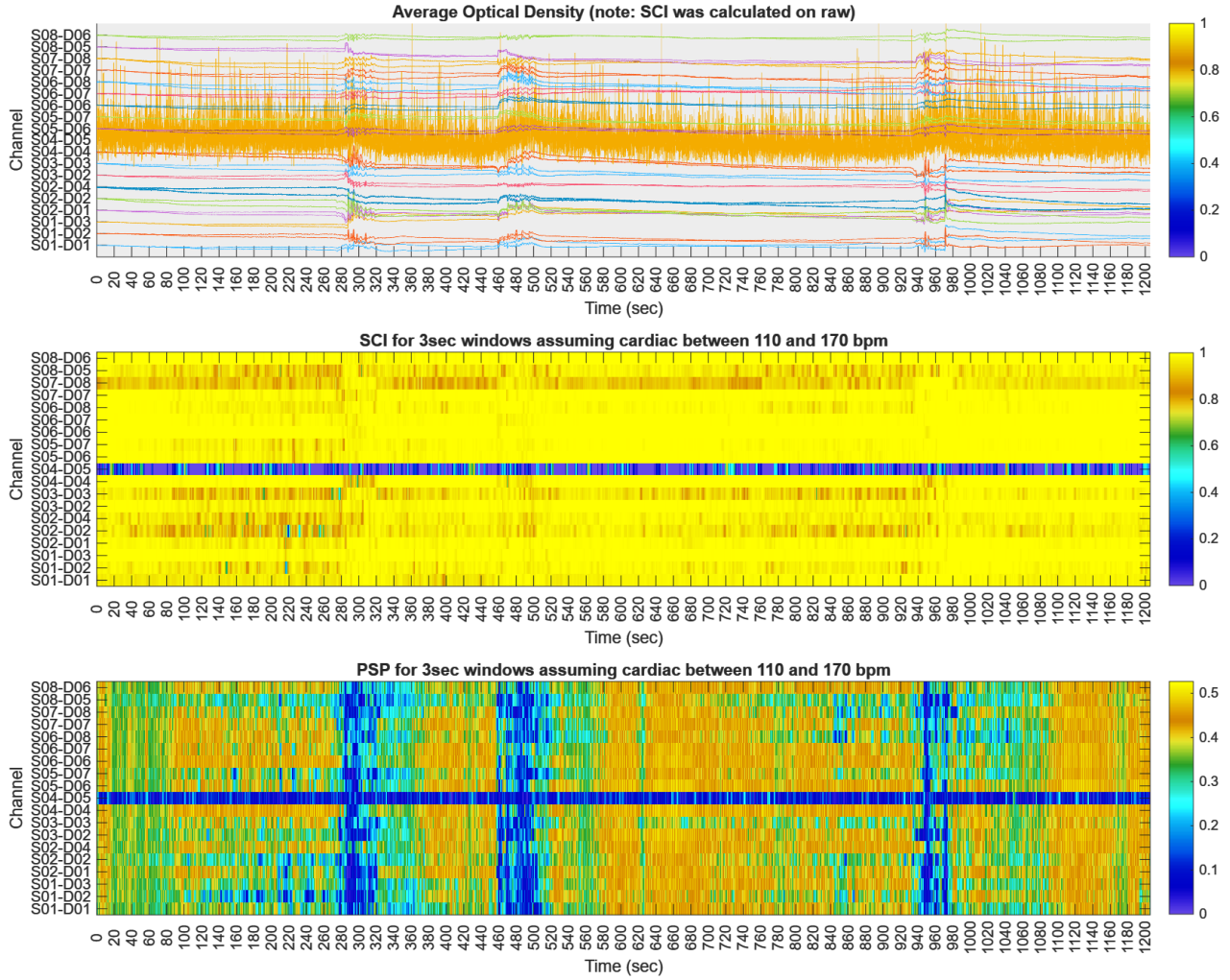
Quality control:
SCI and PSP continued

Example

- Channel S04-D05 had poor scalp contact throughout the acquisition
  - Reliably detected by SCI
  - Sometimes also detected by PSP as it is in this example

- Three large motion artifacts
  - Not detected by SCI

    SCI actually *increased*
  - Reliably detected by PSP



Quality control:
SCI and PSP continued

- SCI and PSP are **best used <u>together</u>**:
  SCI is most sensitive to poor contact with the scalp
  PSP is most sensitive to motion artifacts
  Consider a timepoint to be "high quality" when both **SCI>thresh <u>AND</u> PSP>thresh**

- **Common thresholds**
  SCI > 0.6 (infants) or 0.7 (adults)
  PSP > 0.1

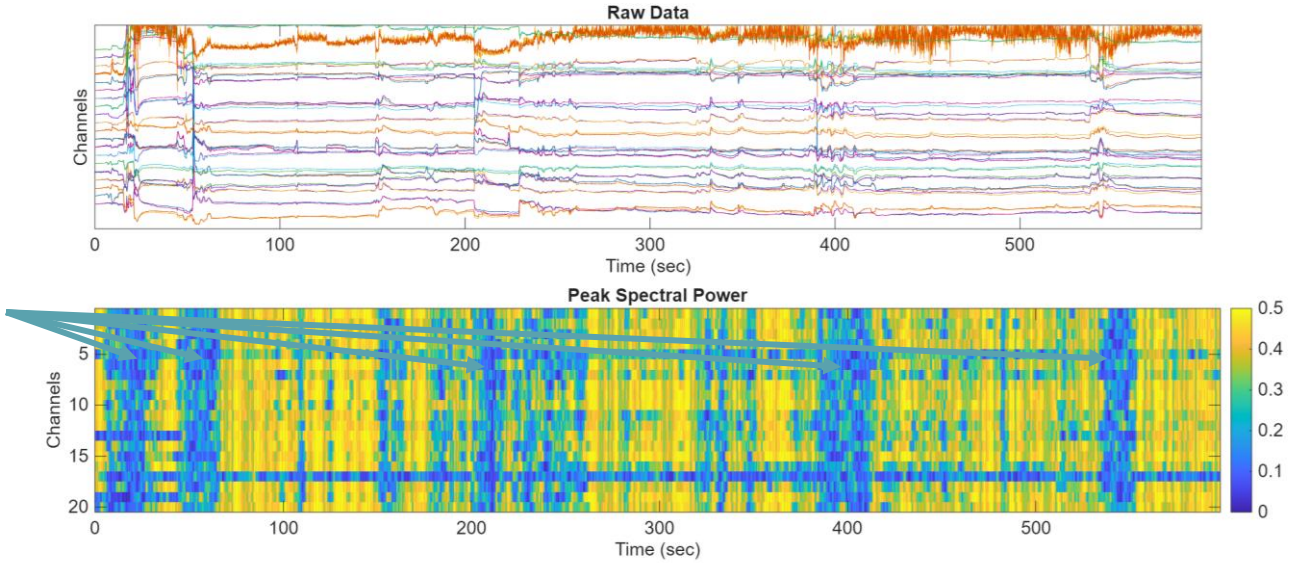
- SCI and PSP should be **calculated on <u>raw intensities</u>** (not optical densities)
  - Calculating on OD can be overly sensitive

- Consider calculating quality metrics on a ***<u>motion-corrected</u> copy of the raw intensities*** to prevent flagging artifacts that will be sufficiently corrected
  - Avoid throwing out salvageable data

Quality control:
Selecting the optimal resting-state timeframe

- **Background**
  - We generally collect more resting state data than we need and then reduce to the cleanest subsample
  - The analysis should use an equal duration from each acquisition
    - Otherwise, the longer acquisitions would be sensitive to additional slow patterns which could greatly alter your results

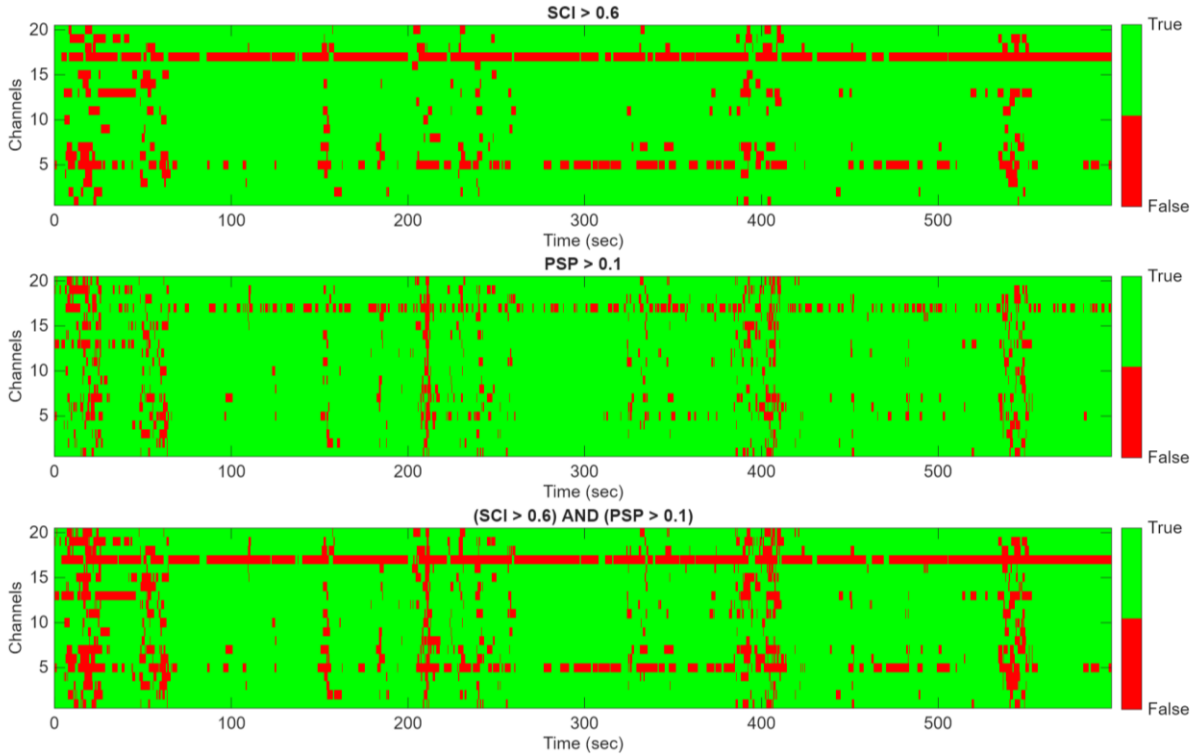
Avoid these as much as possible



Quality control:
Selecting the optimal resting-state timeframe

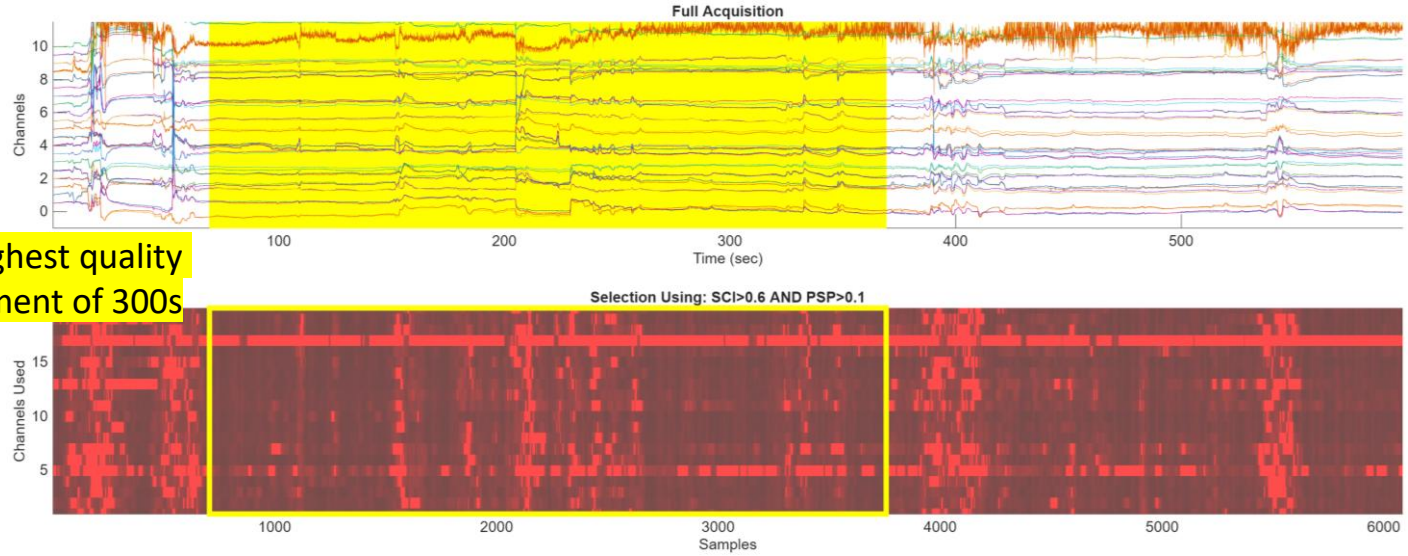
- **Method**
  1. Calculate the Boolean matrix of **SCI>thresh AND PSP>thresh**

     Indicates where/when the signal is sufficiently high quality



Quality control:
Selecting the optimal resting-state timeframe

- **Method cont.**
  1. Calculate the Boolean matrix of **SCI>thresh <u>AND</u> PSP>thresh**
  2. (Optional) Temporarily ignore channels that have very few high-quality timepoints (e.g., <30%)
  3. In a sliding window of the desired length, calculate the overall percentage of timepoints that are high quality across all channels and select the window with the highest overall quality



Bright red indicates failure of one or both thresholds

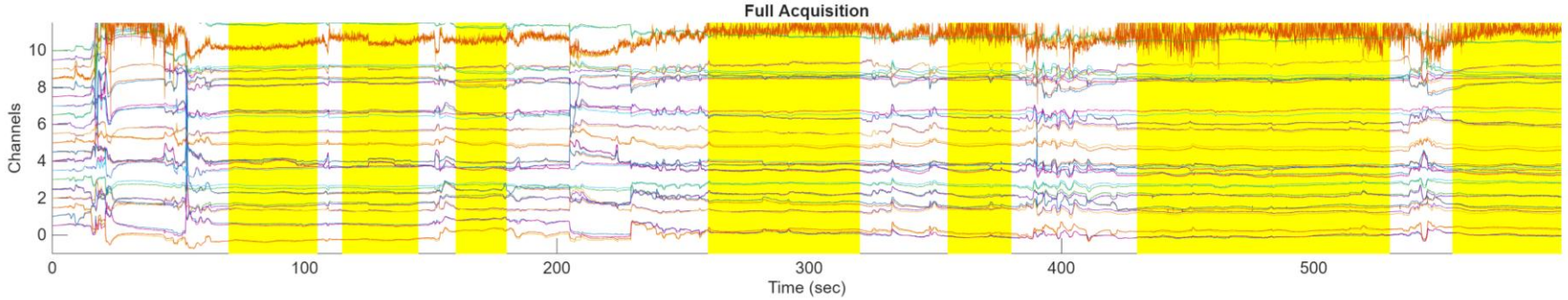
Darker reds indicate proximity to thresholds

Quality control:
Selecting the optimal resting-state timeframe

- There are methods for using multiple non-contiguous subsamples, but this should be a last resort as it is insufficiently supported in the literature. In general:

Select subsamples with a minimum duration (e.g., 60s) that sum to desired total duration (e.g., 600s)

- Independently preprocess the subsamples
- Normalize and concatenate before analysis



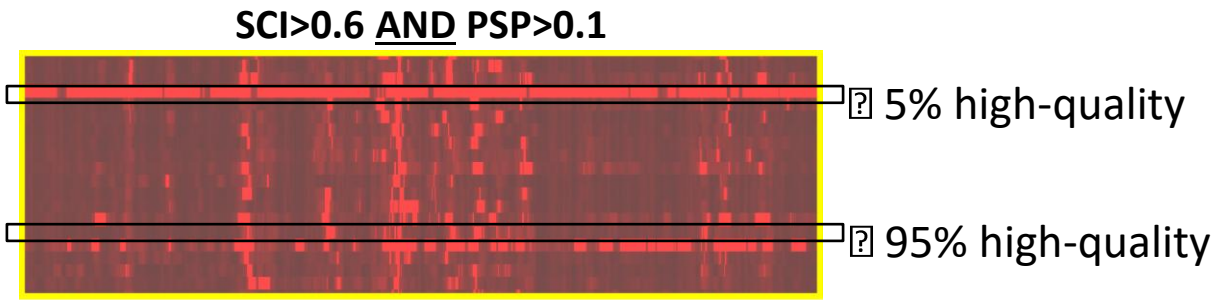
Quality control: Excluding channels

- **Method**

  Use the Boolean matrix of **SCI>thresh AND PSP>thresh** that was generated during subsample selection

  Exclude channels that do not reach a minimum threshold of quality (e.g., <70% of timepoints)

Note that the actual matrix is Boolean. This shaded matrix is for visual inspection only.

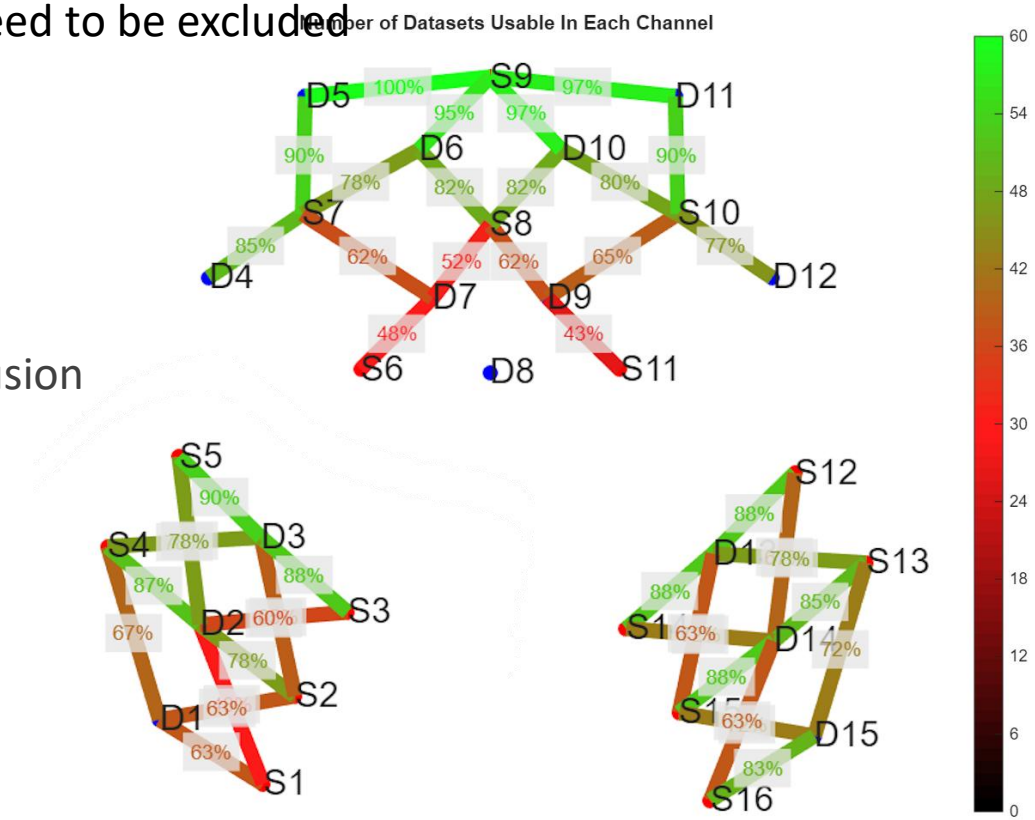


Quality control: Final thoughts

- Consider **excluding an entire acquisition** if:
  - The best subsample is not sufficiently high quality overall, or
  - Too many channels (or a critically important channel) would need to be excluded

Check for:

- spatial patterns in channel exclusions
  - e.g., systemic cap fit issue
- whether any participant variables are related to acquisition exclusion
  - e.g., disproportionally excluding younger participants



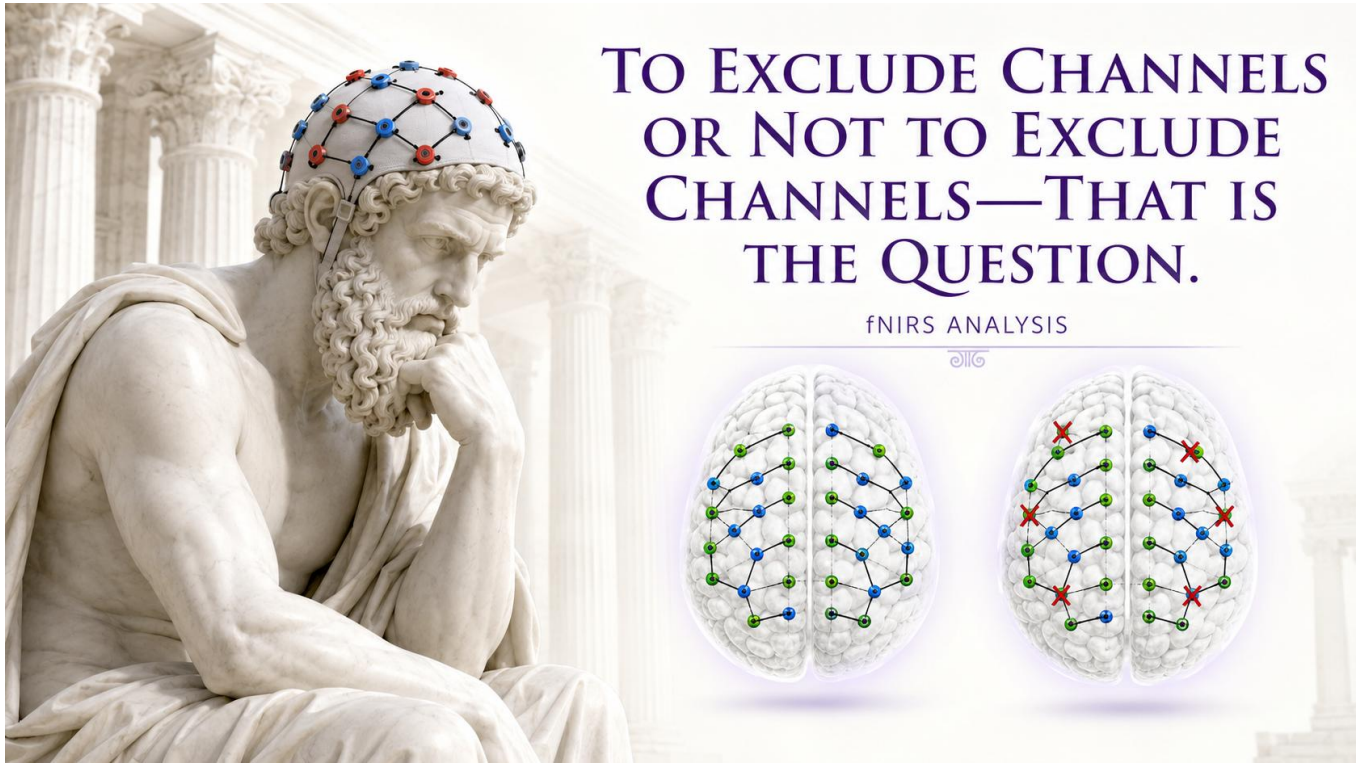
Quality control: Final thoughts

Whether or not to exclude channels is a matter of personal philosophy:

1. **Keep all channels trusting that the noise will wash out** over a large enough sample

2. **Exclude channels that are extremely low quality** as they risk negatively impacting the analysis

Note that some analyses may require you to keep all channels (e.g., graph theory)



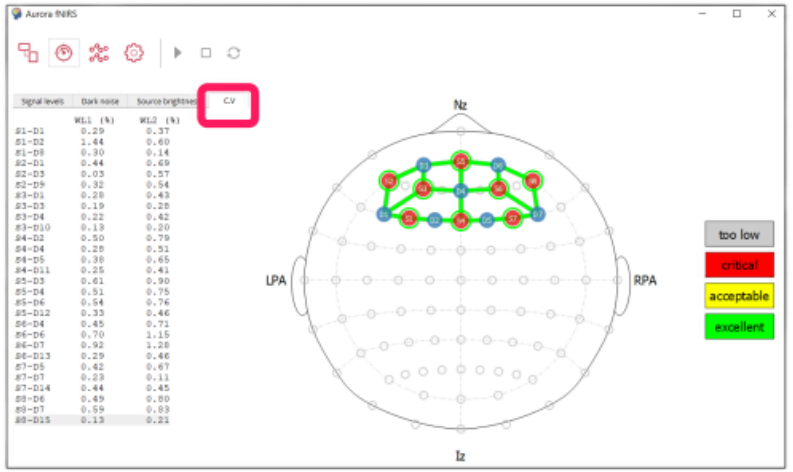
Quality control: SCI/PSP or CV?

Considerations for SCI/PSP

- Appropriate HR window is important
- Wavelength-dependent
- Can be rather conservative
- Threshold is arbitrary

Coefficient of variation (CV)

- Simple to implement
- Less conservative than SCI/PSP
  - Some noisy channels can pass through
- Threshold is arbitrary



CV = standard deviation / mean

too low	
critical	> 7.5
acceptable	> 2.5 and < 7.5
excellent	< 2.5

Unit = %

Temporal signal-to-noise ratio (tSNR) is the inverse: mean / standard deviation

Converting to optical density

- Converts arbitrary raw intensities into **meaningful, linear units**
  Required for MBLL

- No parameters and no notable issues to watch for
  Note that signal dropout will be changed from very low values to very high values

$$OD = -log10(\frac{raw\ signal}{mean\ of\ raw\ signal})$$

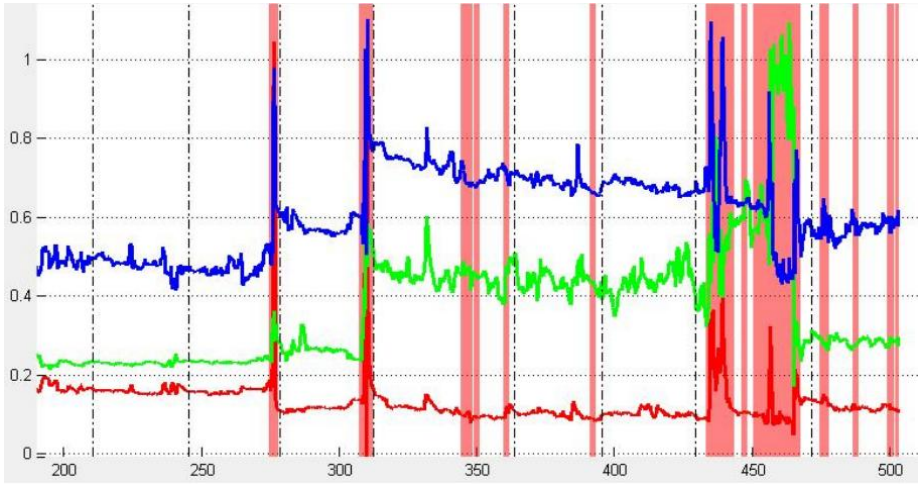
Removing artifacts: Spline interpolation

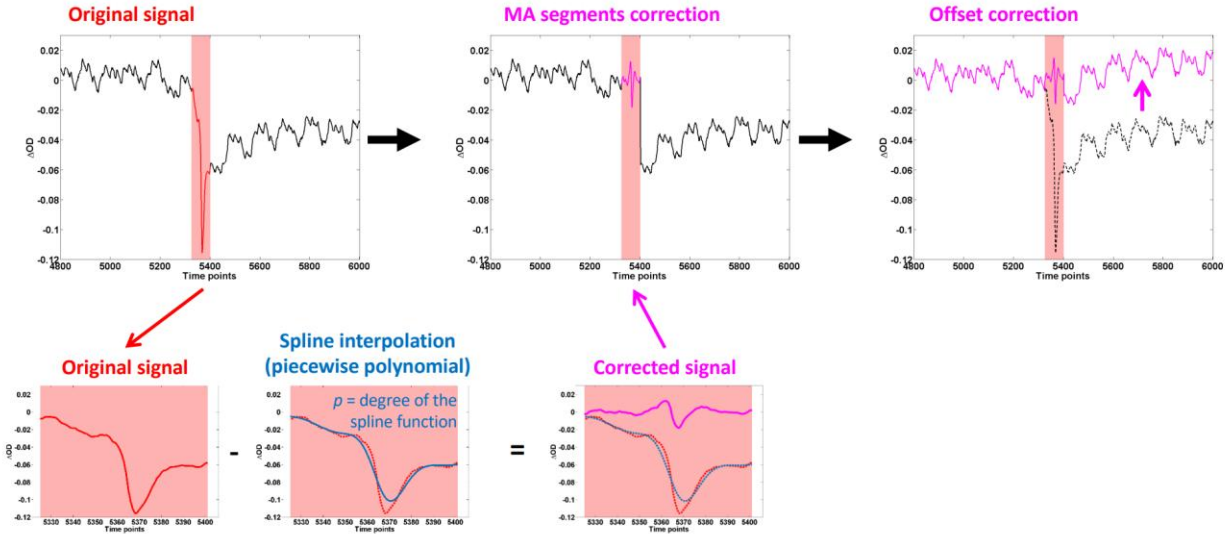
- **Overview**
  - Historically, very widely used
  - Surgical approach – corrects each artifact independently instead of modifying the entire timeseries
  - For each artifact:

    Fit a spline to the window and nearby samples

    Interpolate the flagged values using the spline function

    Shift <u>all</u> downstream samples up/down to correct any offset



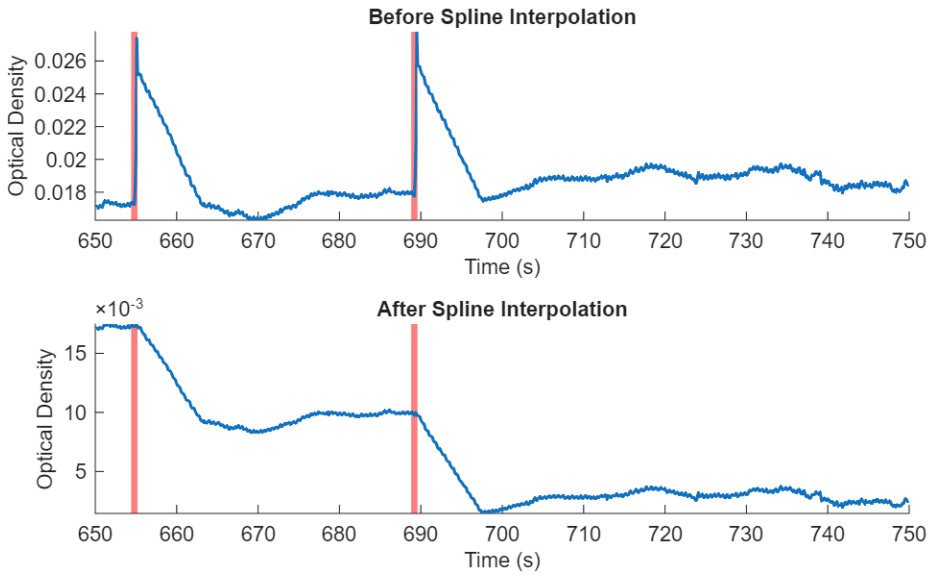


Removing artifacts: Spline interpolation

- **Two main problems**

1. Spline parameter must be optimized for each recording
2. Can introduce new baseline shifts when spike artifacts are followed by a prolonged drift

   Flagging the entire spike + drift is not always feasible as it may span for 10s of seconds

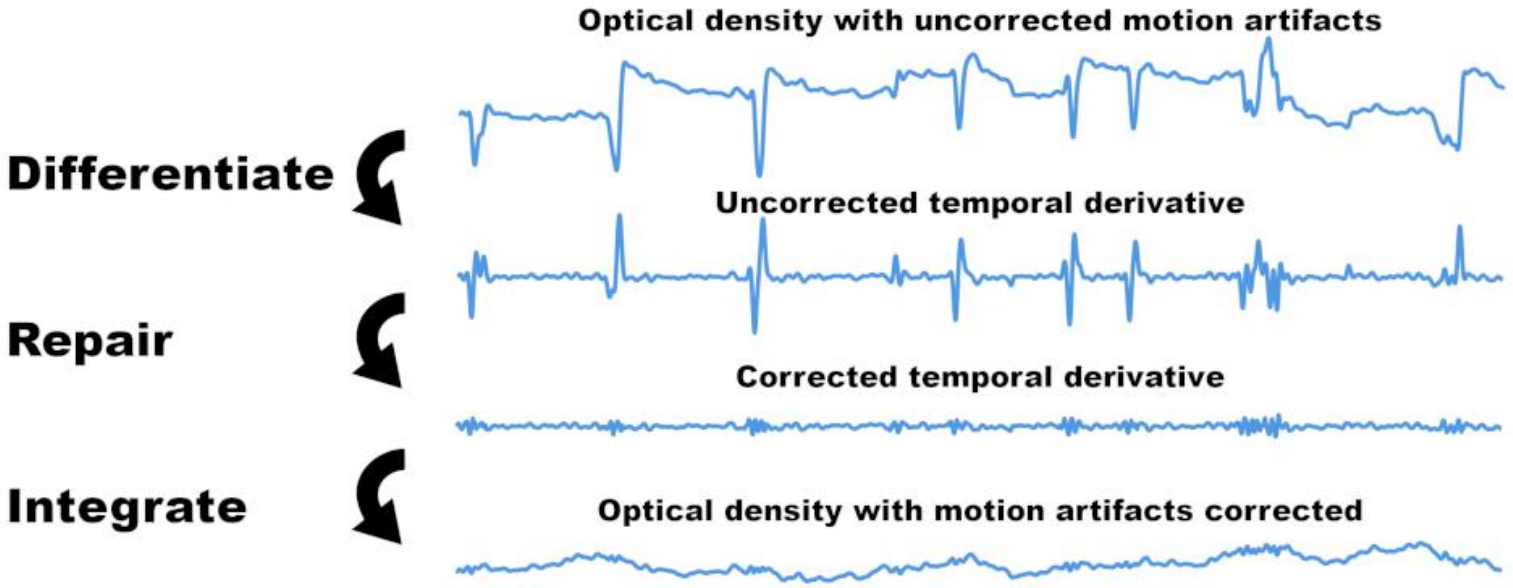


Uncorrected drift results in downstream baseline shifts

Should be used like a scalpel, not like a sledgehammer

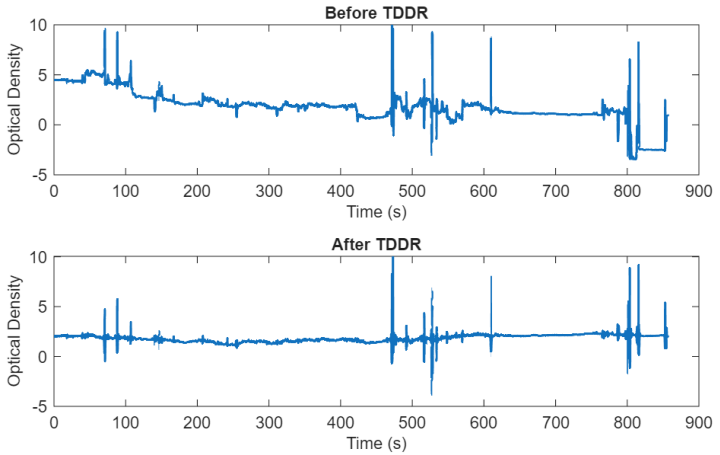
Removing artifacts:
Temporal Derivative Distribution Repair (TDDR)

- **Overview and method**
  - Newer method that modifies the entire timeseries
  - Operates on the **temporal derivative of the signal**
  - Repairs by **down-weighting high frequency components**
  - No parameters, entirely data-driven

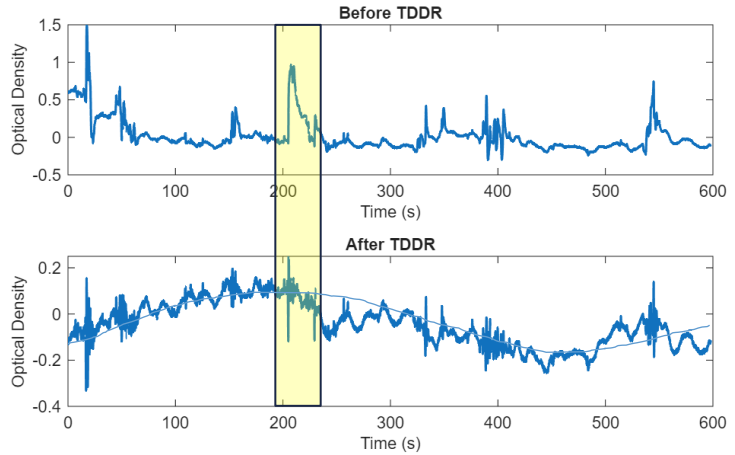


Removing artifacts:
Temporal Derivative Distribution Repair (TDDR)

- **Pros**
  - Excellent at correcting baseline shifts
  - Good at correcting drift
  - Can corrects or reduce minor spikes
  - No parameters, entirely data-driven
- **Cons**
  - May introduce a slow linear trend or oscillation



Example 1: several baseline shifts are corrected

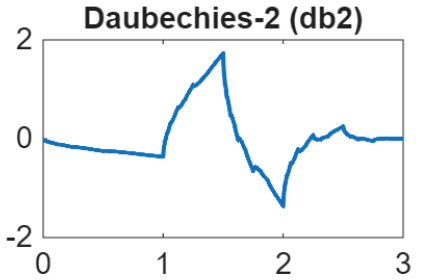


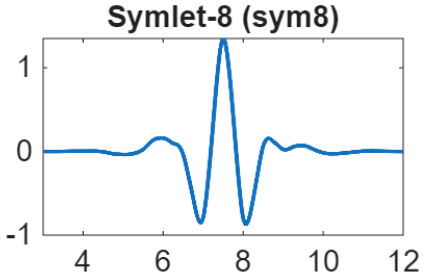
Example 2: spike with drift corrected, oscillation introduced

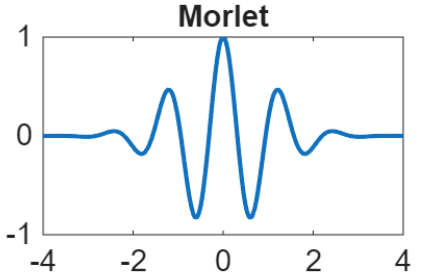
Removing artifacts: Wavelet filter

- **Wavelets**
  - Short, wave-like oscillations
  - Start and end at zero
  - Many different shapes for different purposes

- **Continuous Wavelet Transform**
  - Transforms from time-domain to a time-frequency domain
    - Represents the signal as the **sum of many copies of the mother wavelet** that have been **scaled to select for different frequencies**

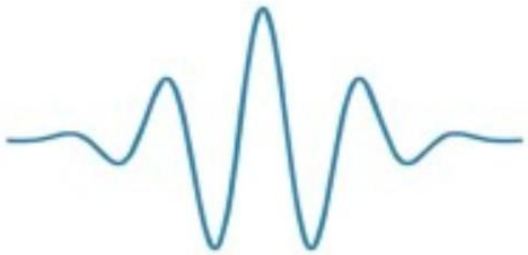






Removing artifacts: Wavelet filter

Helpful video to understand the wavelet transform (see 10:44- 11:15)



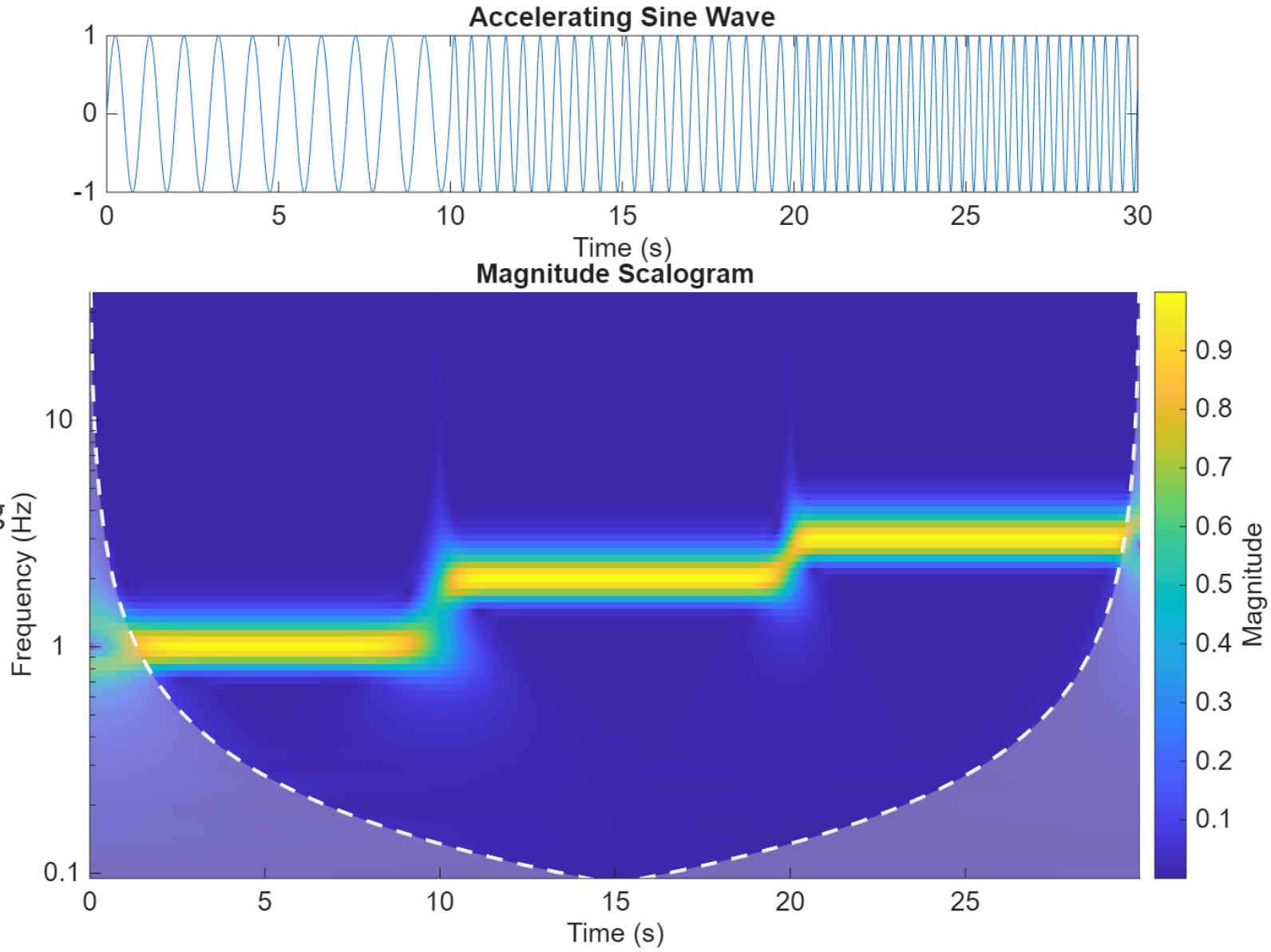
$$F(\tau, s) = \frac{1}{\sqrt{|s|}} \int_{-\infty}^{+\infty} f(t)\, \psi^* \left(\frac{t - \tau}{s} \right) dt$$

https://www.youtube.com/watch?v=kuuUaqAjeoA&t=645s

Removing artifacts: Wavelet filter

Continuous Wavelet Transform

- **Timing is represented** in this space
  - Unlike Fourier transform
- The **frequency axis is exponential**
- Accuracy is low outside of the dashed line (*cone of influence*)
  - Indicates where the wavelet was too long to fit inside the signal
- The bottom row corresponds to a wavelet so scaled so large that it fills the entire signal
  - Can only be fully represented at the midpoint
- Uses mother wavelets specialized in representing smooth signals



Removing artifacts: Wavelet filter

- **Continuous Wavelet Transform**

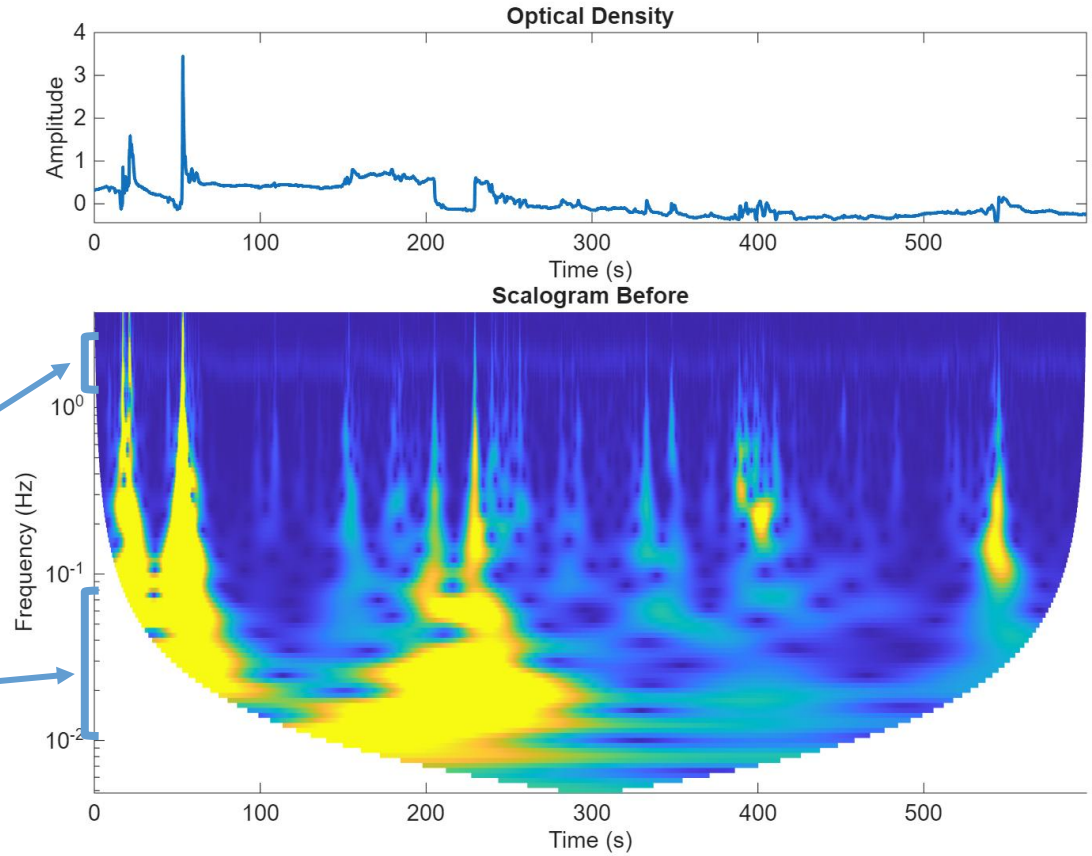
  Artifacts originate at the high frequencies and contaminate lower frequencies

  Scalograms can also be used for visual inspection

  e.g., identify times without cardiac

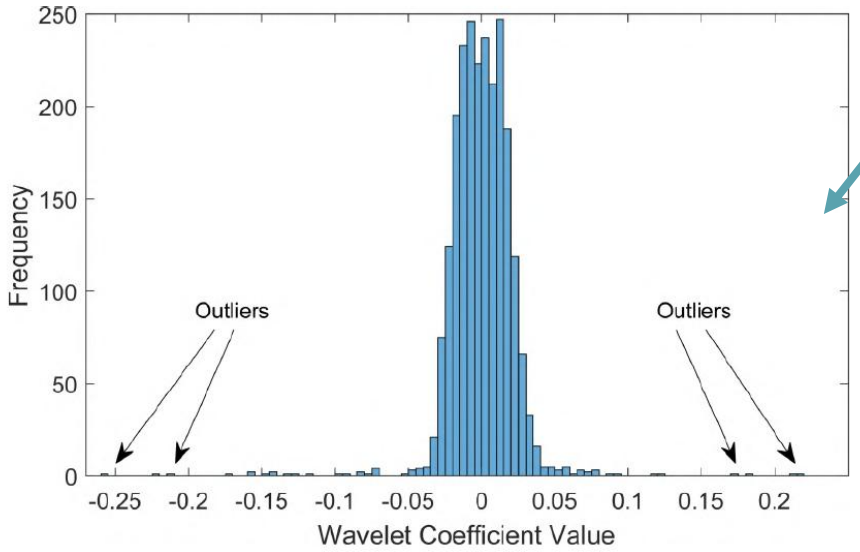
  Cardiac pulsation band

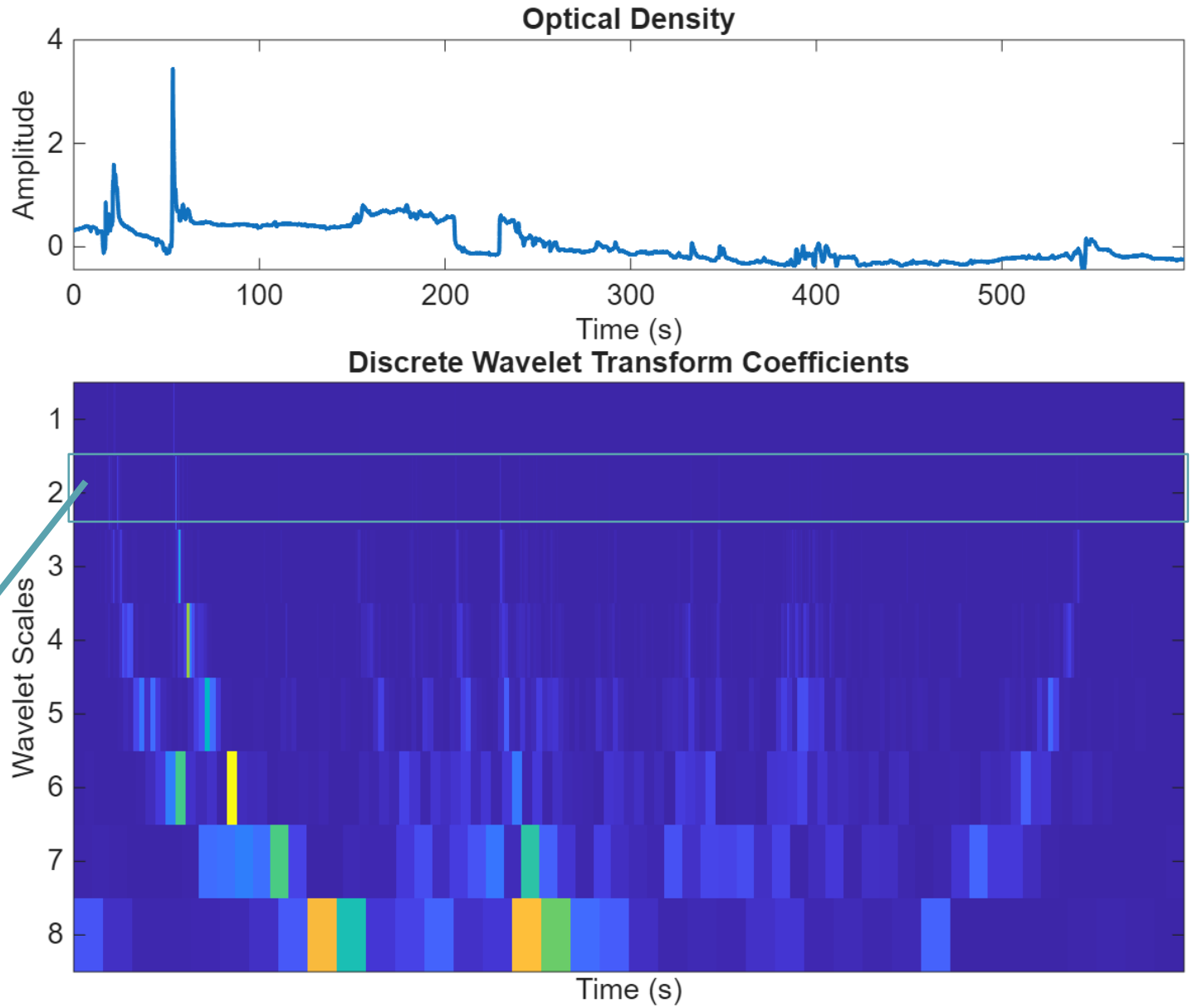
  Resting state activity



Removing artifacts: Wavelet filter

- **Discrete Wavelet Transform**
  - Collapses to a minimal set of frequency bands and timepoints
  - Uses mother wavelets specialized in detecting spikes/edges/etc.
  - Still contains the information required to reconstruct the original signal
  - The distribution of coefficients at each scale is the basis for filtering

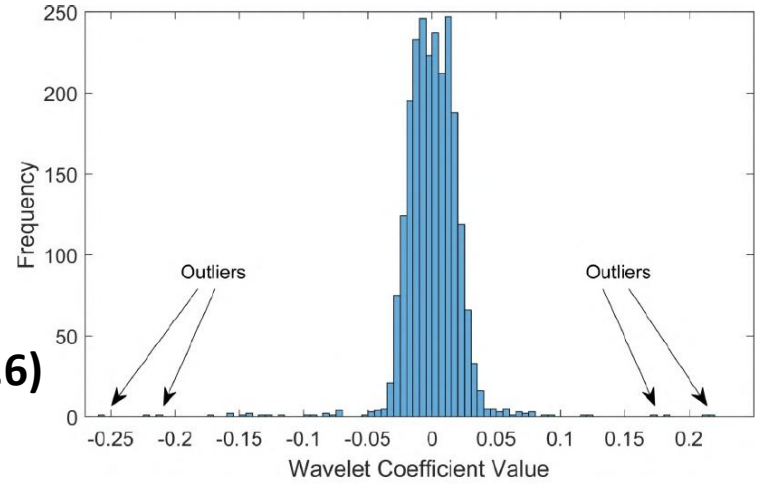




Removing artifacts: Wavelet filter

- Operates on the distribution of coefficients in each scale of a discrete wavelet transform

- Outliers represent **shapes that occurred rarely** in the signal
  - For fNIRS, these are almost always **spike artifacts**



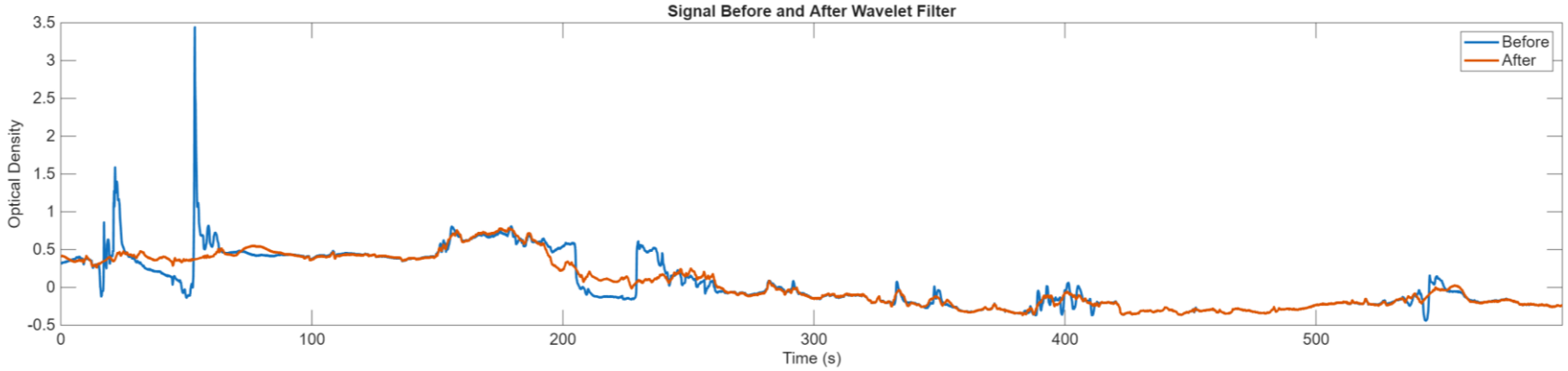
- **Method**

  Set a threshold that select these outliers (**suggested values 1.0 <= iqr <= 1.6)**

  Assign their coefficients to be zero

  Reconstruct the signal without those infrequent shapes

.

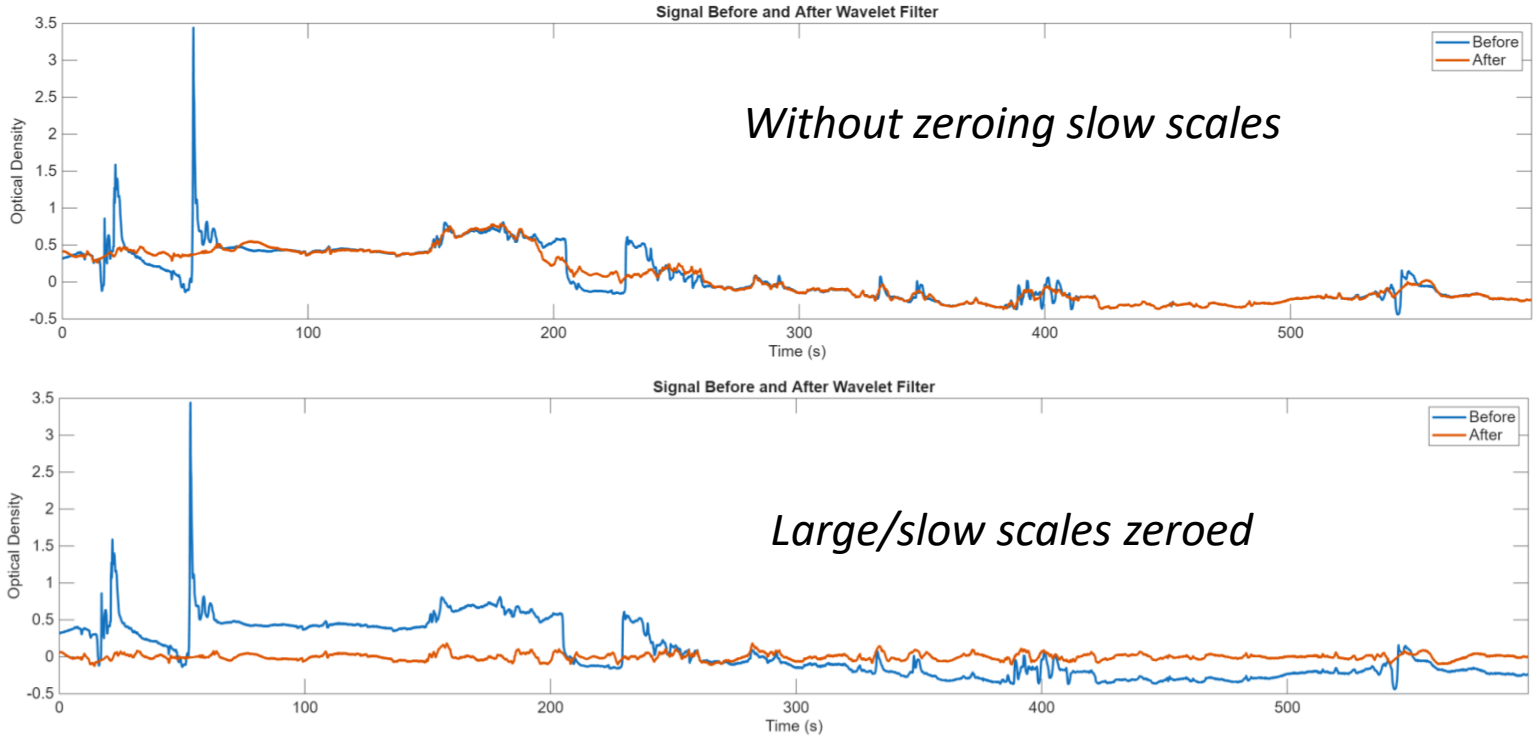


Removing artifacts: Wavelet filter

- Optionally, you can also **zero out the largest scales** to **reconstruct without the low frequencies**
  - May have unexpected consequences if the signal is short or the sampling rate is low
    - This causes the slowest scale will span into higher frequencies
  - Highly effective, but generally only suitable for moderately fast tasks

Enabled by default in NIRS Toolbox.

Disabled by default in our code.



Removing artifacts: Wavelet filter

Frequencies most affected by zeroing the slowest scale:

Example: 2 min @ 1 Hz

The slowest scale would predominantly span up to 0.0625 Hz

Resting state often uses a 0.009 – 0.08 Hz bandpass. This would require 14.5 min @ 9 Hz to avoid the band of interest.

Duration (min)	1 Hz	2 Hz	3 Hz	4 Hz	5 Hz	6 Hz	7 Hz	8 Hz	9 Hz	10 Hz
2	0.0625	0.0625	0.09375	0.0625	0.078125	0.09375	0.109375	0.0625	0.070313	0.078125
2.5	0.0625	0.0625	0.09375	0.0625	0.078125	0.09375	0.054688	0.0625	0.070313	0.078125
3	0.0625	0.0625	0.046875	0.0625	0.078125	0.046875	0.054688	0.0625	0.070313	0.078125
3.5	0.0625	0.0625	0.046875	0.0625	0.039063	0.046875	0.054688	0.0625	0.070313	0.039063
4	0.03125	0.03125	0.046875	0.03125	0.039063	0.046875	0.054688	0.03125	0.035156	0.039063
4.5	0.03125	0.03125	0.046875	0.03125	0.039063	0.046875	0.054688	0.03125	0.035156	0.039063
5	0.03125	0.03125	0.046875	0.03125	0.039063	0.046875	0.027344	0.03125	0.035156	0.039063
5.5	0.03125	0.03125	0.023438	0.03125	0.039063	0.023438	0.027344	0.03125	0.035156	0.039063
6	0.03125	0.03125	0.023438	0.03125	0.039063	0.023438	0.027344	0.03125	0.035156	0.039063
6.5	0.03125	0.03125	0.023438	0.03125	0.019531	0.023438	0.027344	0.03125	0.035156	0.019531
7	0.03125	0.03125	0.023438	0.03125	0.019531	0.023438	0.027344	0.03125	0.035156	0.019531
7.5	0.03125	0.03125	0.023438	0.03125	0.019531	0.023438	0.027344	0.03125	0.017578	0.019531
8	0.015625	0.015625	0.023438	0.015625	0.019531	0.023438	0.027344	0.015625	0.017578	0.019531
8.5	0.015625	0.015625	0.023438	0.015625	0.019531	0.023438	0.027344	0.015625	0.017578	0.019531
9	0.015625	0.015625	0.023438	0.015625	0.019531	0.023438	0.027344	0.015625	0.017578	0.019531
9.5	0.015625	0.015625	0.023438	0.015625	0.019531	0.023438	0.013672	0.015625	0.017578	0.019531
10	0.015625	0.015625	0.023438	0.015625	0.019531	0.023438	0.013672	0.015625	0.017578	0.019531
10.5	0.015625	0.015625	0.023438	0.015625	0.019531	0.023438	0.013672	0.015625	0.017578	0.019531
11	0.015625	0.015625	0.011719	0.015625	0.019531	0.011719	0.013672	0.015625	0.017578	0.019531
11.5	0.015625	0.015625	0.011719	0.015625	0.019531	0.011719	0.013672	0.015625	0.017578	0.019531
12	0.015625	0.015625	0.011719	0.015625	0.019531	0.011719	0.013672	0.015625	0.017578	0.019531
12.5	0.015625	0.015625	0.011719	0.015625	0.019531	0.011719	0.013672	0.015625	0.017578	0.019531
13	0.015625	0.015625	0.011719	0.015625	0.009766	0.011719	0.013672	0.015625	0.017578	0.009766
13.5	0.015625	0.015625	0.011719	0.015625	0.009766	0.011719	0.013672	0.015625	0.017578	0.009766
14	0.015625	0.015625	0.011719	0.015625	0.009766	0.011719	0.013672	0.015625	0.017578	0.009766
14.5	0.015625	0.015625	0.011719	0.015625	0.009766	0.011719	0.013672	0.015625	0.008789	0.009766
15	0.015625	0.015625	0.011719	0.015625	0.009766	0.011719	0.013672	0.015625	0.008789	0.009766

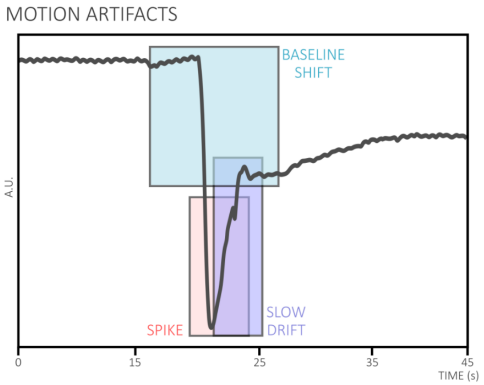
Removing artifacts: TDDR + Wavelet filter

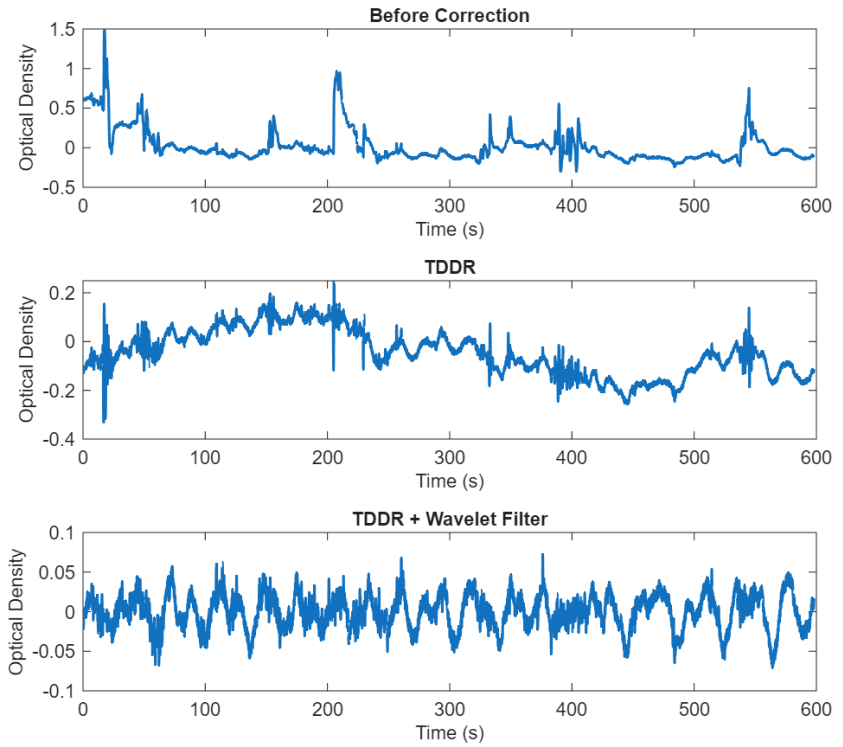
- **Summary**

  Together, TDDR + Wavelet corrects for all common transient artifacts

  Performing Wavelet after TDDR can also remove any low frequencies that were introduced

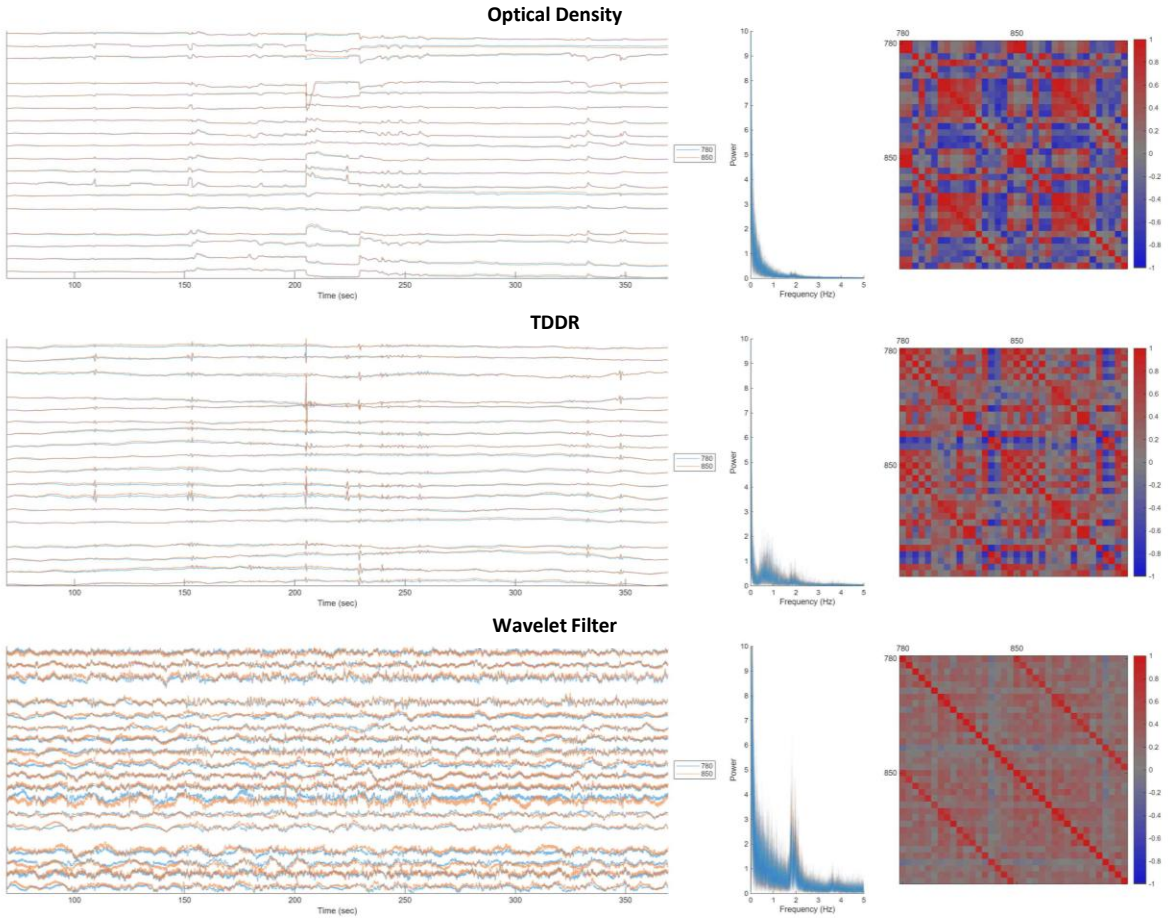
	TDDR	Wavelet Filter
Spikes	~	✅
Drift Artifacts	✅	❌
Baseline Shifts	✅	❌
Low Frequency Noise	❌	✅





Removing artifacts: Visual inspection

- **Timeseries**
  - No visible transient artifacts
  - Visible cardiac and respiration
- **Power spectrum**
  - Cardiac bump should become more pronounced
- **Correlation matrices**
  - Reduced overall magnitudes
  - Mostly positive
  - Wavelengths should be highly correlated within each channel



Break

Converting to concentrations

Modified Beer-Lambert Law (MBLL) converts optical densities into **relative changes in hemoglobin concentrations (HbO and HbR)**

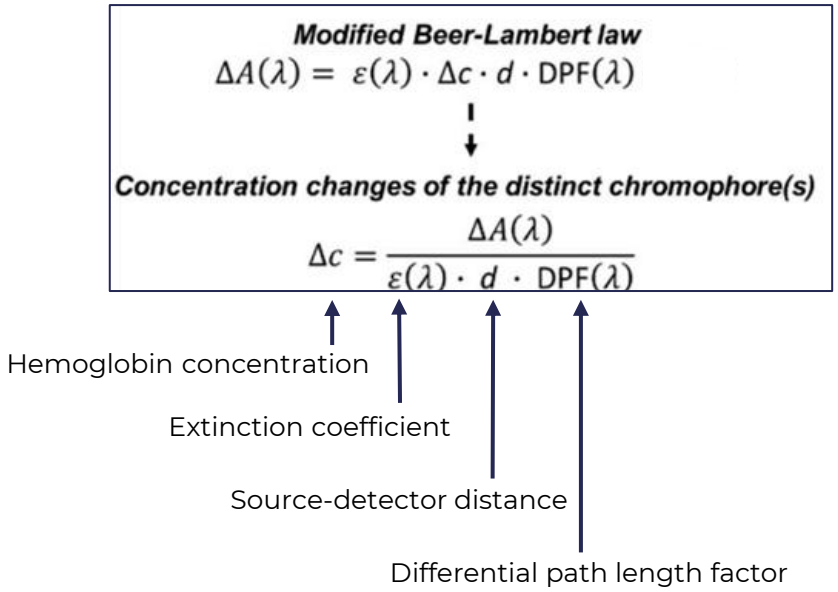
- *Note that continuous-wave fNIRS does not allow for absolute concentrations or changes*

- Differential path length factor (DPF) can be adjusted to **account for age**

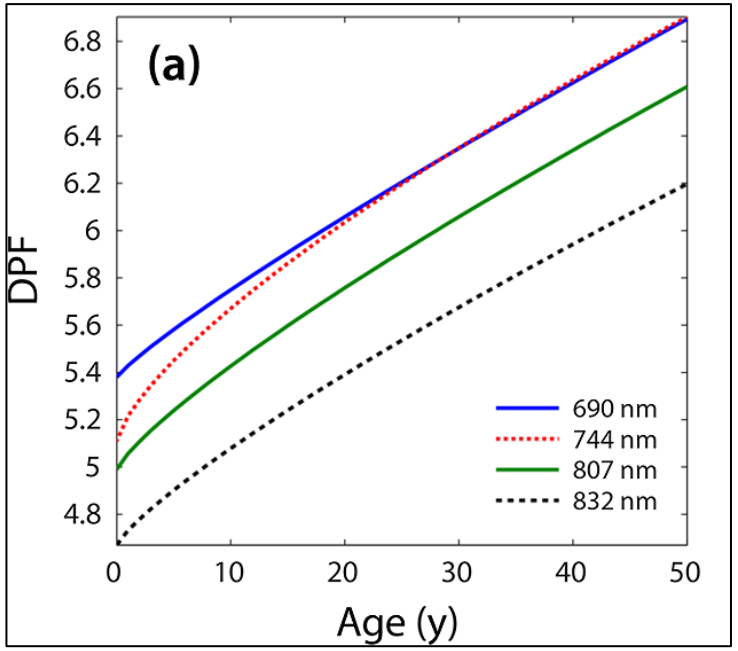
  Scholkmann and Wolf, 2013

- Accurate channel lengths are important for some types of analyses

  E.g., scales the Betas in GLMs



Converting to concentrations



Scholkmann and Wolf 2013

$$\mathrm{PPF}^{\lambda} = \mathrm{DPF}^{\lambda} \cdot \mathrm{PVF}^{\lambda}$$

$$\Delta\,\mathrm{OD}^{\lambda} \cong \left(\sum_i \epsilon_i^{\lambda} \cdot c_i\right) \cdot L \cdot \mathrm{PPF}^{\lambda}.$$

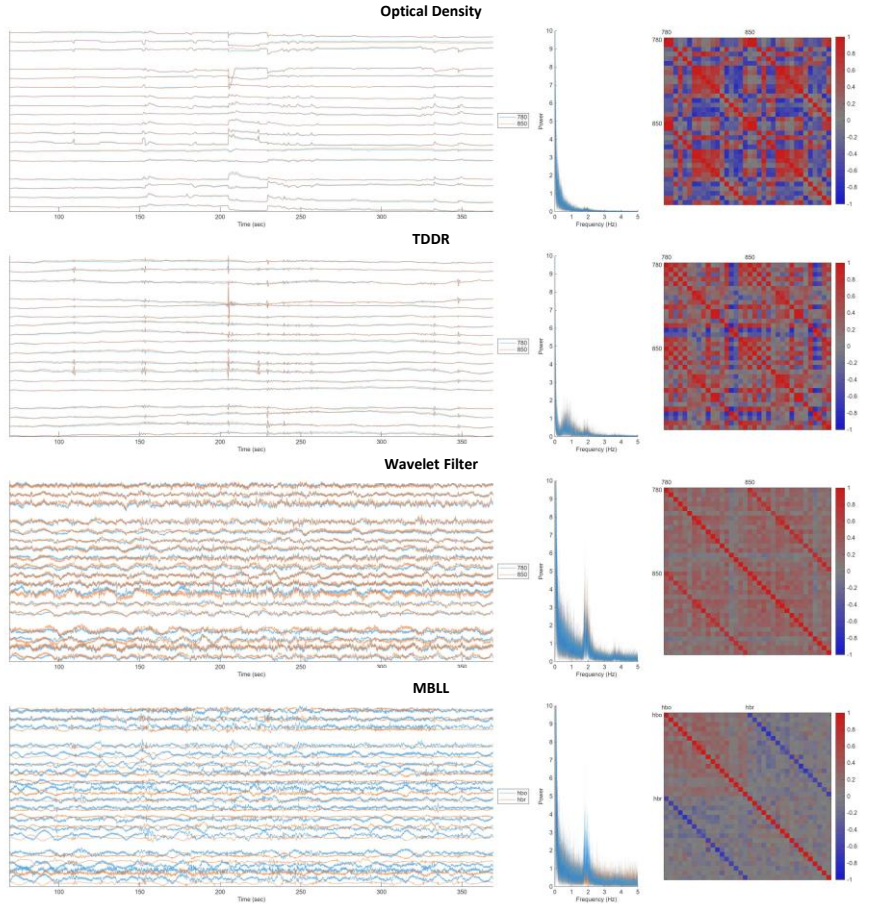
Whiteman et al., 2018

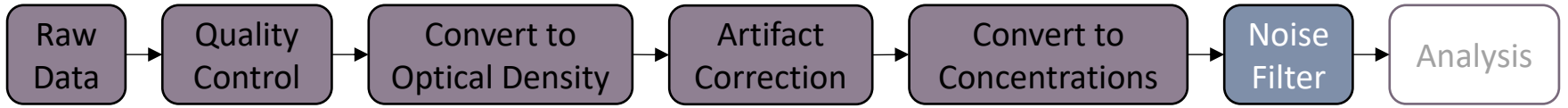
Converting to concentrations

- **<u>Visual inspection</u>**

- **Timeseries**
  - HbO/HbR signals should visibly anticorrelate in each channel

- **Correlation matrices**
  - HbO/HbR signals should anticorrelate in each channel
    - blue diagonal line in the top right and bottom left quadrants
  - Should still see widespread correlations due to shared physiology

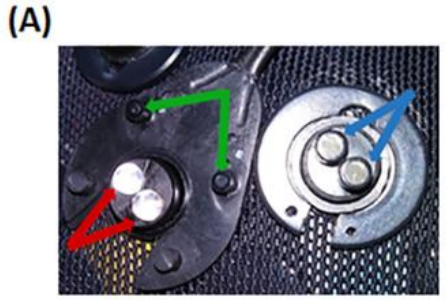


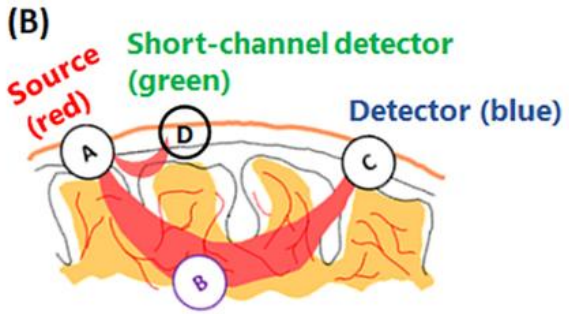


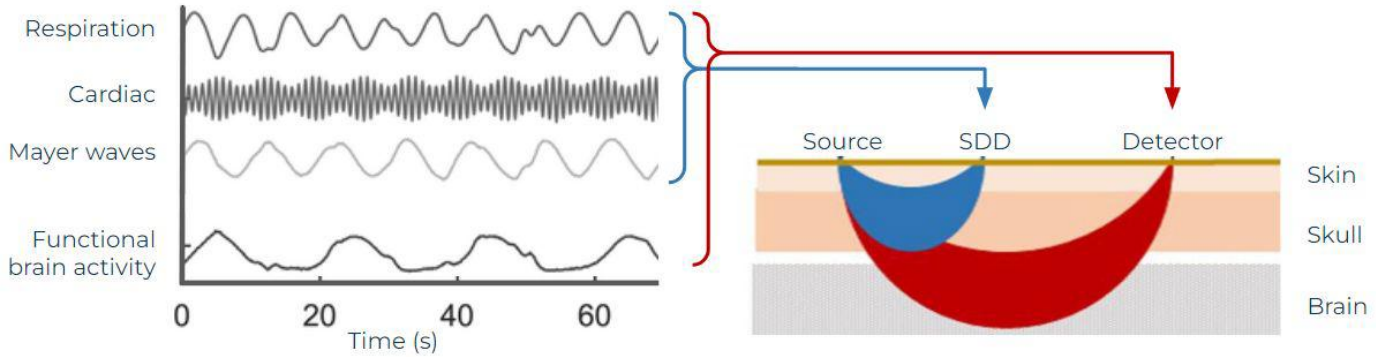
Removing physiological noise: Overview and considerations

- **Main options:**
  - Regress out measured physiology (for example, using external monitors or short channels)
    - Physiology may also be inferred from **short distance detectors**
  - Temporal filter to remove frequencies that you expect the physiology to occupy (e.g., bandpass)
  - Global signal regression or PCA analysis is an alternative if short channels are not available
    - Relies heavily on having many channels and good coverage

Removing physiological noise:
Bandpass filtering





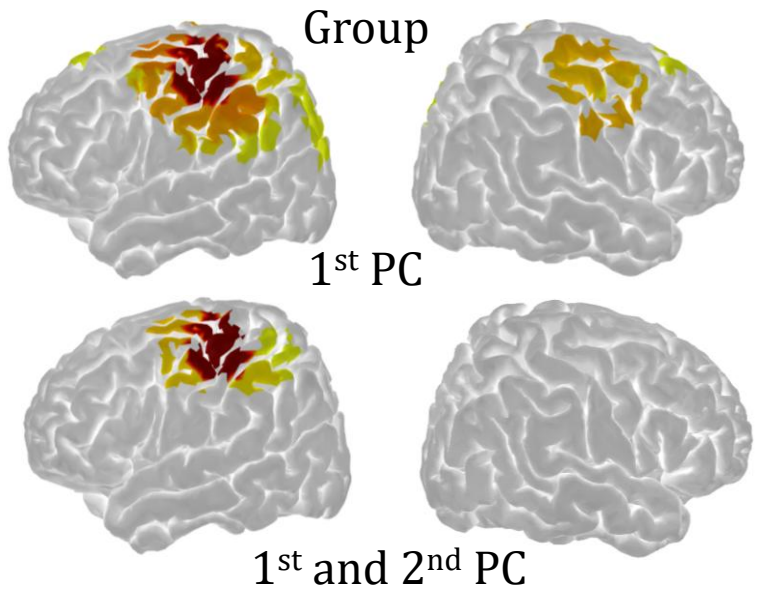


- Closest short channel?
- Most correlated short channel?
- All short channels?
- Components extracted from short channels?

Zhou et al., 2020

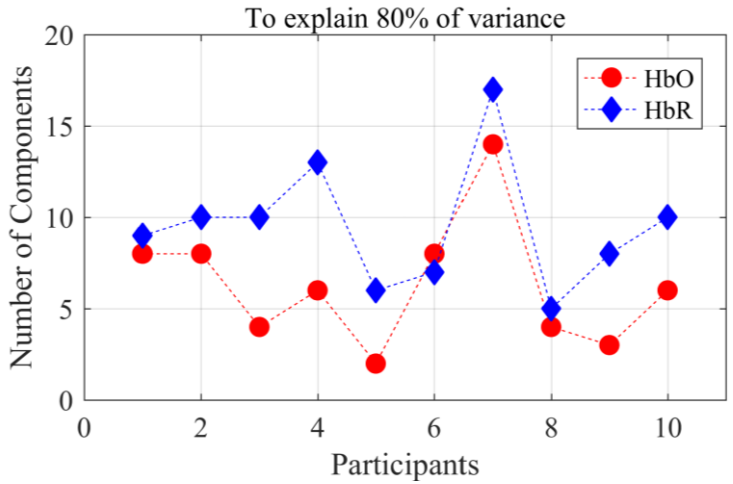
Santosa et al., 2020

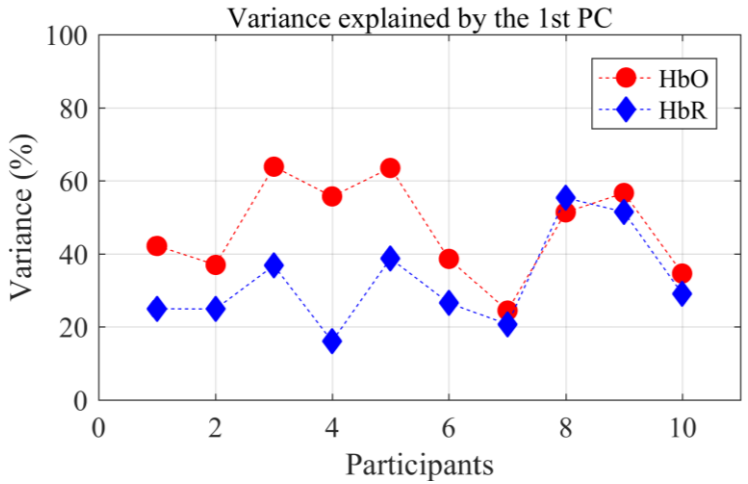
Removing physiological noise: Principal Component Analysis (PCA) as proxy of global systemic noise

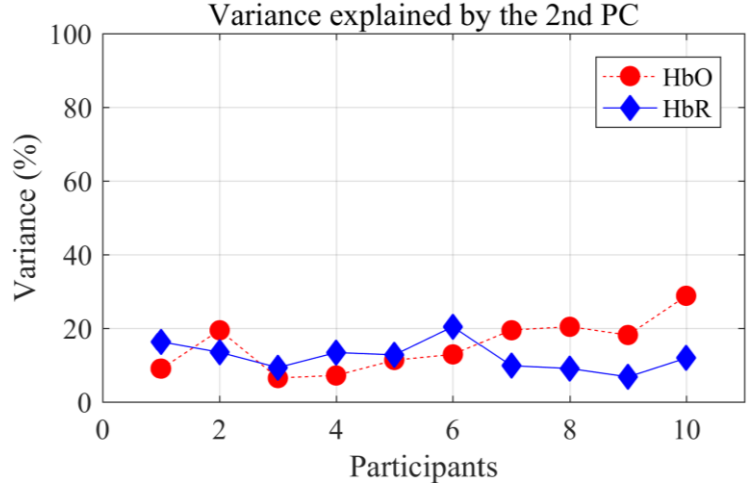


More details about PCA:

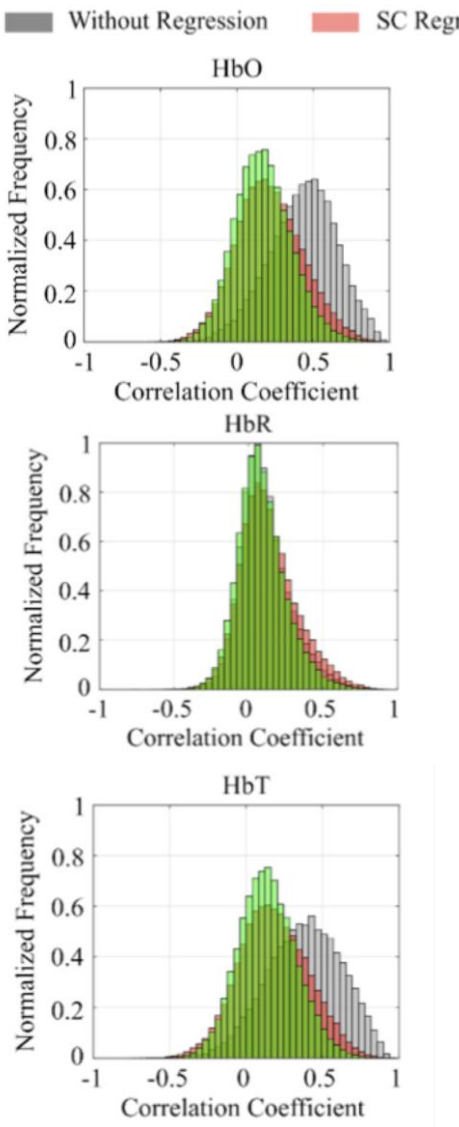
- PCA is applied separately for HbO and HbR;
- The user can define the **number of componets** directly or based on the **amount of variance** that the combined PCs represents.

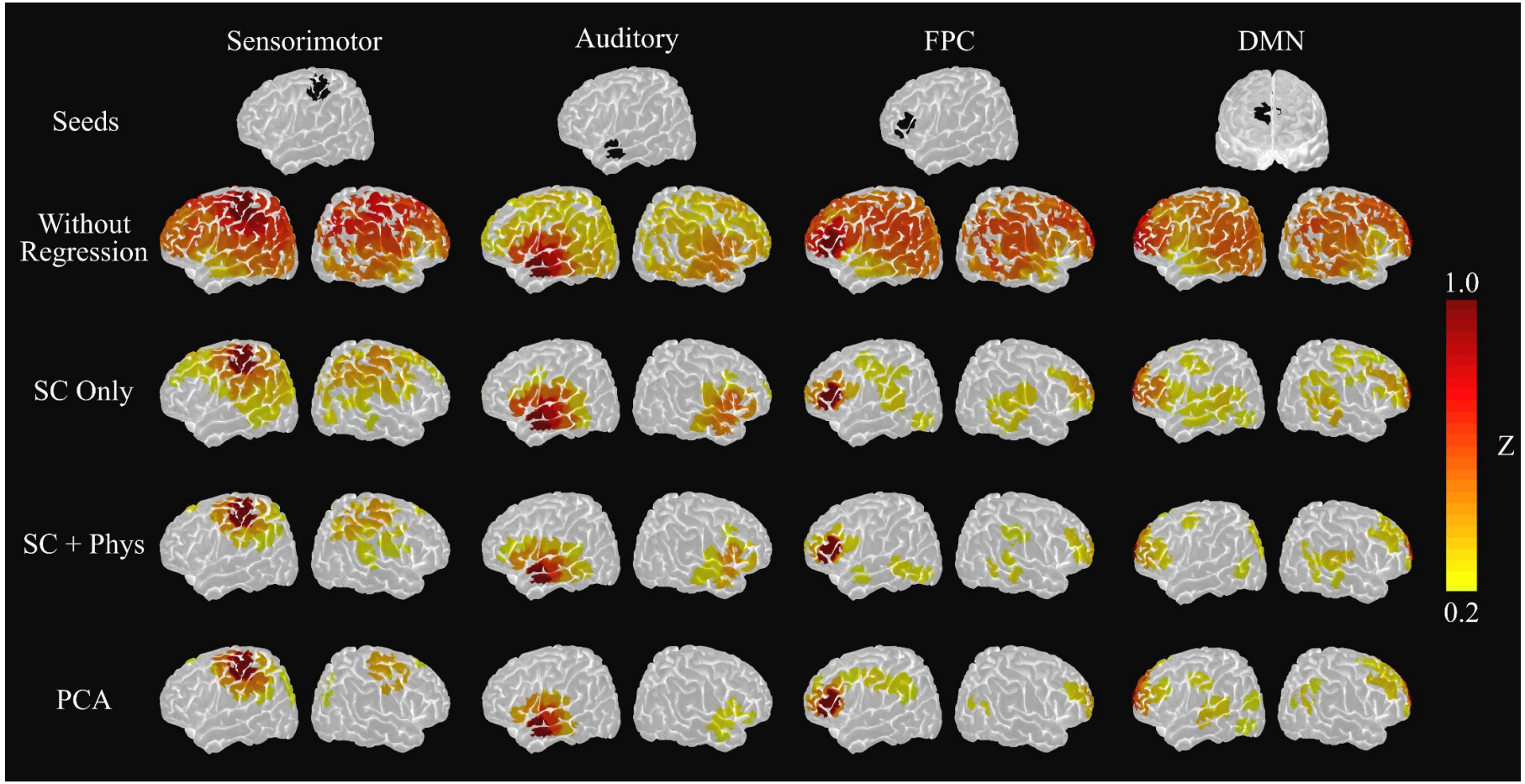






Preprocessing shapes the networks you recover

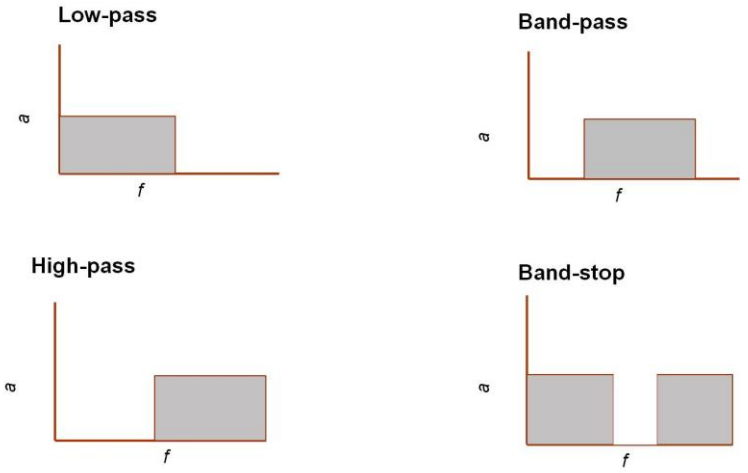


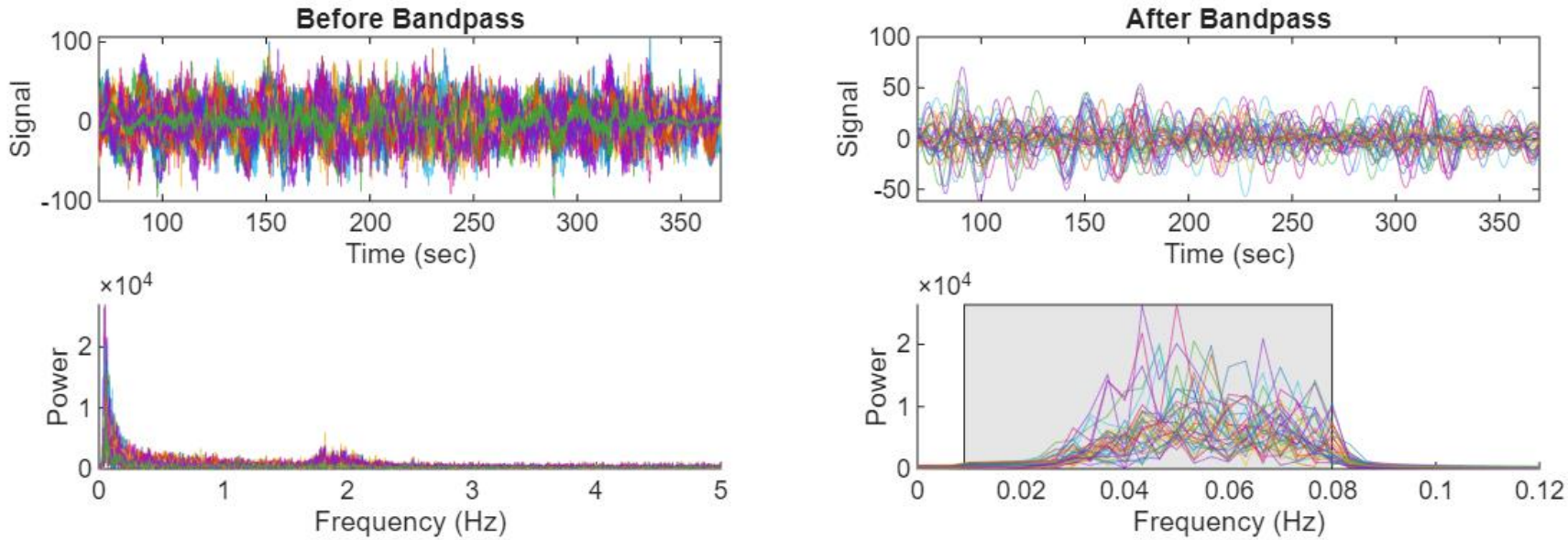


Abdalmalak, Novi et al., Front. Neurosci. (2022)

Removing physiological noise: Bandpass filtering

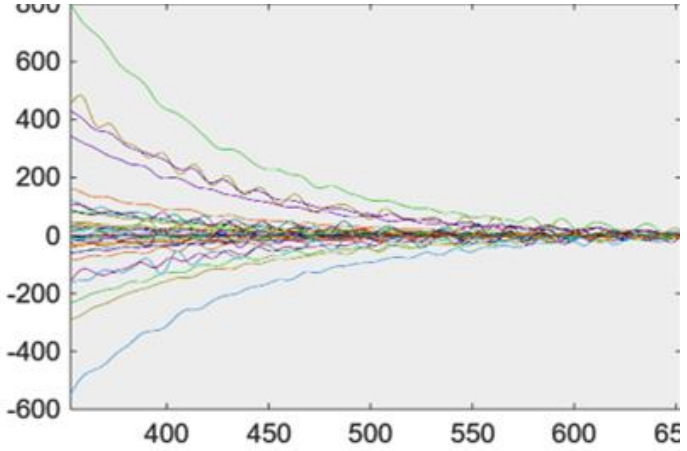
- Removes frequencies outside of a specified passband
  - Typically, ultra-slow trends (drift) and high frequency oscillations (cardiac)
- **For resting state data, the passband is generally 0.009 – 0.08 Hz**

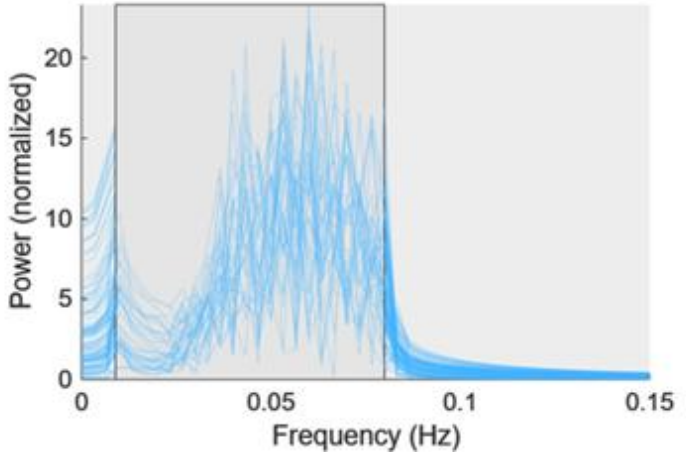


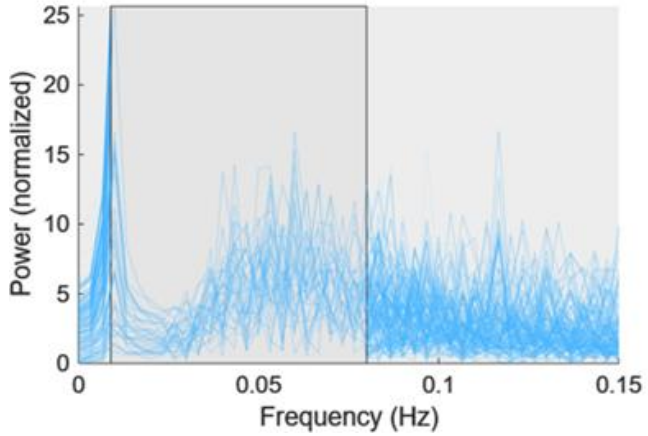


Removing physiological noise: Bandpass filtering

- **Artifacts to look out for**
- Spike at the upper bound
- Spike at the lower bound
- Failure to restrict spectra to the passband
- Large edge artifacts (esp. common in shorter samples)



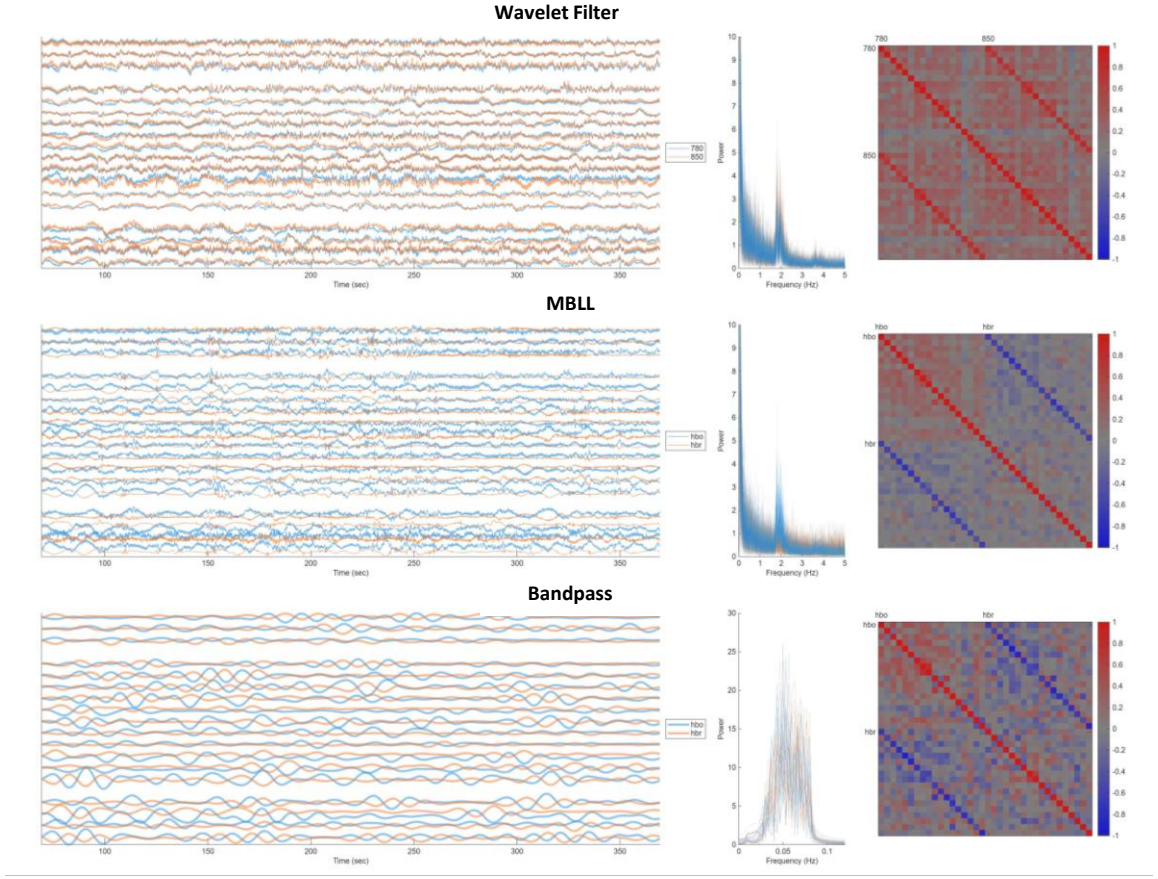




Removing physiological noise: Bandpass filtering

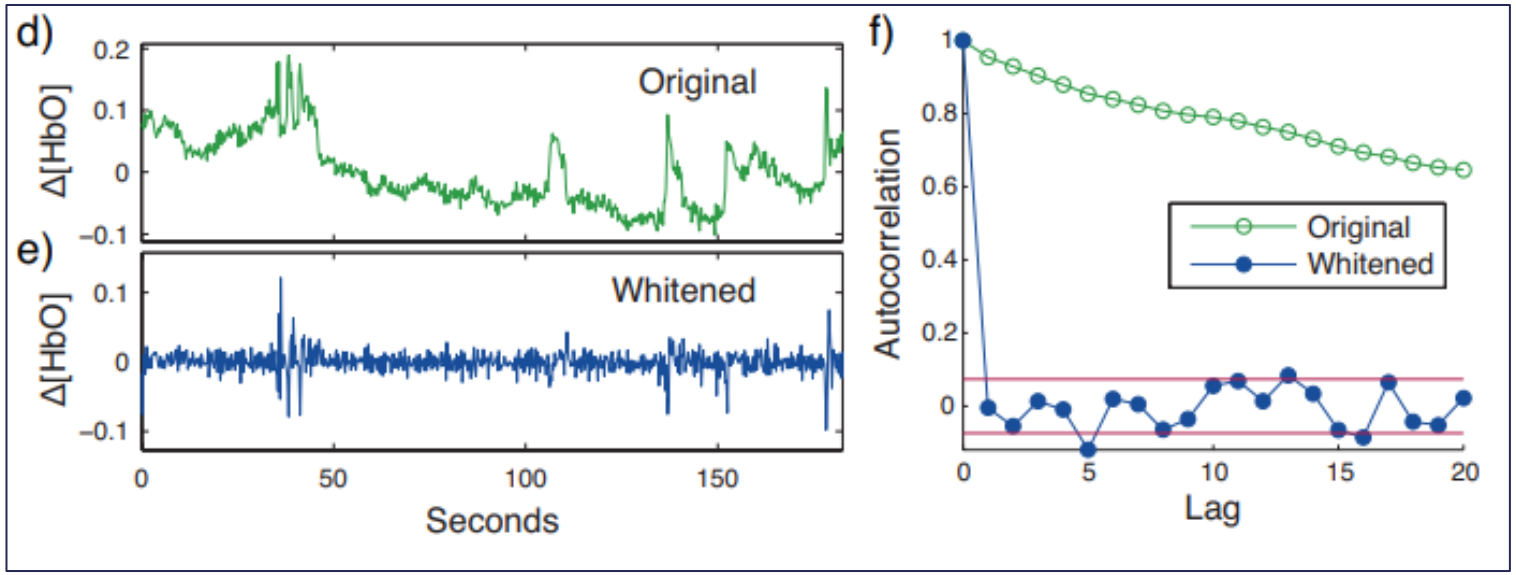
Visual inspection

- **Timeseries**
  - Smooth
  - No visible physiology
  - No edge artifacts
- **Power spectrum**
  - Limited to the passband frequencies
  - No bumps at the passband's edges
- **Correlation matrices**
  - New patterns should emerge in the correlation matrix, ideally with clusters of channels



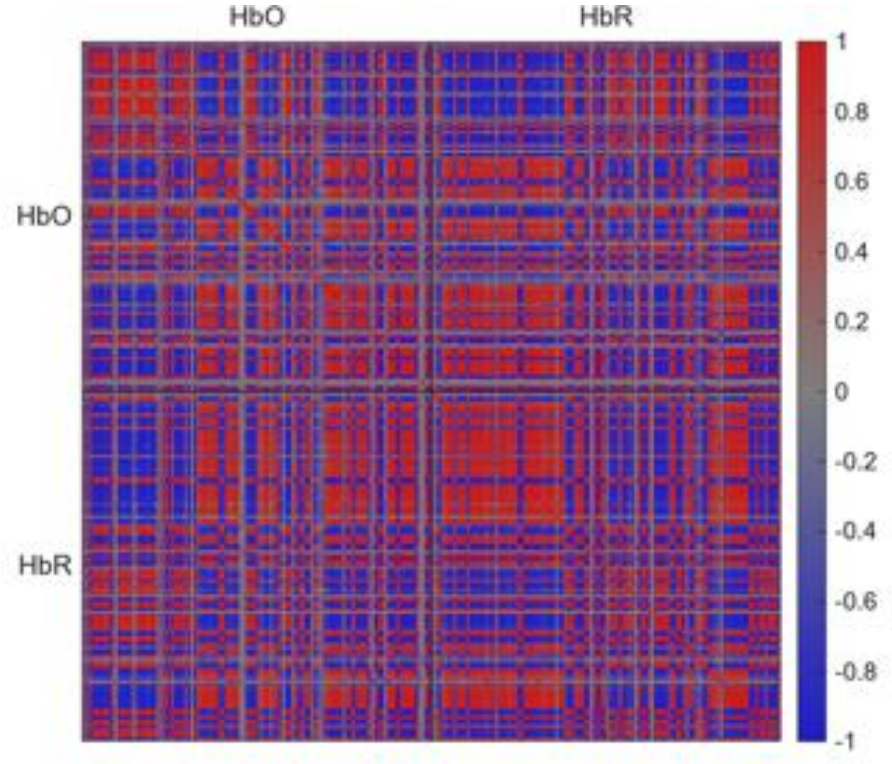
Removing noise: Prewhitening Filter

- Removes or reduces the temporal autocorrelation in the data
  - i.e., makes datapoints more independent
  - Critically important if analyzing with a GLM
- Interacts with bandpass: both utilize and subsequently reduce the high frequency components

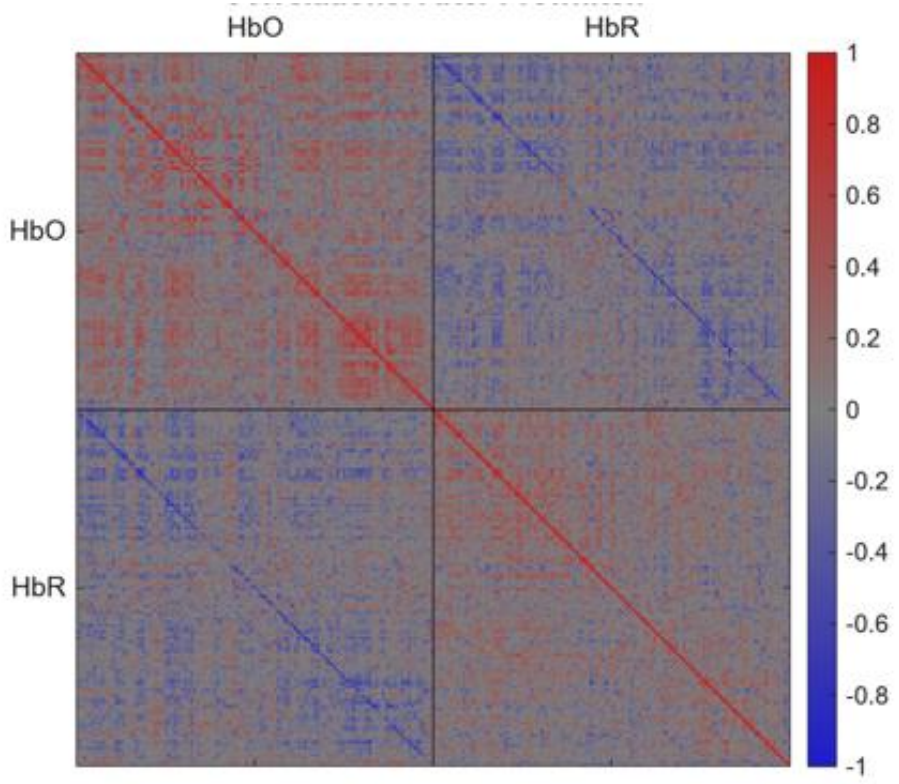


Removing noise: Prewhitening Filter

- **Warning**: most implementations induce a massive artifact in the first few seconds that must be trimmed out



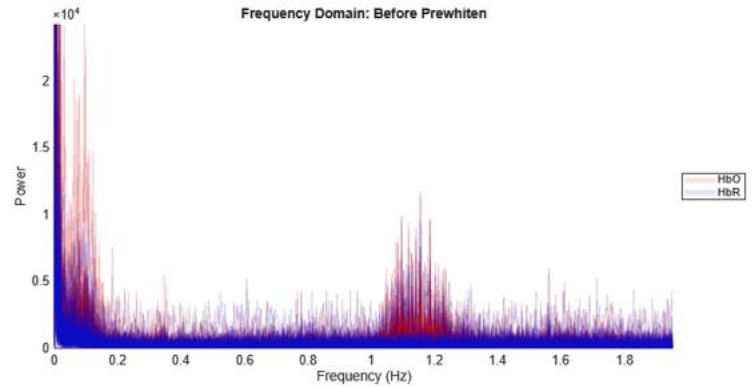
Without Trimming

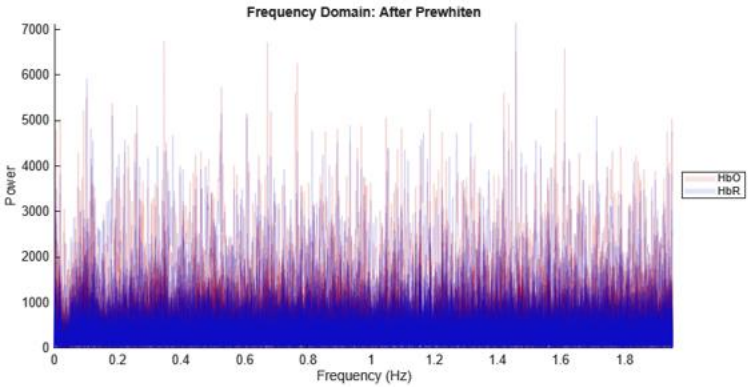


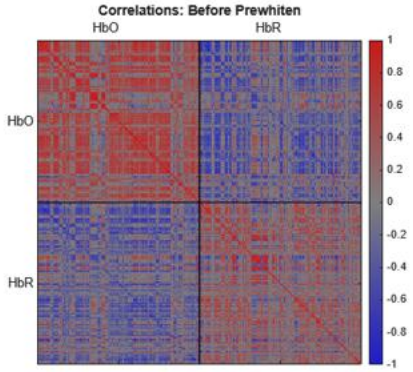
With Trimming

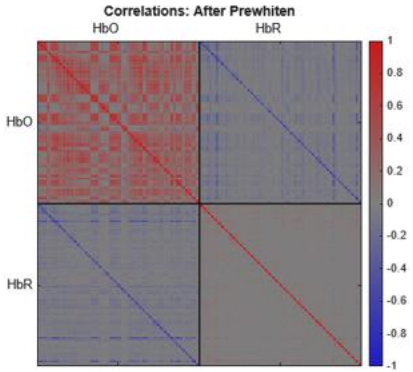
Removing noise: Prewhitening Filter

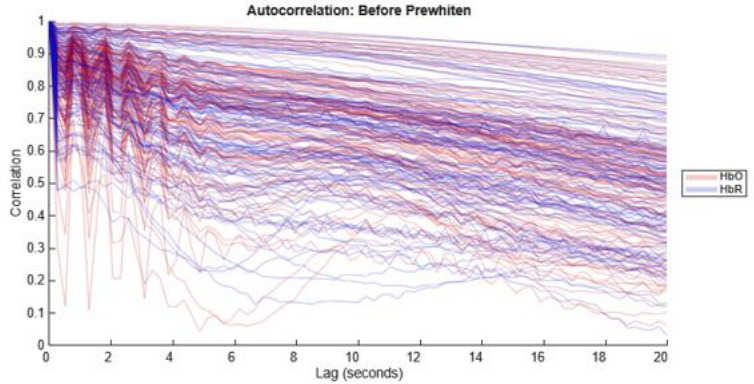
- Prewhitening before bandpass while high frequencies are still intact

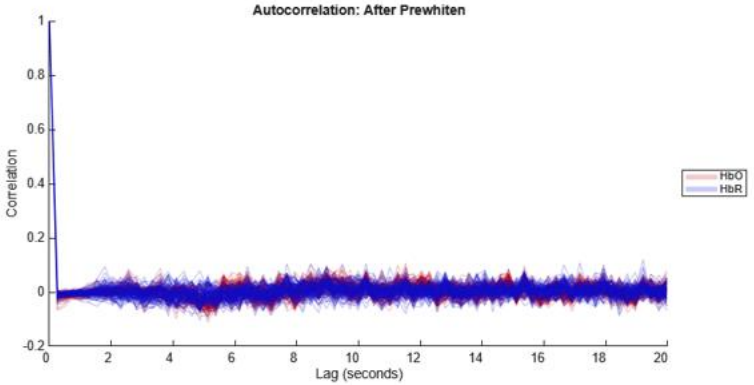






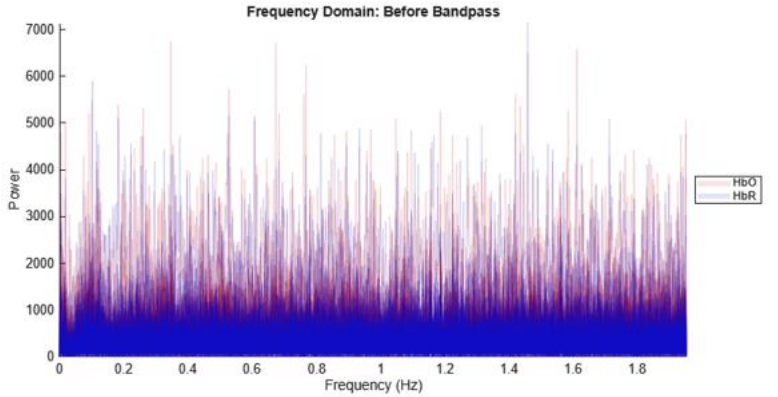


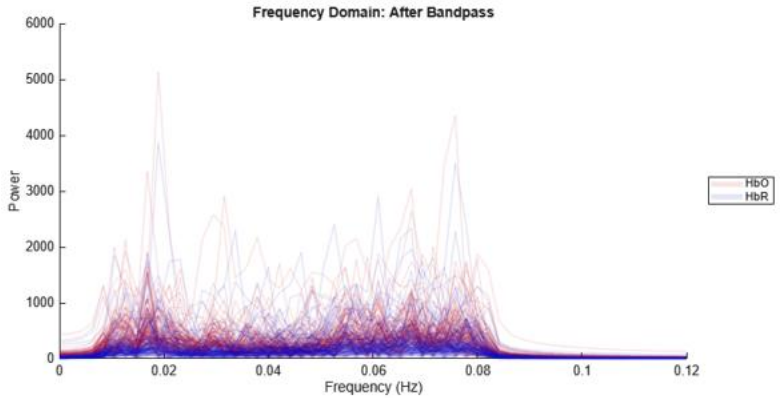


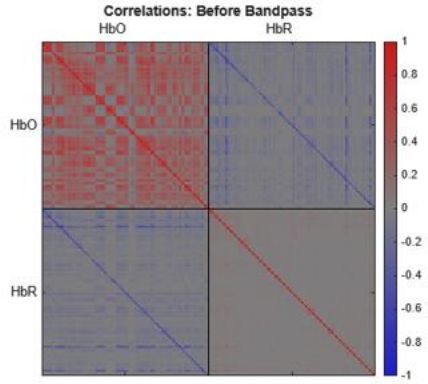


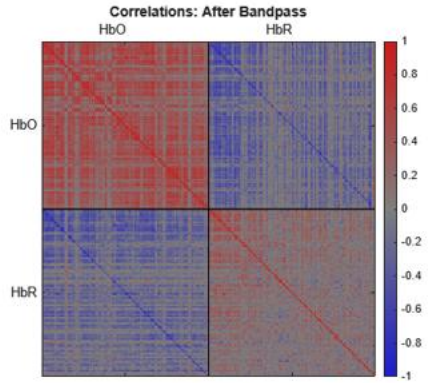
Removing noise: Prewhitening Filter

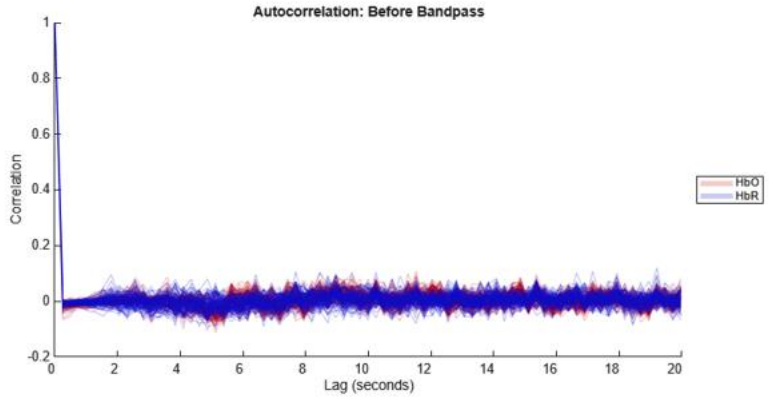
- However, following up with a bandpass induces autocorrelations

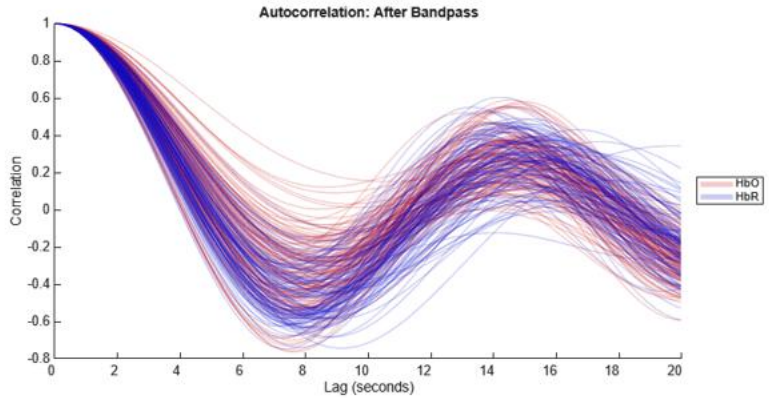








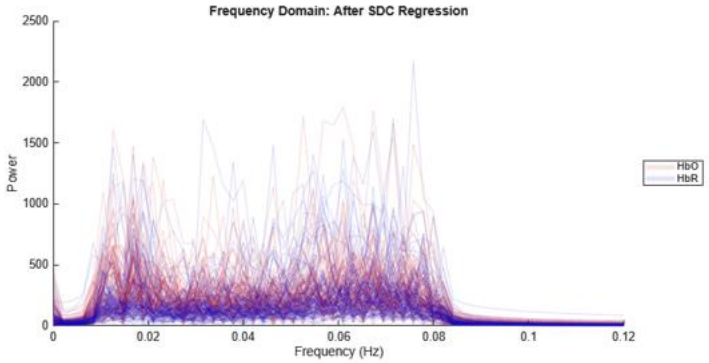


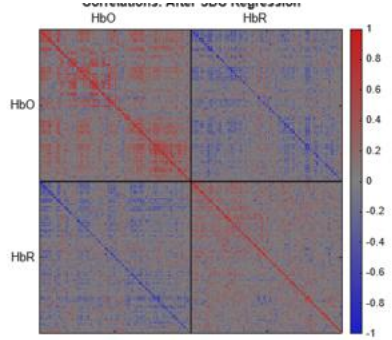


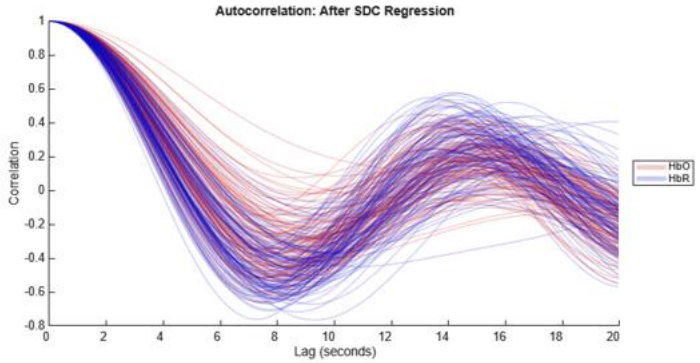
Removing noise: Prewhitening Filter

- The end results of Bandpass→Prewhiten and Prewhiten→Bandpass are mostly similar in the demo dataset, but can be more varied

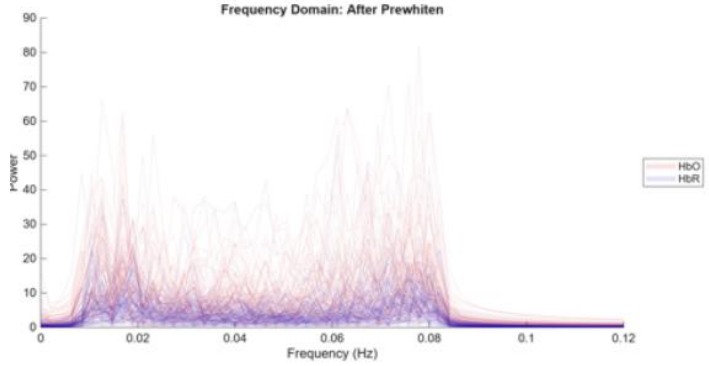
Prewhiten First

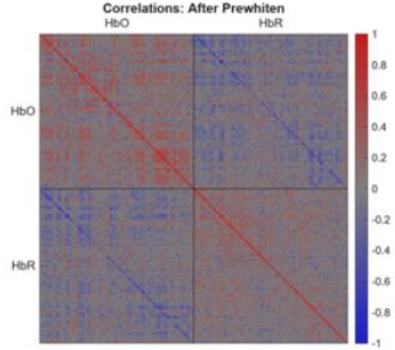


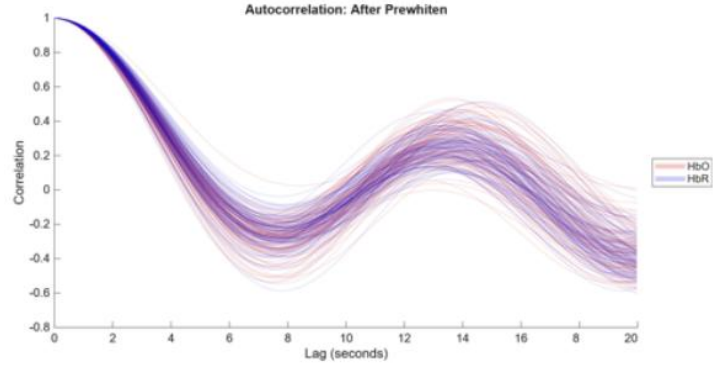




Bandpass First







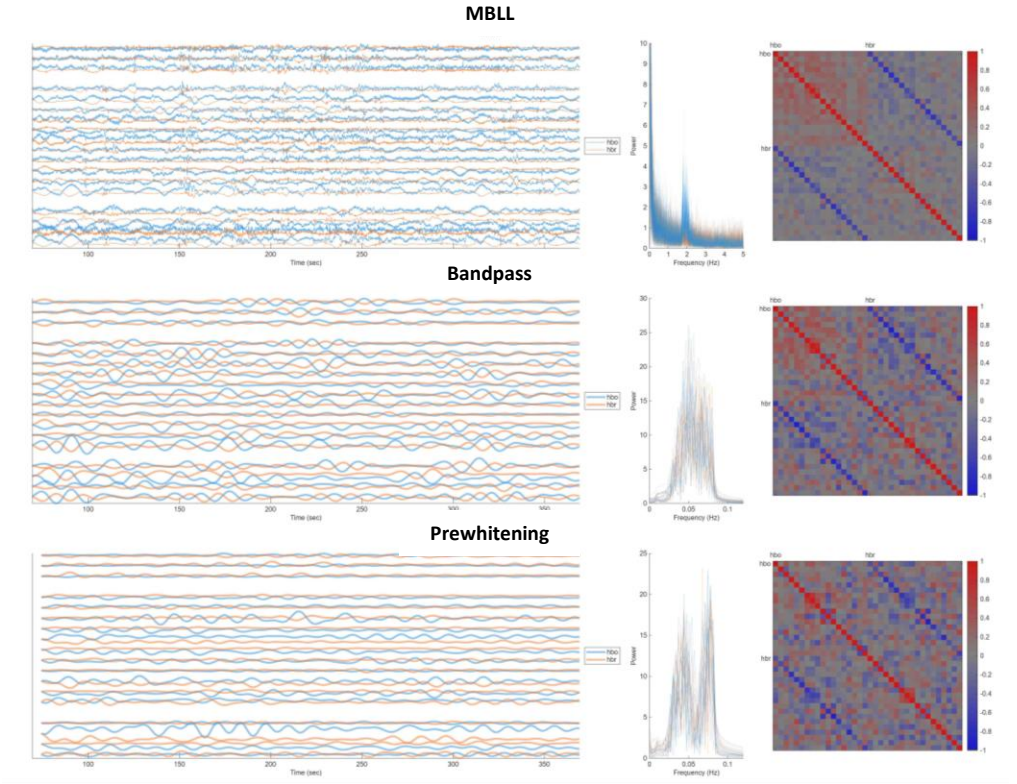
Removing physiological noise: Prewhitening

Visual inspection:

- Auto-correlation should be flattened
- **Timeseries**

  Artifacts that were shrunk down but not fully removed could be amplified and show up again
- **Correlation matrices**

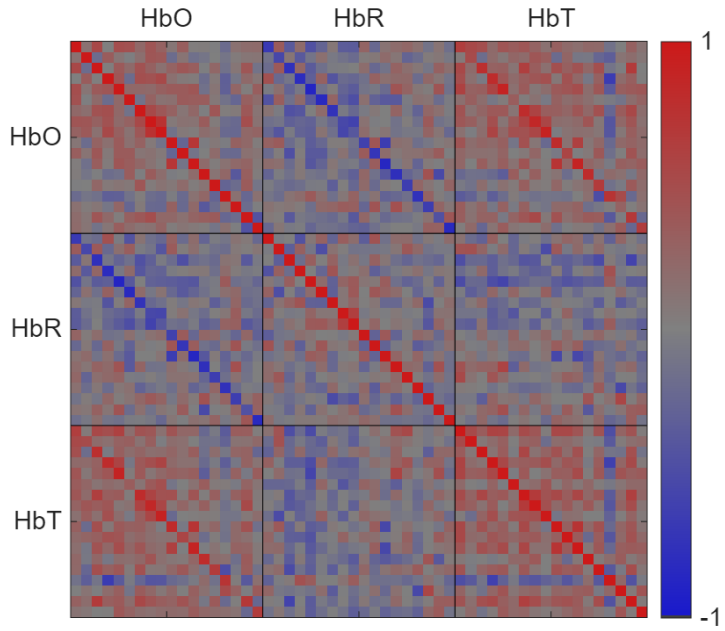
  Noticeably reduced magnitudes



Calculating total hemoglobin (HbT)

- Commonly used in resting state analyses (Novi et al., 2016, Abdalmalak et al., 2022)
- Largely **biased towards HbO** due to its larger amplitudes, but includes some of the nuisances that are unique to HbR
- Not mandatory, but quite prevalent in recent literature

$$HbT = HbO + HbR$$



What might you do differently for task analyses?

Import

- Define events

Quality Control

- Remove clean segment selection
- Could add trial exclusion (e.g., based on SCI/PSP from 2s before the trial until 16 after)
  - If using a GLM, excluded trials should be converted into individual regressors of no interest
- Assess trial-related motion / artifacts (e.g., ERA on accelerometer)

Preprocessing

- Consider using wavelet filter's slow scale zeroing as a highpass (if task is fast enough)
- Remove bandpass or greatly expand the passband (must not remove task frequencies or interfere with prehwitening)
- Include SDCs in GLM , don't regress them out in advance (if applicable)

OLS GLM requires more preprocessing than AR-IRLS, which can handle more noise

Hands-on

Let’s look at some scenarios

Scenario 1: No short channel regression

Scenario 2: No Motion artifact correction

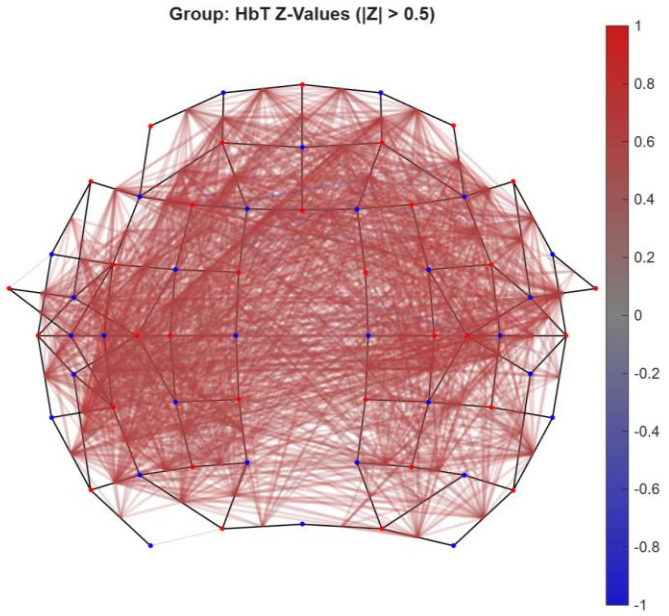
Scenario 3: Shorter segment (2 minutes)

Scenario 4: Wider band-pass filtering (0.01 – 0.15 Hz)

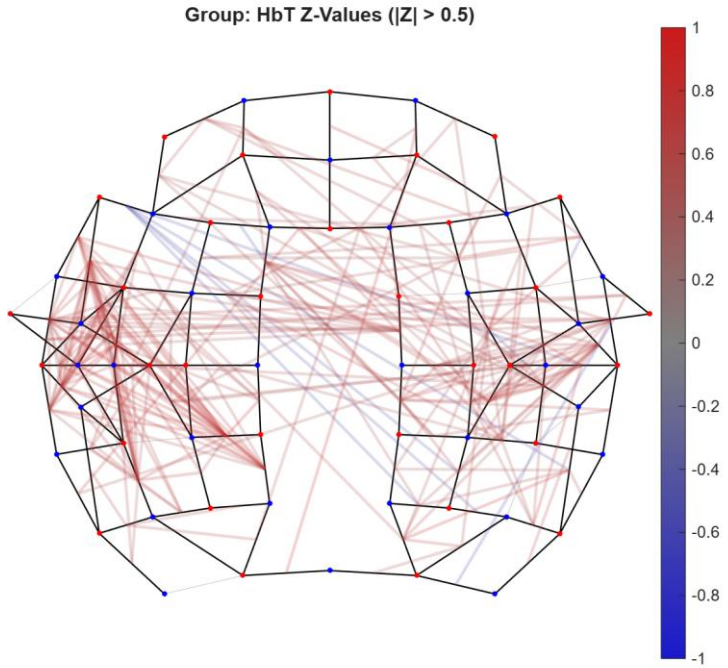
Scenario 5: No prewhitenning

Hands-on

Scenario 1: No short channel regression

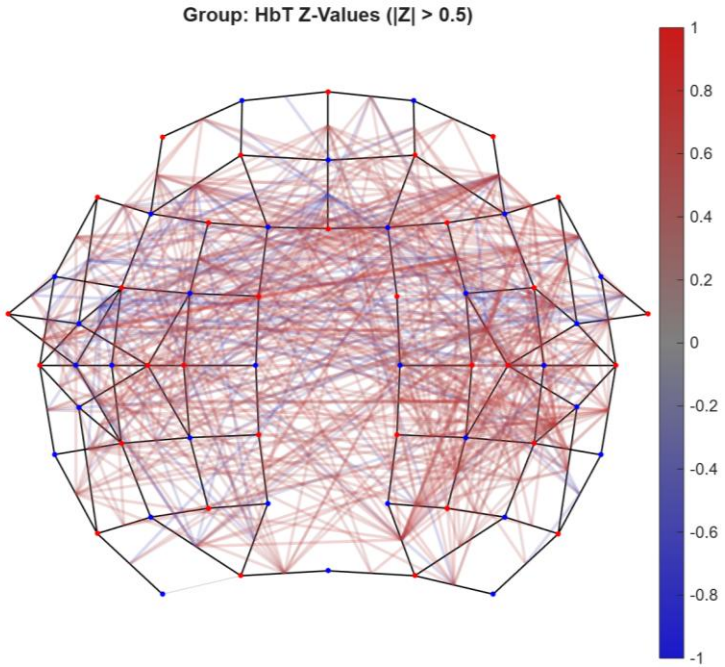


Scenario 2: No Motion artifact correction

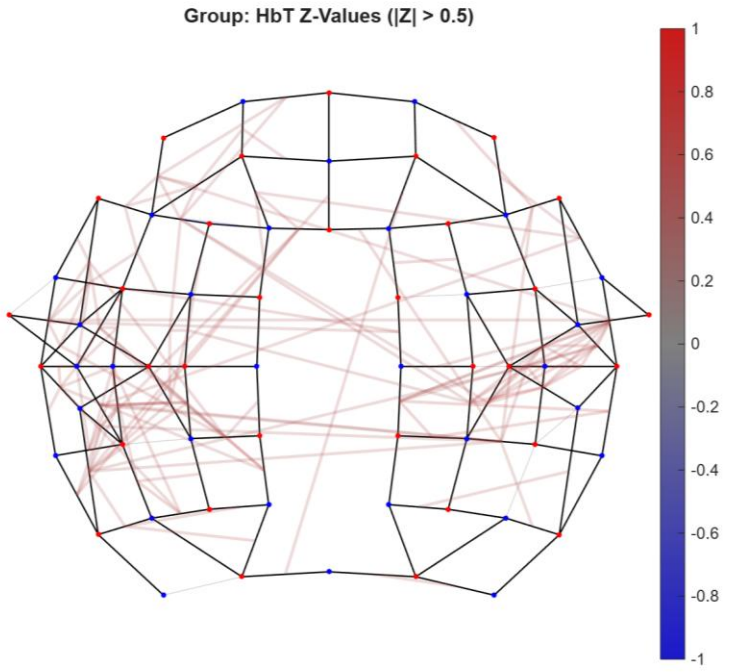


Hands-on

- Scenario 3: Shorter segment (2 minutes)

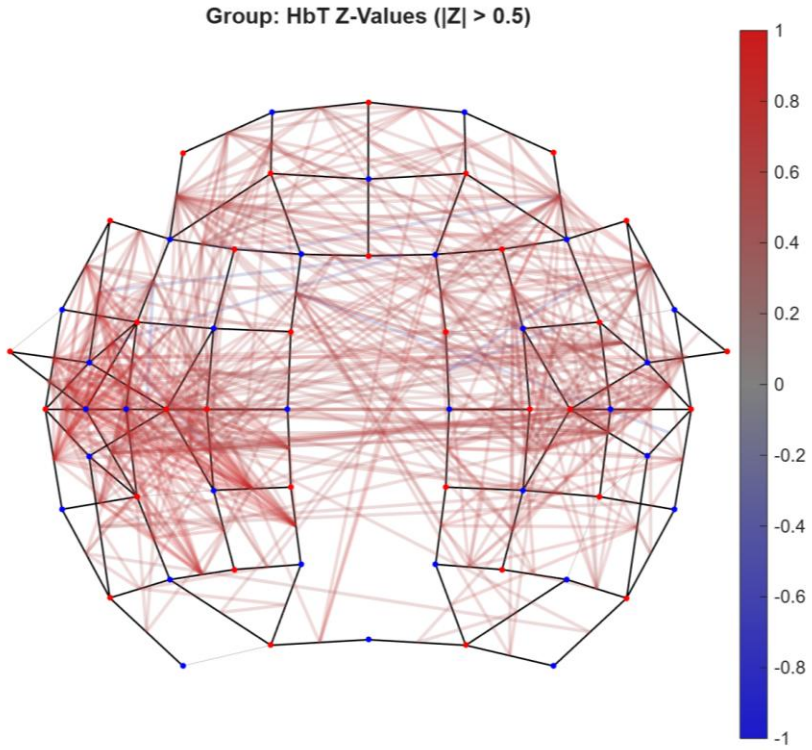


Scenario 4: Wider band-pass filtering



Hands-on

- Scenario 5: No prewhitenning



Thank you for your attention!

Any questions?

THE CHANNELS ARE OPEN.

