Wavelets and wavelet convolution
and brain music

Dr. Frederike Petzschner
Translational Neuromodeling Unit
06.03.2015
Why are we doing this?

We know that EEG data contain oscillations. Our goal is to disentangle these oscillations (alpha, gamma, theta waves)
What are oscillations?
Simply think of music...

Notes:

- Amplitude
  - Quieter
  - Louder

- Frequency
  - Lower pitch
  - Higher pitch

Music: Superposition of notes/waves:

- Wave 1:
- Wave 2:
- Superposition:
Decoding ‘brain music’

We want to decode brain music.
Knowing which frequency was present, when and how strong over time!

So at EACH POINT IN TIME we want to know the frequency, phase and amplitude of the underlying signal.
What have we done so far?

We model our signal as a linear combination of sine waves
→ Fourier Transform: frequency domain representation

What's the problem with that?

EEG data

No temporal weighting (Fourier transform)

Frequency changes over time
What would we want?

Something that includes a temporal weighting

- EEG data
- No temporal weighting (Fourier transform)
- Strong temporal weighting
What would we want?

Something that includes a temporal weighting

- EEG data
- No temporal weighting (Fourier transform)
- Strong temporal weighting
- Boxcar temporal weighting
What would we want?

Something that includes a temporal weighting

- EEG data
- No temporal weighting (Fourier transform)
- Strong temporal weighting
- Boxcar temporal weighting
- Gaussian temporal weighting
- Morlet wavelet
What is this talk about?

Frequency information at each point in time is a weighted sum of the frequency information of the instantaneous time AND the neighboring time.

Time-frequency representations retain advantages of both time and frequency domain
While making only small sacrifices to precision:

Frequency information at each point in time is a weighted sum of the frequency information of the instantaneous time AND the neighboring time.
How to make a Morlet wavelet

Instead of using many sine waves with different frequencies, time-frequency decomposition uses many wavelets with different frequencies.
srate = 500; % sampling rate in Hz
f = 10; % frequency of the sine wave in Hz
time = -1:1/srate:1; % time, from -1 to 1 second in steps of 1/sampling-rate

sine_wave = exp(2*pi*1i*f.*time); % complex wavelet

% make a Gaussian
s=6/(2*pi*f);
gaussian_win = exp(-time.^2./(2*s^2));

% and together they make a wavelet!
wavelet = sine_wave .* gaussian_win;
Wavelet Convolution as a Bandpass Filter

Remember a convolution is the time-varying mapping between a kernel (here the wavelet) and a signal (EEG).
Limitations of the approach so far

- It works like a bandpass filter nice, but...
- time-frequency analysis means we want the power and phase which are not directly apparent
- convolution with a Morlet wavelet depends on the phase offset between wavelet and data
Integrals are “multiplication, taking changes into account” and the dot product is “multiplication, taking direction into account”.

\[ \vec{a} \cdot \vec{b} = a_x \cdot b_x + a_y \cdot b_y = |\vec{a}| |\vec{b}| \cos(\theta) \]
The solution

Use complex Morlet wavelets
Reminder: What do we want?

Extract estimates of time-varying frequency band-specific power and phase from EEG!

What will we do to get there?
Calculate complex Morlet Wavelets
(3D: time, real and imaginary part)
Recap complex numbers

<table>
<thead>
<tr>
<th>Fun Fact</th>
<th>Negative Numbers (-x)</th>
<th>Complex Numbers (a +bi)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invented to answer</td>
<td>“What is 3 – 4?”</td>
<td>“What is sqrt(-1)?”</td>
</tr>
<tr>
<td>Strange because...</td>
<td>How can you have less than nothing?</td>
<td>How can you take the square root of less than nothing?</td>
</tr>
<tr>
<td>Intuitive meaning</td>
<td>“Opposite”</td>
<td>“Rotation”</td>
</tr>
<tr>
<td>Considered absurd until</td>
<td>1700s</td>
<td>Today 😊</td>
</tr>
<tr>
<td>Multiplication cycle</td>
<td>1, -1, 1, -1...</td>
<td>1, i, -1, -i...</td>
</tr>
<tr>
<td>[&amp; general pattern]</td>
<td>x, -x, x, -x...</td>
<td>x, y, -x, -y...</td>
</tr>
<tr>
<td>Use in coordinates</td>
<td>Go backwards from origin</td>
<td>Rotate around origin</td>
</tr>
<tr>
<td>Measure size with</td>
<td>Absolute value</td>
<td>Pythagorean Theorem</td>
</tr>
<tr>
<td>[ ]</td>
<td>$\sqrt{(-x)^2}$</td>
<td>$\sqrt{a^2 + b^2}$</td>
</tr>
</tbody>
</table>

http://betterexplained.com/articles/a-visual-intuitive-guide-to-imaginary-numbers/
Recap complex numbers

\[ i^2 = -1 \]

Rotate 1 to -1

http://betterexplained.com/articles/a-visual-intuitive-guide-to-imaginary-numbers/
Recap complex numbers

http://betterexplained.com/articles/a-visual-intuitive-guide-to-imaginary-numbers/
Recap complex numbers

4-8i

\[ \text{real} = M \cos(\theta) \]
\[ \text{imag} = M \sin(\theta) \]
\[ \text{real} + \text{imag} = M \cos(\theta) + M \sin(\theta) \]
\[ \text{real} + \text{imag} = M \left[ \cos(\theta) + \sin(\theta) \right] \]
\[ a + ib = M \left[ \cos(\theta) + i \sin(\theta) \right] \]

\[ M = \sqrt{\left(\text{real}^2 + \text{imag}^2\right)} \]
\[ \theta = \arctan\left(\frac{\text{imag}}{\text{real}}\right) \]
Complex Morelet

- Remember a Morlet is created by multiplying a sine wave with a Gaussian.

- A complex Morlet is created by multiplying a complex sine wave with a Gaussian.

\[
cmw = Ae^{-t^2/2s^2}e^{j2\pi ft}
\]

\[
A = \frac{1}{(s\sqrt{\pi})^{1/2}}
\]
Why ???

\[ real = M \cos(\theta) \]
\[ \text{imag} = M \sin(\theta) \]
\[ \sin \theta = \cos \left( \frac{\pi}{2} - \theta \right) \]

Rotate 1 to -1
% parameters...
srate = 500; % sampling rate in Hz
f = 10; % frequency of wavelet in Hz
time = -1:1/srate:1; % time, from -1 to 1 second in steps of 1/sampling-rate
s = 6/(2*pi*f);

% and together they make a wavelet
wavelet = exp(2*pi*1i*f.*time) .* exp(-time.^2./(2*s^2));

Wavelet=
...
-0.0002 - 0.0001i
-0.0002 - 0.0001i
-0.0003 - 0.0001i
-0.0003 - 0.0001i
-0.0004 - 0.0001i
-0.0004 - 0.0000i
-0.0005 + 0.0000i
...

Projection onto real and time axes

Projection onto imaginary and time axes
Complex Wavelets
Walking in imaginary space...
Why do we need that?

Remember our problem:

A) A wavelet at 10 Hz

B) One-cycle sine wave at 10 Hz

C) Result of convolution

D) Dot product depends on relative phase
   - dot product > 0
   - dot product < 0
   - dot product = 0
Why do we need that?

Remember our problem:

A) Signal and wavelet

B) Dot product in polar space

Length provides information about similarity of the one cycle sine and Morlet wavelet

Orientation provides information about the phase

We get the bandpass filter BUT in addition we also get information about the phase AND the amplitude!!!!
We get the bandpass filter BUT in addition we also get information about the phase AND the amplitude!!!
That was one step of the convolution…

... now we need to get from one point to a time series of power and phase values for ONE frequency band.
Now look at that in 3D again...
Concrete considerations: Question answer session

- Lowest frequency?
  
  *Hypothesis driven: e.g. looking at alpha 5-6 Hz*

- Highest frequency?
  
  *Hypothesis and sampling rate driven: you can’t use frequencies higher than the Nyquist frequency (sampling rate 500Hz, max 250 Hz better would be 125 Hz) → If no expectations: 4-60Hz*

- How many frequencies?
  
  20-30 for 4-60Hz

- Linear or logarithmic spacing of frequencies?
  
  *both correct. As frequencies are often conceptualized on log space log spacing makes sense as you get equal distance data (especially if you are interested in lower-frequencies)*

- How long should wavelets be?
  
  Long enough so that the lowest-frequency wavelet tapers to zero
Concrete considerations: Question answer session

• How many cycles should be used for the Gaussian Taper

defines the width of the wavelet, non-trivial parameter, will influence the results ➔ trade of between temporal and frequency precision.

if you are looking for transient changes ➔ smaller number of cycles
if you are looking for frequency-band activity over an extended period of time (e.g. visual stimulation, working memory) ➔ larger number of cycles
Take home:
- We want to decode brain music
- Because the main frequency components of that music may change over time we want something that takes both the time and frequency domain into account
- What we can use is a Morlet Wavelet: combination of a Gaussian with a Sine
- Allows us to get a time-frequency representations of our data that retain advantages of both domains
- To get at the actual phase and power information we need to use a complex Morlet Wavelet

Recommended Reading: chapter 12-13 Cohen 2014
Concrete considerations: Question answer session

- How strong is the frequency smoothing (incorporation of neighboring frequencies)?

  *reported in terms of full-width at half-maximum (FWHM) = frequency at which power is at 50% on the left and right sides of the peak*
Wavelet families

Group of wavelets that share the same properties but have different frequencies.

How to construct a wavelet family:

1. Don’t use frequencies lower than your epoch (1s data no less than 1 Hz → better 4Hz or faster)
2. Don’t choose frequencies above Nyquist frequency (one-half of the sampling rate)
3. Not much gain from 0.1 Hz increase → 15-30 frequencies between 3 Hz- 60 Hz should be enough