JPND research
EU Joint Programme – Neurodegenerative Disease Research

Amsterdam team: prof. dr. Jaap van Dieën, dr. Sjoerd Bruijn, dr. Bernadette van Wijk
dr. Moira van Leeuwen

# Foot Placement Control & Corticomuscular Coherence

## Concepts &

## Analysis Pipeline

Amsterdam Movement Sciences
VU VRIJE UNIVERSITEIT AMSTERDAM
Amsterdam UMC University Medical Centers

JPND research
EU Joint Programme – Neurodegenerative Disease Research

Amsterdam team: prof. dr. Jaap van Dieën, dr. Sjoerd Bruijn, dr. Bernadette van Wijk
dr. Moira van Leeuwen

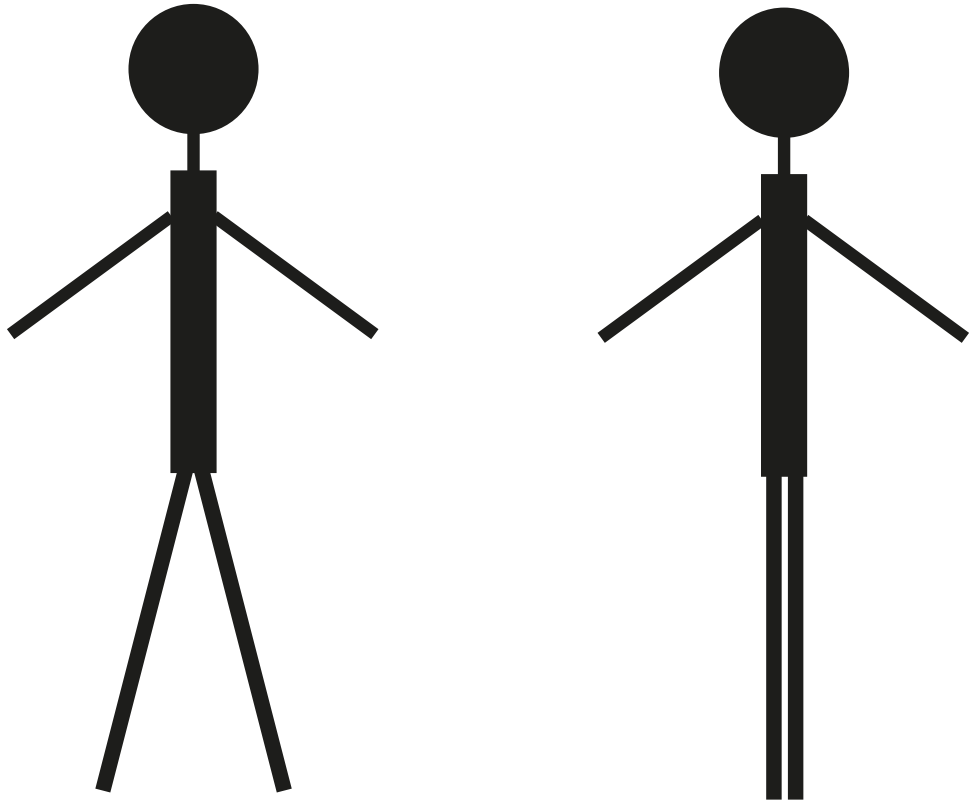
# Foot Placement Control & Corticomuscular Coherence

## Concepts & Analysis Pipeline

Amsterdam Movement Sciences
VU VRIJE UNIVERSITEIT AMSTERDAM
Amsterdam UMC University Medical Centers

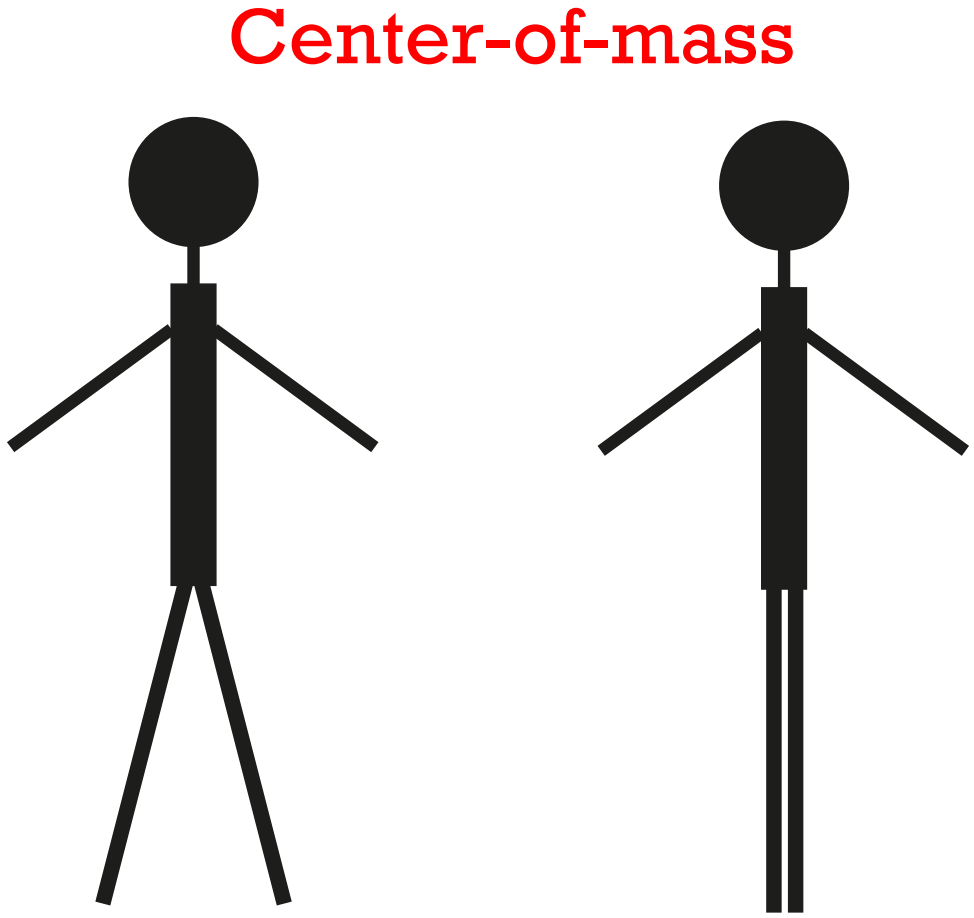
## Stability control

Coordinating the base of support wrt the center of mass



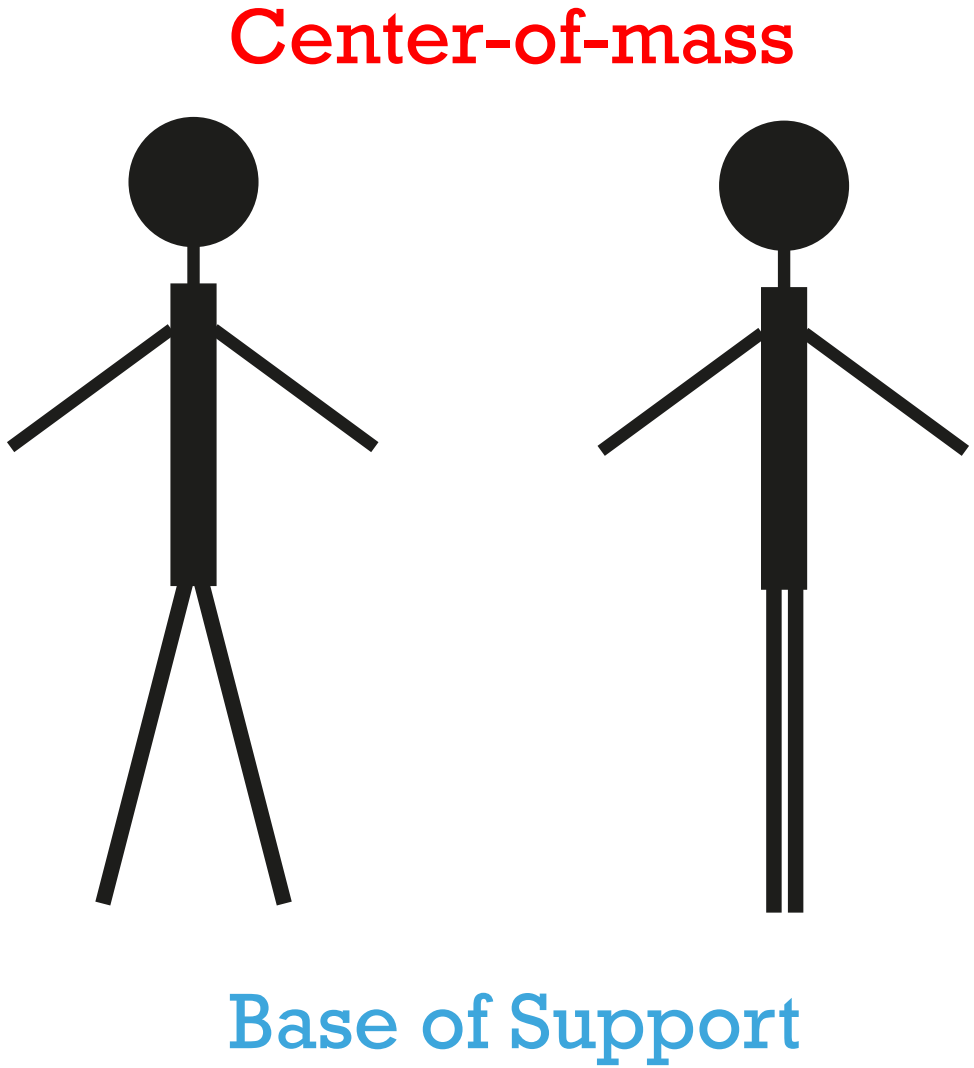
## Stability control

Coordinating the base of support wrt the center of mass



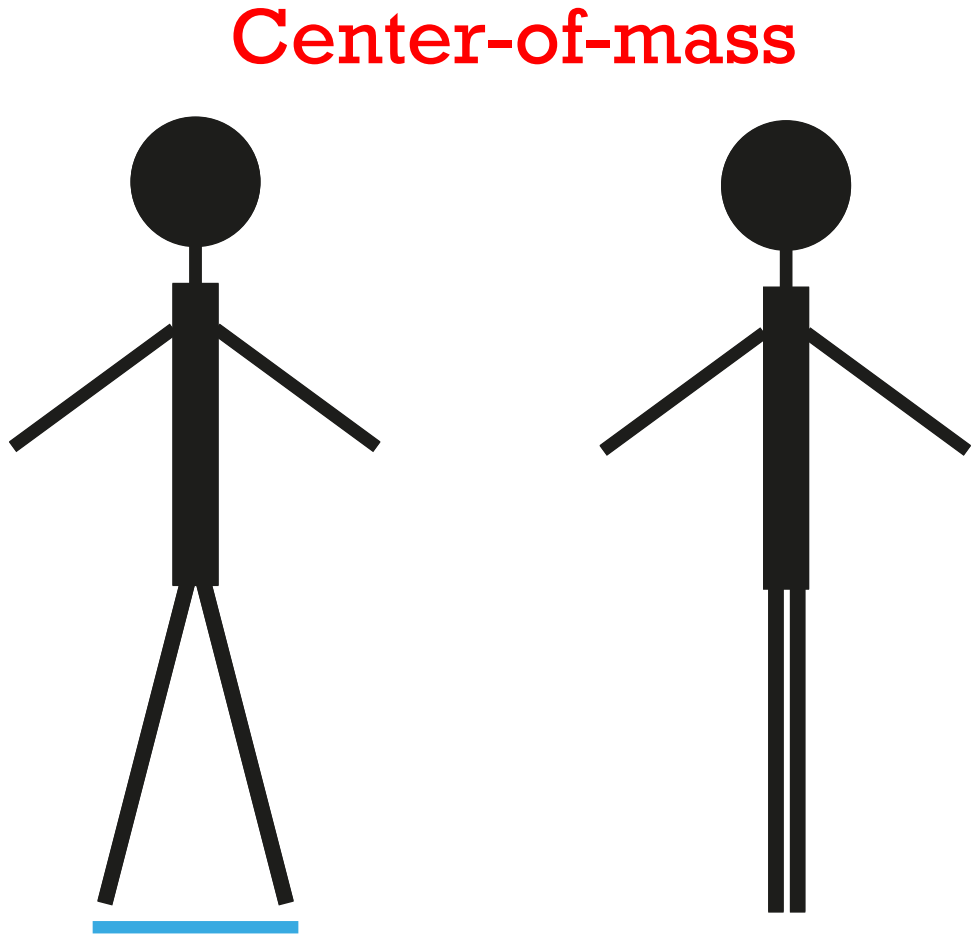
## Stability control

Coordinating the base of support wrt the center of mass



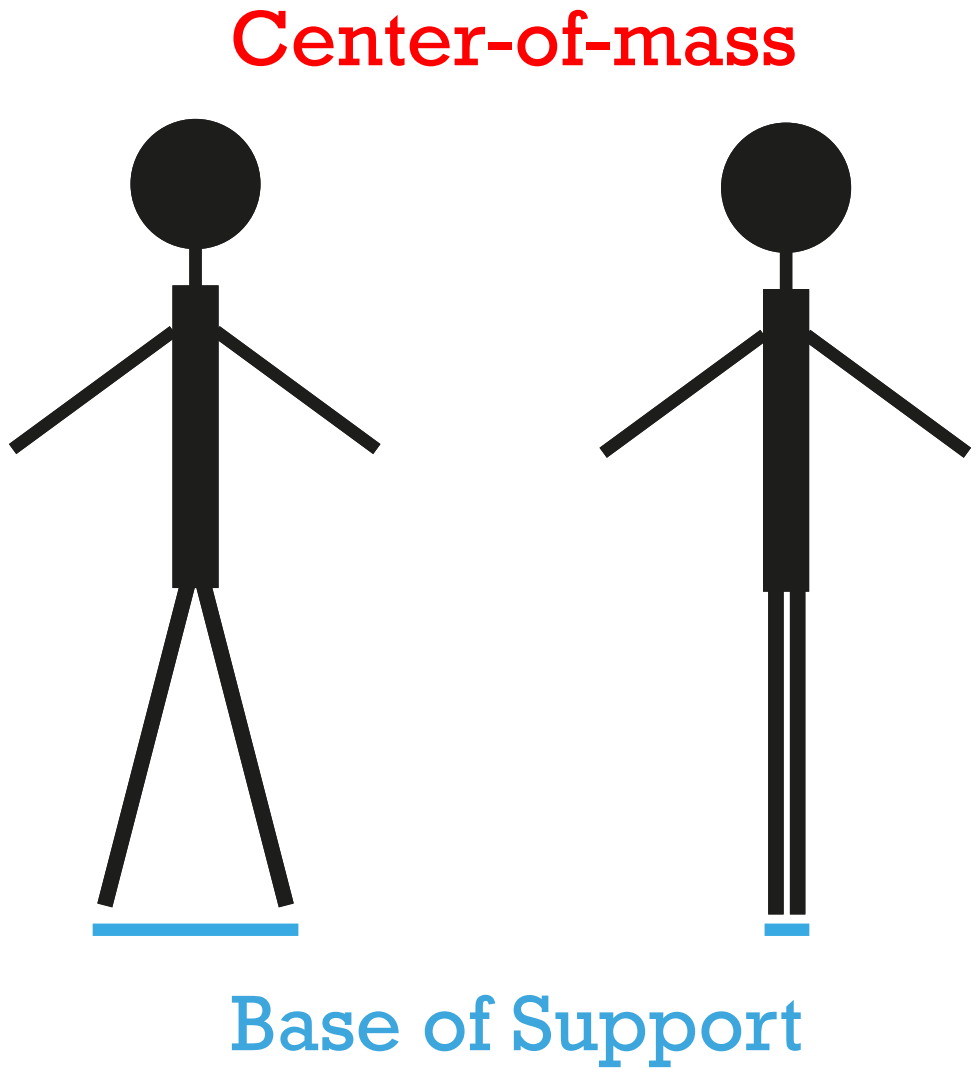
## Stability control

Coordinating the base of support wrt the center of mass



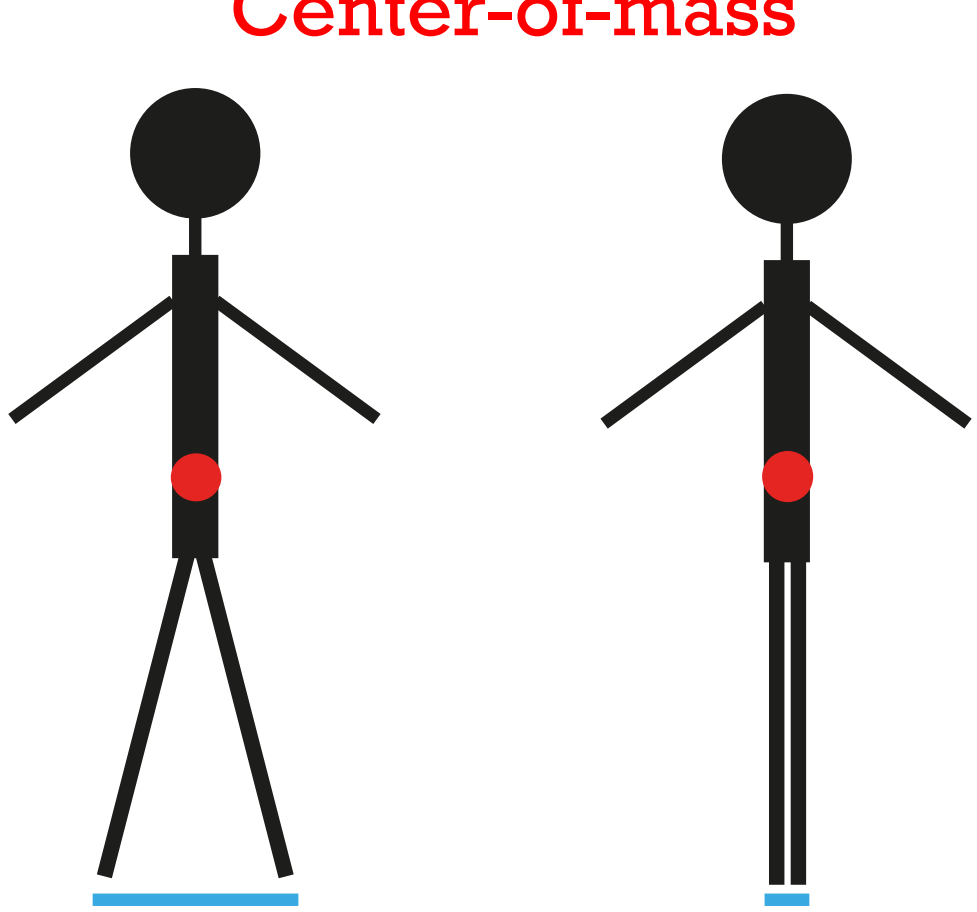
## Stability control

Coordinating the base of support wrt the center of mass



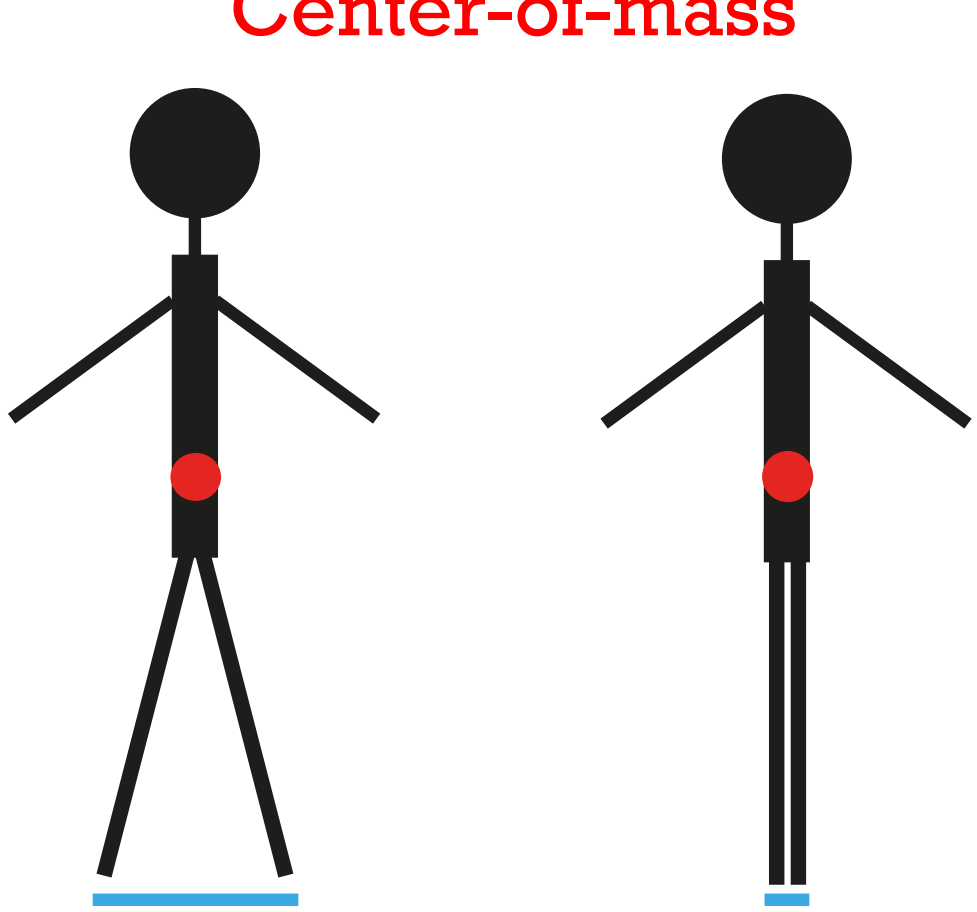
## Stability control

Coordinating the base of support wrt the center of mass



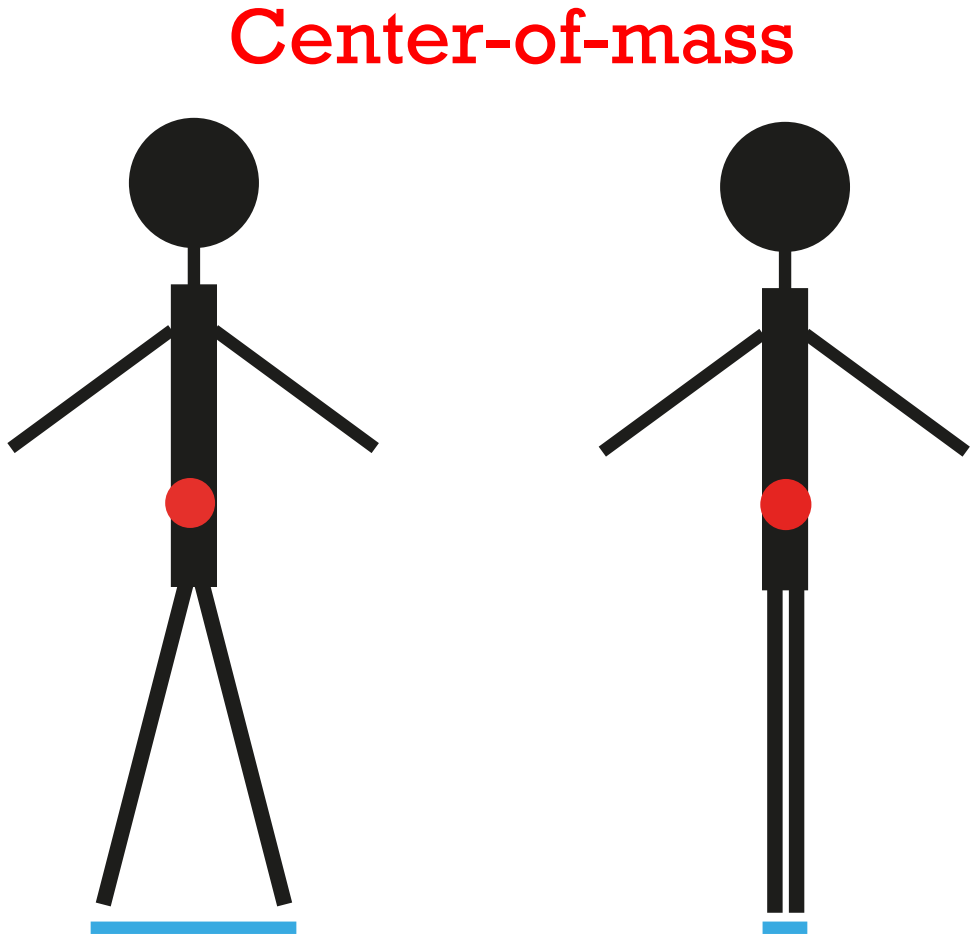
## Stability control

Coordinating the base of support wrt the center of mass



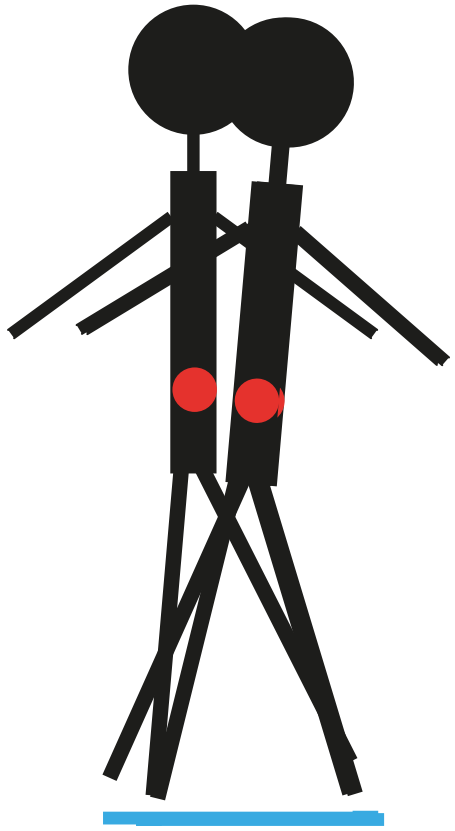
## Stability control

Coordinating the base of support wrt the center of mass



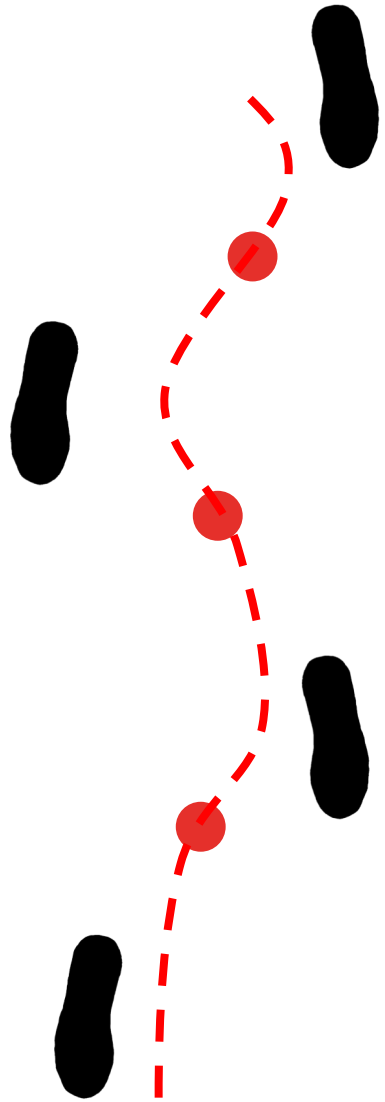
## Stability control

Foot placement is dominant stability mechanism during walking



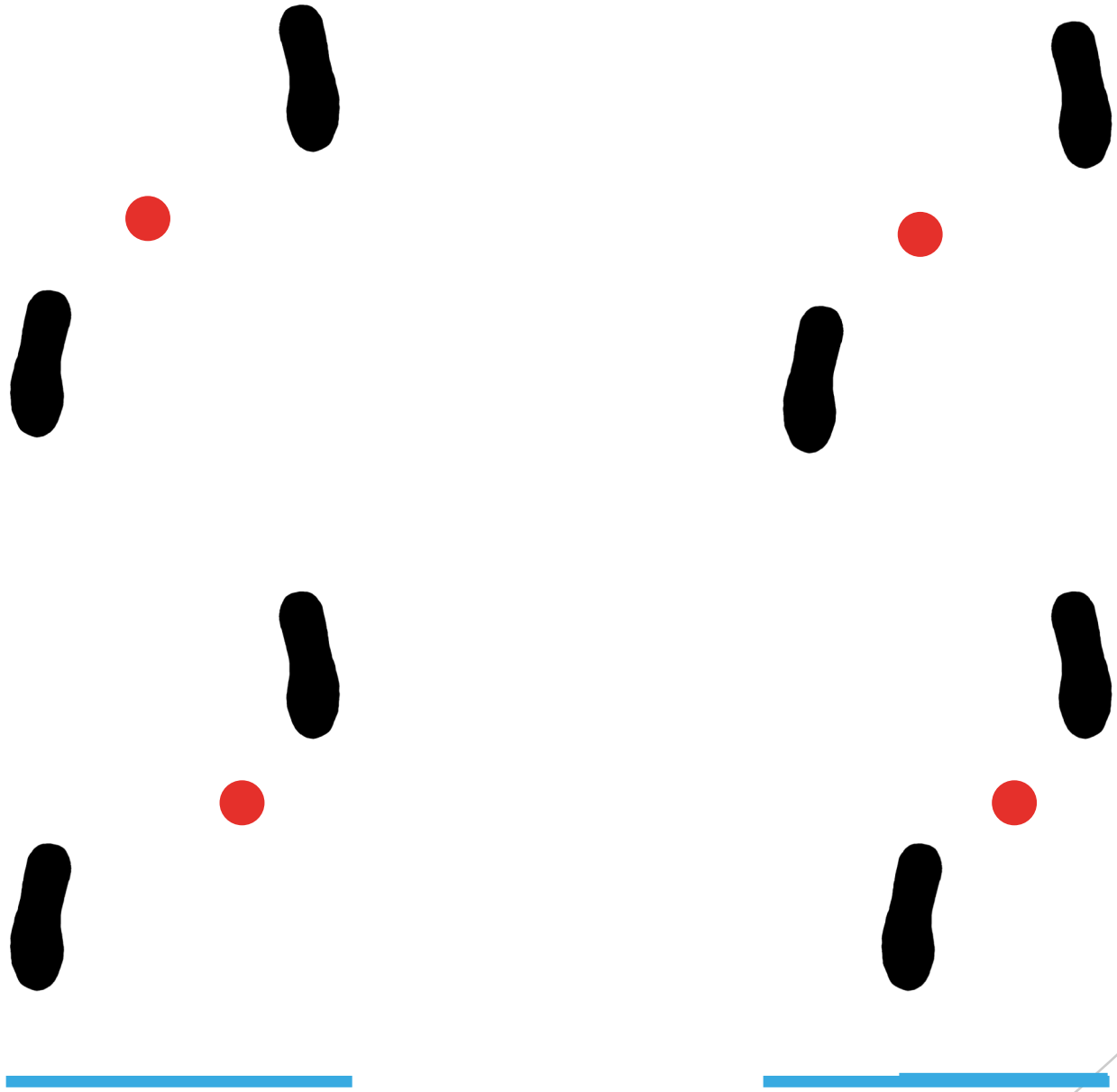
## Stability control

Foot placement is dominant stability mechanism during walking



## Stability control

Foot placement is dominant stability mechanism during walking



## Stability control

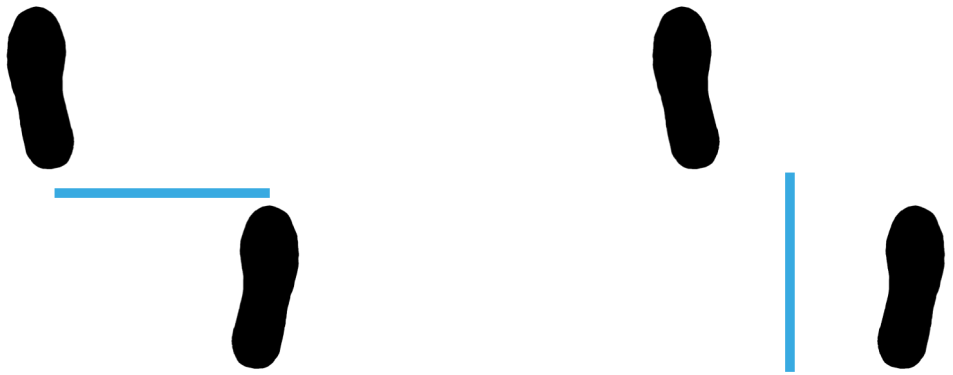
Foot placement control can be quantified using a linear feedback model

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

FP

Step Width

Step Length

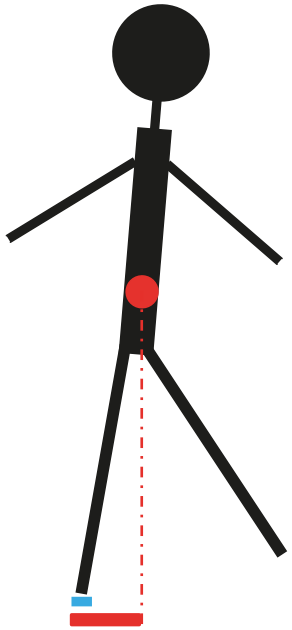


## Stability control

Foot placement control can be quantified using a linear feedback model

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

$CoM_{pos}$
$CoM_{vel}$



ML/AP
CoM position

Wrt stance leg

Derivative

## Stability control

Foot placement control can be quantified using a linear feedback model

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

| | |
|---|---|
| $\beta_{pos}$ | Regression coefficient<br>Gain |
| $\beta_{vel}$ | Strength of FP response |

## Stability control

Foot placement control can be quantified using a linear feedback model

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

$i$ Percentage of swing phase

## Stability control

Foot placement control can be quantified using a linear feedback model

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

$\varepsilon$ Residual

Foot placement error

Motor noise/loose control

## Stability control

Foot placement control can be quantified using a linear feedback model

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$


```
stats = regstats(foot_combined,pred_combined(:,1:order),'linear');
```

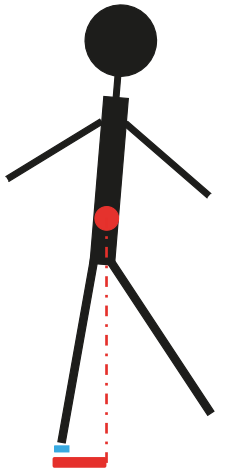
```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

## Predictors & Dependent variable

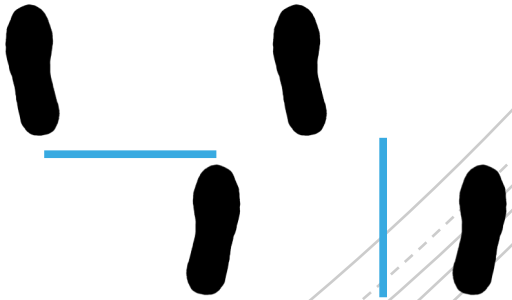
`pred_combined`

`foot_combined`

- Pelvis marker
- Heel markers

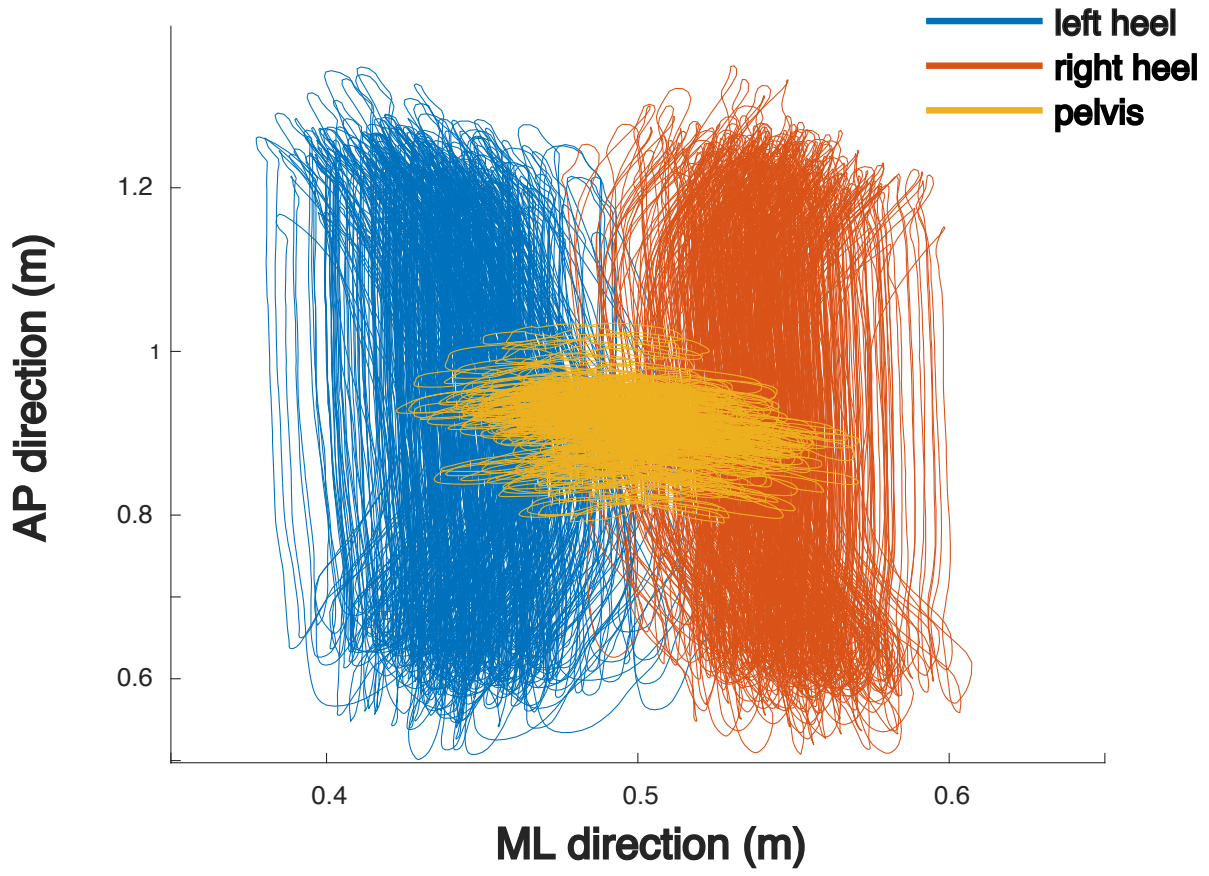


- Heel markers



Pelvis and heel markers

# Quick check: kinematic data



## Predictors & Dependent variable

`pred_combined`

`foot_combined`

Gait event detection

- Heelstrike & toe-off events
- Heel markers

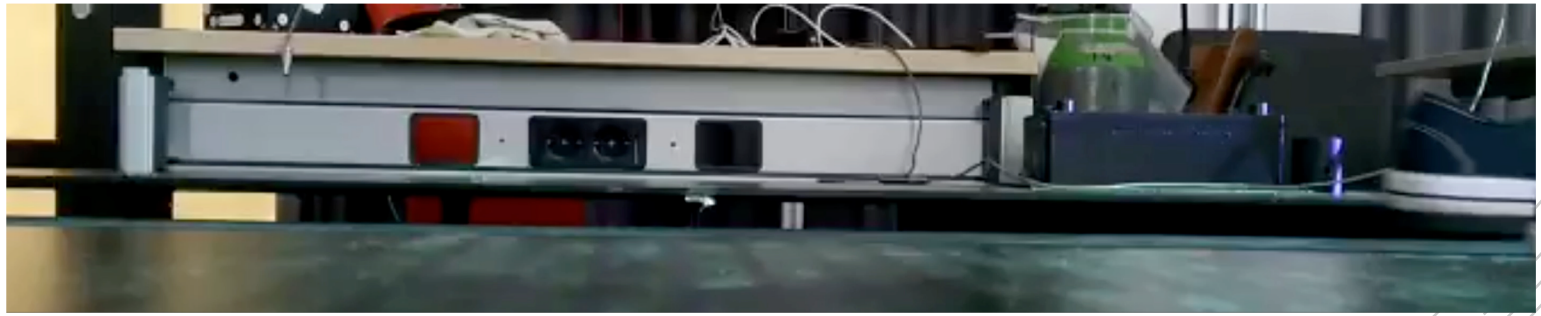
Gait event detection

Detecting events based on AP velocity

Zeni et al.( 2008)

# Anteriorior posterior velocity heelmarker

- AP velocity + → - : Heelstrike
- AP velocity - → + : Toe-off



## Gait event detection

# Detecting events based on AP velocity

Zeni et al.( 2008)

[lhs,rto,rhs,lto] = calc_events_zeni(L_heel_AP,R_heel_AP,fs,fs_t)

- L_heel_AP : left heelmarker position data AP direction
- R_heel_AP : right heelmarker position data AP direction
- fs: sampling frequency motion capture system
- fs_t: threshold to avoid false heelstrike detection within fs_t seconds of a detected heelstrike

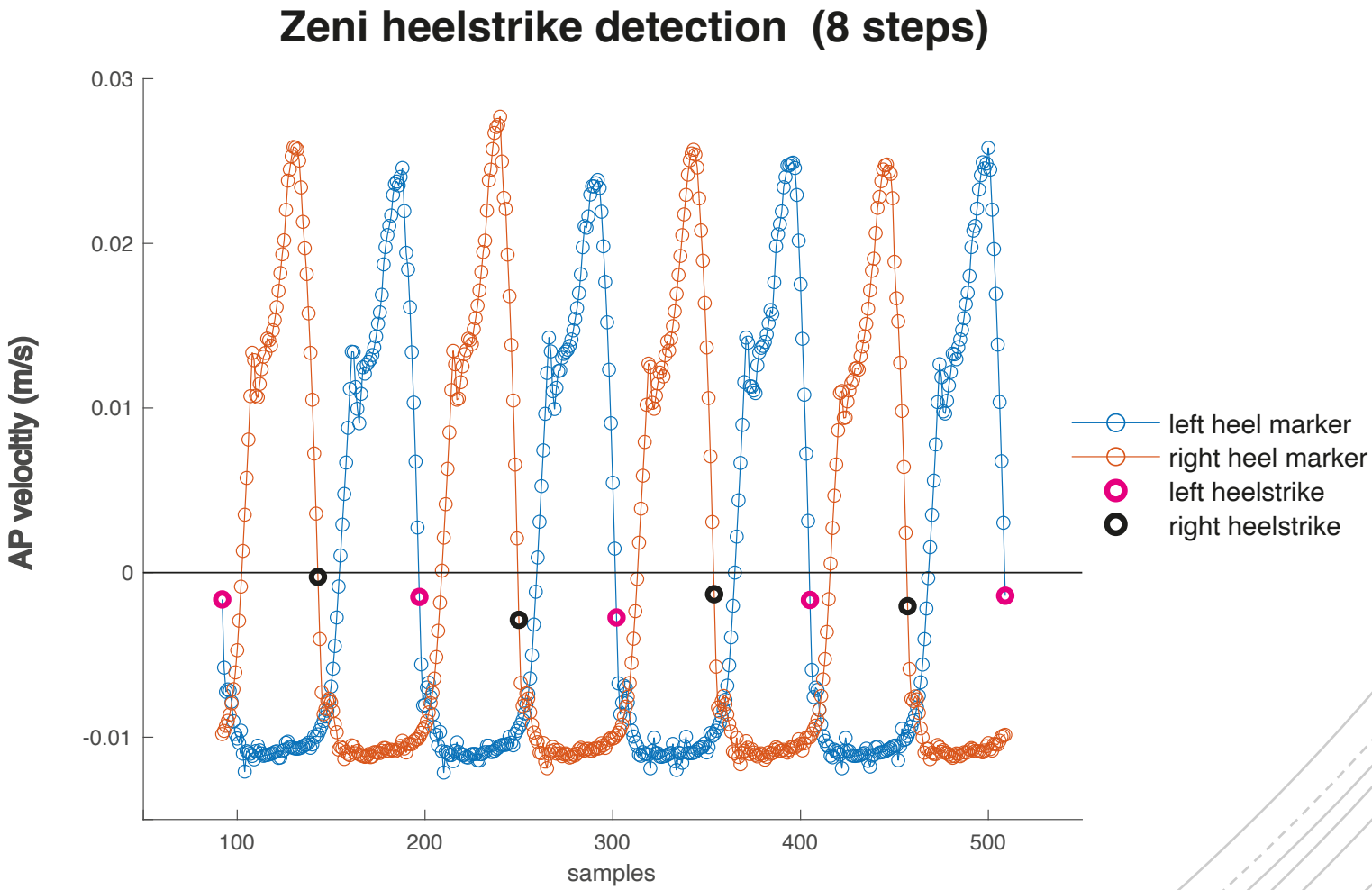
LHS RTO RHS LTO

## Gait event detection

# Detecting events based on AP velocity

Zeni et a.( 2008)

[lhs,rto,rhs,lto] = calc_events_zeni(L_heel_AP,R_heel_AP,fs,fs_t)

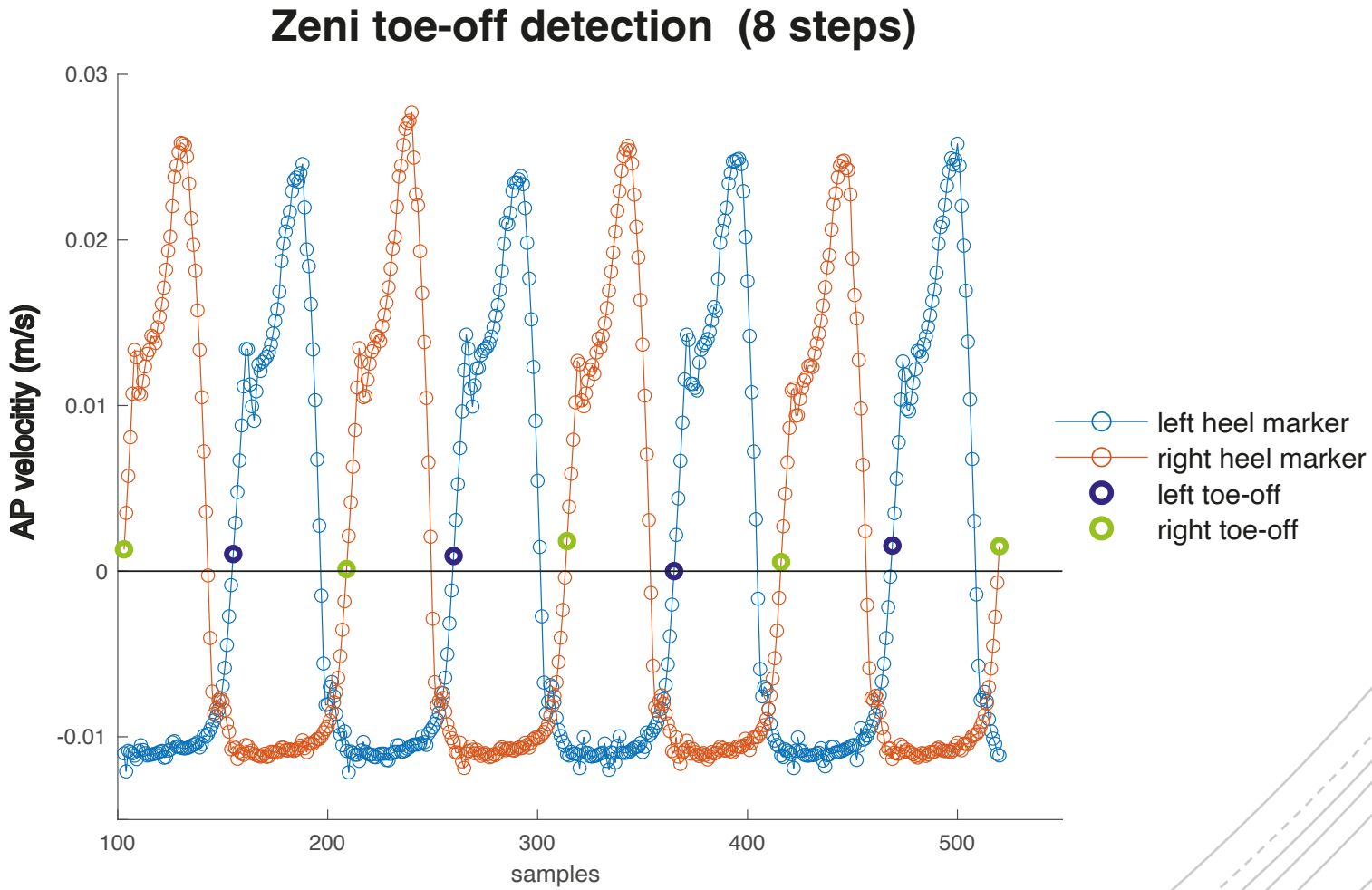


Gait event detection

# Detecting events based on AP velocity

Zeni et a.( 2008)

[lhs,rto,rhs,lto] = calc_events_zeni(L_heel_AP,R_heel_AP,fs,fs_t)

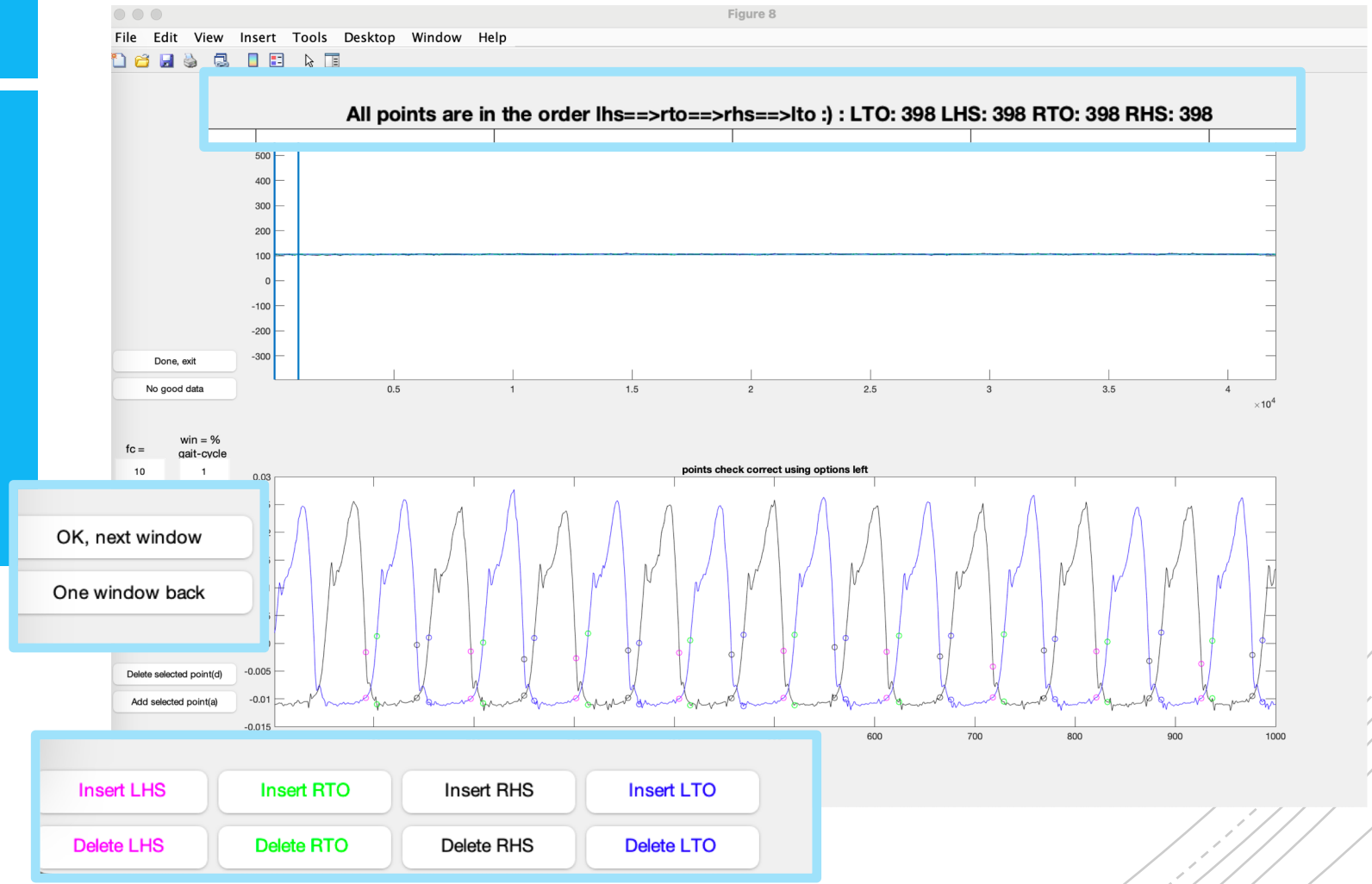


=VU 3D model=

[events,notgood]=check_signal(events, signal, fs)

# Gait event detection

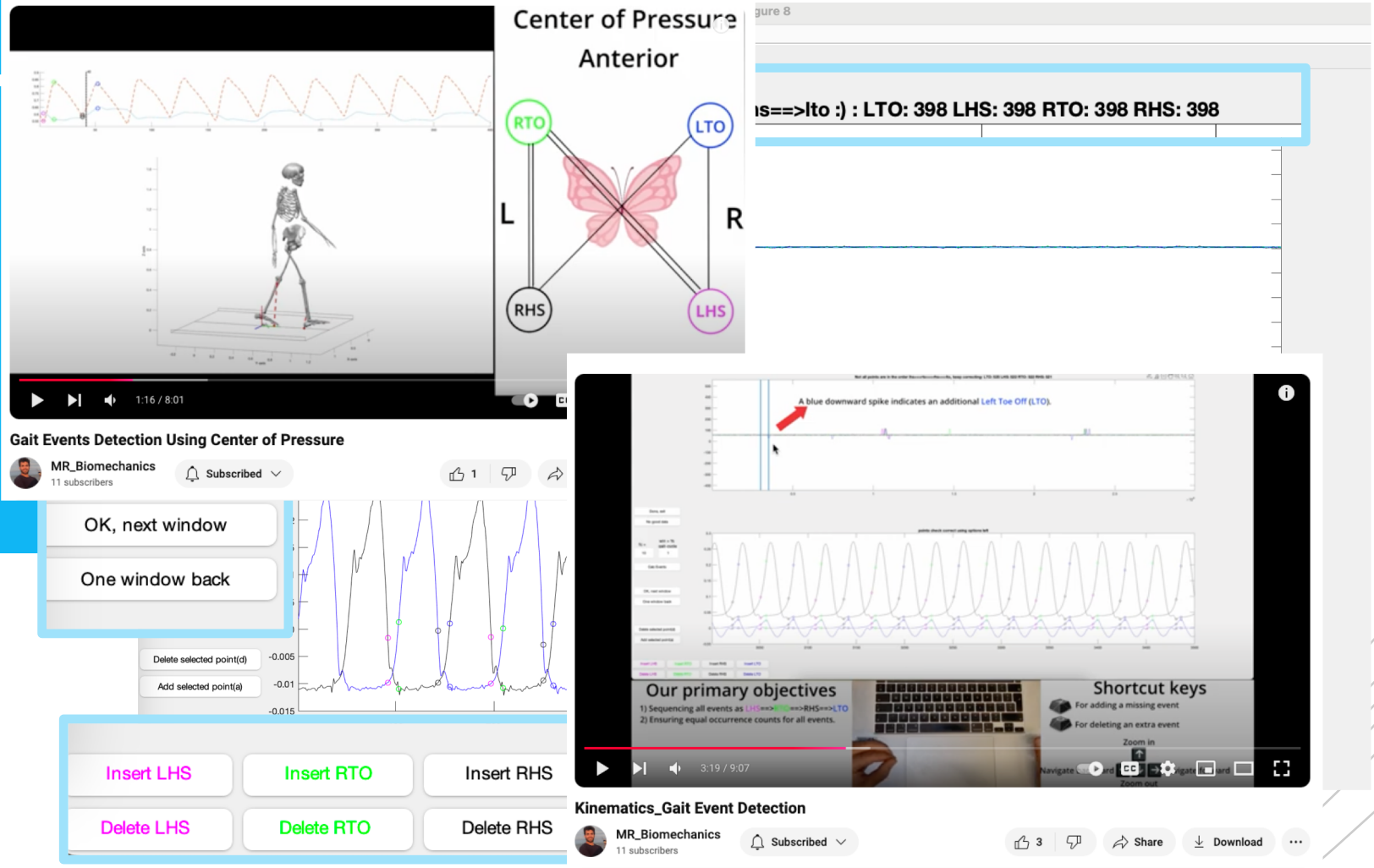
## Checking calculated events in GUI



[events,notgood]=check_signal(events, signal, fs)

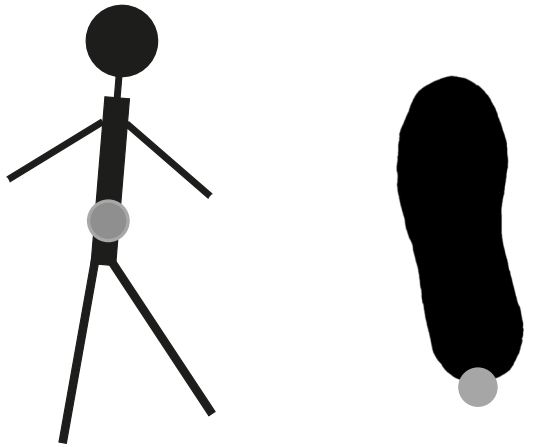
# Gait event detection

## Checking gait events in GUI

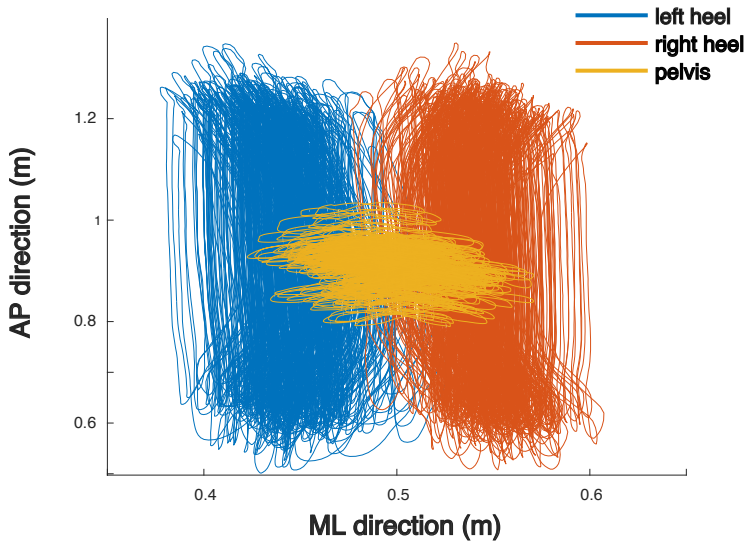


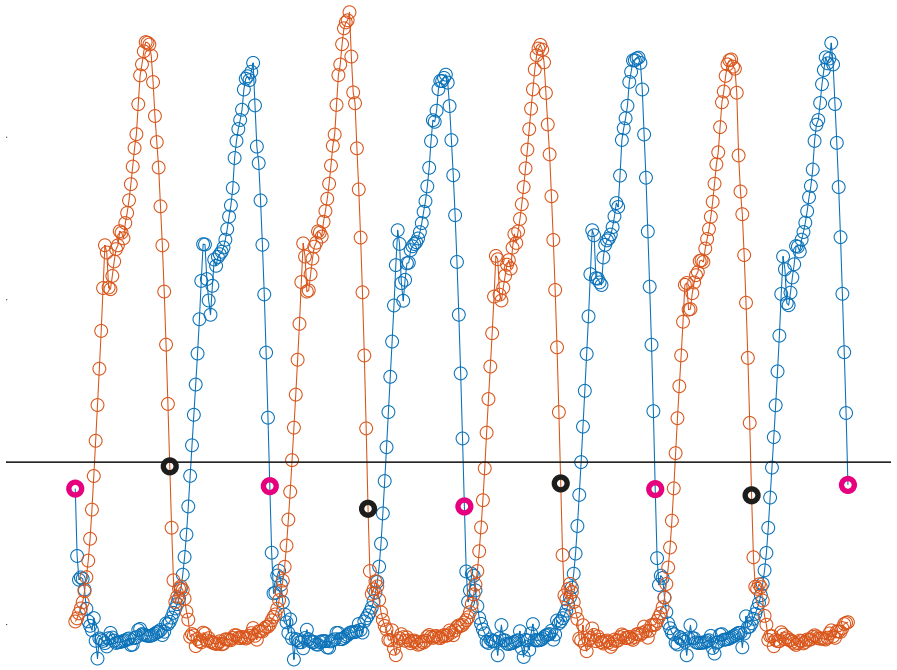
# Foot placement model

Pelvis marker
Heel markers
Gait events







```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

## Foot placement model

Function preparing
foot_combined
&
pred_combined

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
stats = regstats(foot_combined,pred_combined(:,1:order),'linear');
```

MATLAB®

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing
foot_combined
&
pred_combined

# COM

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing
foot_combined
&
pred_combined

# Rfoot Lfoot

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing
foot_combined
&
pred_combined

# events

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing
foot_combined
&
pred_combined

# fsopto

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing foot_combined & pred_combined

# pred_samples 1:51

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing
foot_combined
&
pred_combined

# order

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

## Foot placement model

Function preparing
foot_combined
&
pred_combined

# removeorigin

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

Foot placement model

Function preparing
foot_combined
&
pred_combined

# centerdata

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

## Foot placement model

Function preparing
foot_combined
&
pred_combined

- Taking derivatives
- Time normalization
- Removing origin
- Centering the data
- Combining Left and Right steps

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Taking derivatives

Foot placement model

Function preparing foot_combined & pred_combined

**COM** **fsopto**

```
COM_vel = calc_derivative(COM,fsopto);
COM_acc = calc_derivative(COM_vel,fsopto);
```

**order**

- **1 → CoM position only**
- **2 → CoM position and velocity**
- **3 → CoM position and velocity and acceleration**

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Time normalization

Foot placement model

Function preparing
foot_combined
&
pred_combined

**COM** **Rfoot** **Lfoot** **events**

```
COM_L = normalizetimebase_step(COM,lto(1:end-1),lhs(2:end));
COM_L_vel = normalizetimebase_step(COM_vel,lto(1:end-1),lhs(2:end));
```

**Swing phase preceding foot placement**

```
foot_L = normalizetimebase_step(Lfoot,rto(2:end),rhs(2:end));
```

**Single stance phase of the placed foot**

```
origin_L = normalizetimebase_step(Rfoot,lto(1:end-1),lhs(2:end));
```

**Single stance phase of stance foot wrt the other foot is placed to**

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Removing origin

## (… with respect to?)

`removeorigin`

Foot placement model

Function preparing
foot_combined
&
pred_combined

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

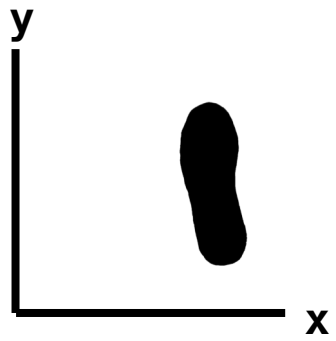
Foot placement model

Function preparing
foot_combined
&
pred_combined

# Removing origin

## (… with respect to?)

removeorigin  RfootLfoot



```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

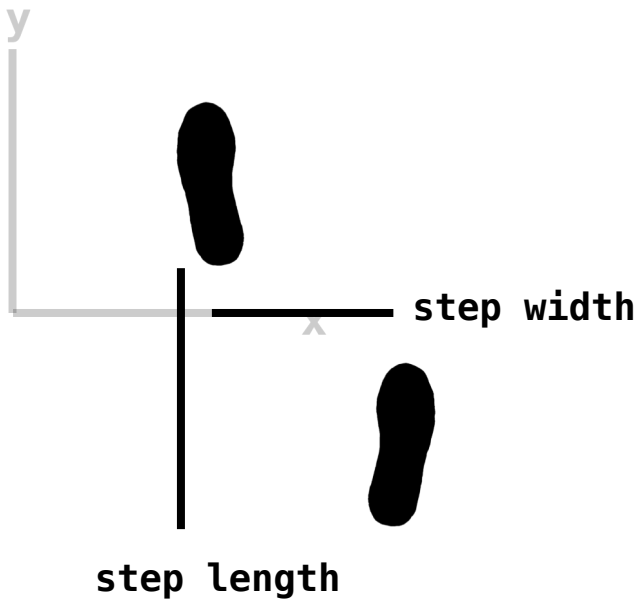
Foot placement model

Function preparing
foot_combined
&
pred_combined

# Removing origin

## (… with respect to?)

removeorigin Rfoot Lfoot



```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

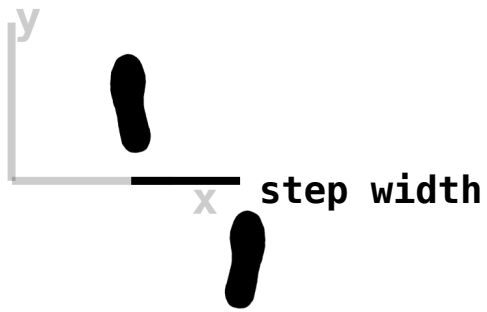
# Removing origin

## (… with respect to?) Mediolateral

Foot placement model

Function preparing
foot_combined
&
pred_combined

removeorigin




```
foot_L = foot_L(25,:)';
origin_L = origin_L(25,:)';
foot_R = foot_R(25,:)';
origin_R = origin_R(25,:)';


if removeorigin
foot_L = foot_L -origin_L;
foot_R = foot_R -origin_R ;
end
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

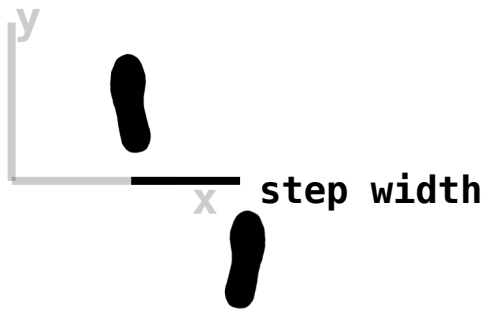
# Removing origin

## (… with respect to?) Mediolateral midstance

Foot placement model

Function preparing foot_combined & pred_combined

removeorigin Rfoot Lfoot




```
foot_L = foot_L(25,:)';
origin_L = origin_L(25,:)';
foot_R = foot_R(25,:)';
origin_R = origin_R(25,:)';


if removeorigin
foot_L = foot_L -origin_L;
foot_R = foot_R -origin_R ;
end
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

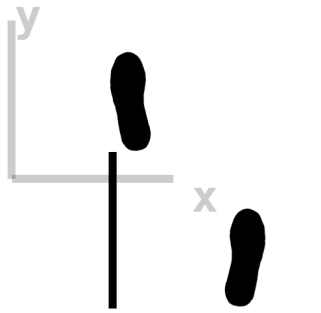
# Removing origin

## (… with respect to?) Anteroposterior

Foot placement model

Function preparing
foot_combined
&
pred_combined

removeorigin  Rfoot Lfoot




step length

```
foot_L = Lfoot(lhs(2:end));
origin_L = origin_L(end,:)';
foot_R = Rfoot(rhs(1:end-1));
origin_R = origin_R(end,:)';


if removeorigin
foot_L = foot_L -origin_L;
foot_R = foot_R -origin_R ;
end
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Removing origin

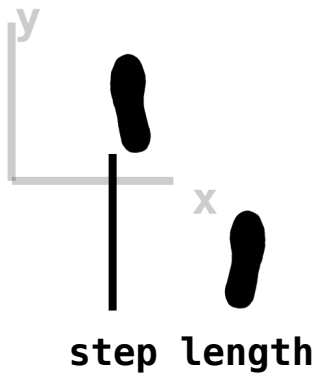
## (… with respect to?) Anteroposterior

Foot placement model

Function preparing
foot_combined
&
pred_combined

removeorigin  Rfoot Lfoot

heelstrike




```
foot_L = Lfoot(lhs(2:end));
origin_L = origin_L(end,:)';
foot_R = Rfoot(rhs(1:end-1));
origin_R = origin_R(end,:)';


if removeorigin
foot_L = foot_L -origin_L;
foot_R = foot_R -origin_R ;
end
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Centering the data

## (subtracting the mean)

Foot placement model

Function preparing foot_combined & pred_combined

centerdata

```
if centerdata
foot_L = foot_L -nanmean(foot_L);
foot_R = foot_R -nanmean(foot_R);
end
```

```
if centerdata
pred_Lstance = pred_Lstance-
repmat(nanmean(pred_Lstance),size(pred_Lstance,1),1);
pred_Rstance = pred_Rstance-
repmat(nanmean(pred_Rstance),size(pred_Rstance,1),1);
end
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Centering the data

## (subtracting the mean)

Foot placement model

Function preparing foot_combined & pred_combined

centerdata

```
if centerdata
foot_L = foot_L -nanmean(foot_L);
foot_R = foot_R -nanmean(foot_R);
end
```

```
if centerdata
pred_Lstance = pred_Lstance-
repmat(nanmean(pred_Lstance),size(pred_Lstance,1),1);
pred_Rstance = pred_Rstance-
repmat(nanmean(pred_Rstance),size(pred_Rstance,1),1);
end
```

$$FP = \text{mean\_step\_width} + \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Combining Left & Right

Foot placement model

Function preparing foot_combined & pred_combined

## Mediolateral

```
foot_combined = [foot_R_sample ; - foot_L_sample];
pred_combined = [pred_Rstance(:,1:order); - pred_Lstance(:,1:order)];
```

## Anteroposterior

```
foot_combined = [foot_R_sample ; foot_L_sample];
pred_combined = [pred_Rstance(:,1:order); pred_Lstance(:,1:order)];
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

# Combining Left & Right

Foot placement model

Function preparing
foot_combined
&
pred_combined

## Mediolateral

```
foot_combined = [foot_R_sample ; - foot_L_sample];
pred_combined = [pred_Rstance(:,1:order); - pred_Lstance(:,1:order)];
```

## Anteroposterior

```
foot_combined = [foot_R_sample ; foot_L_sample];
pred_combined = [pred_Rstance(:,1:order); pred_Lstance(:,1:order)];
```

## Foot placement model

## Outcome measures

```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

## Foot placement model

## Outcome measures

```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;
```

$$R^2 = 1 - \frac{\sum(\text{actual FP} - \text{predicted FP})^2}{\sum(\text{actual FP} - \text{mean FP})^2}$$

## Foot placement model

## Outcome measures

```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;
```

$$R^2 = 1 - \frac{unexplained\ FP\ variance}{total\ FP\ variance}$$

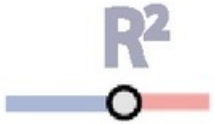
## Foot placement model

# Outcome measures

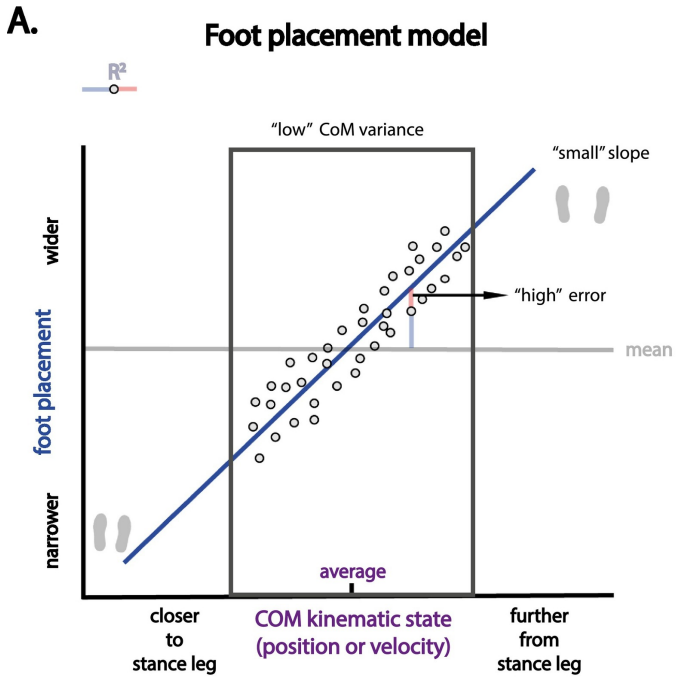
```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;
```




$$R^2 = 1 - \frac{unexplained\ FP\ variance}{total\ FP\ variance}$$



## Foot placement model

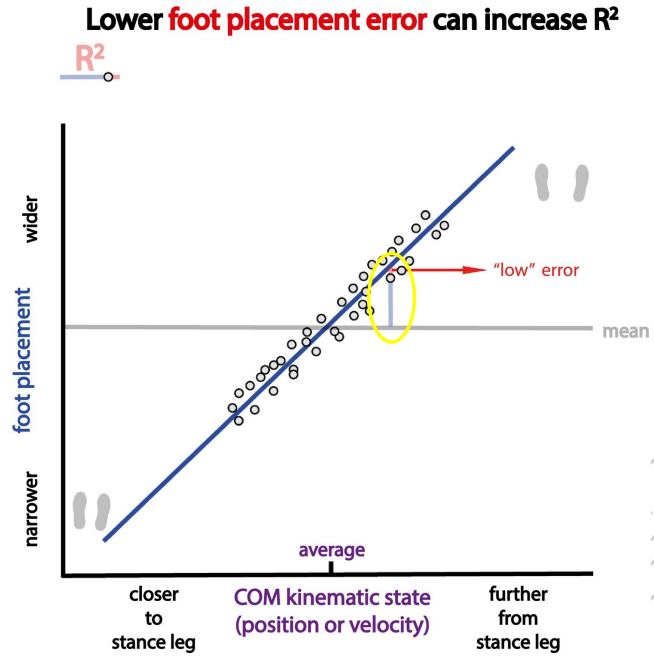
# Outcome measures

```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;

intermediates.error_combined(i_pred_sample,ind_combined) = stats.r;
```


$$R^2 = 1 - \frac{unexplained\ FP\ variance}{total\ FP\ variance}$$



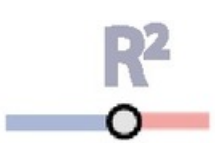
## Foot placement model

# Outcome measures

```
stats = regstatsfoot_combined,pred_combined(:,1:order),'linear');
```

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

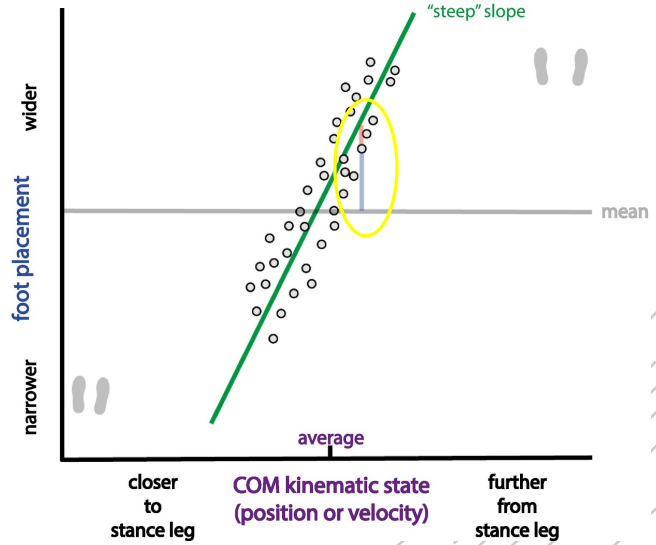
```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;

OUT.Combined_coeff1.data(i_pred_sample) = stats.beta(2);

OUT.Combined_coeff1.data(i_pred_sample) = stats.beta(3);
```





Higher gain (β) can increase $R^2$


$$R^2 = 1 - \frac{unexplained\ FP\ variance}{total\ FP\ variance}$$

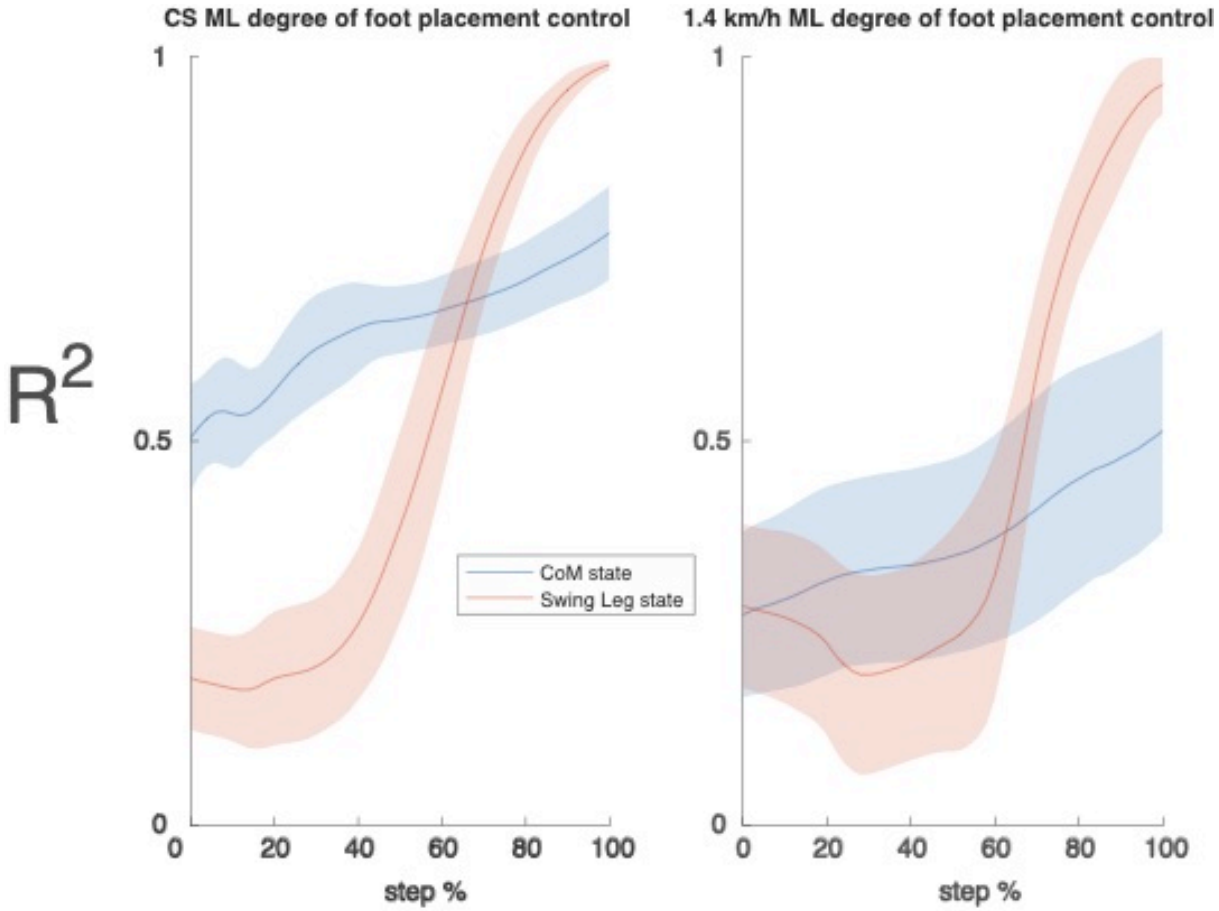


```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

## Foot placement model

## Outcome measures



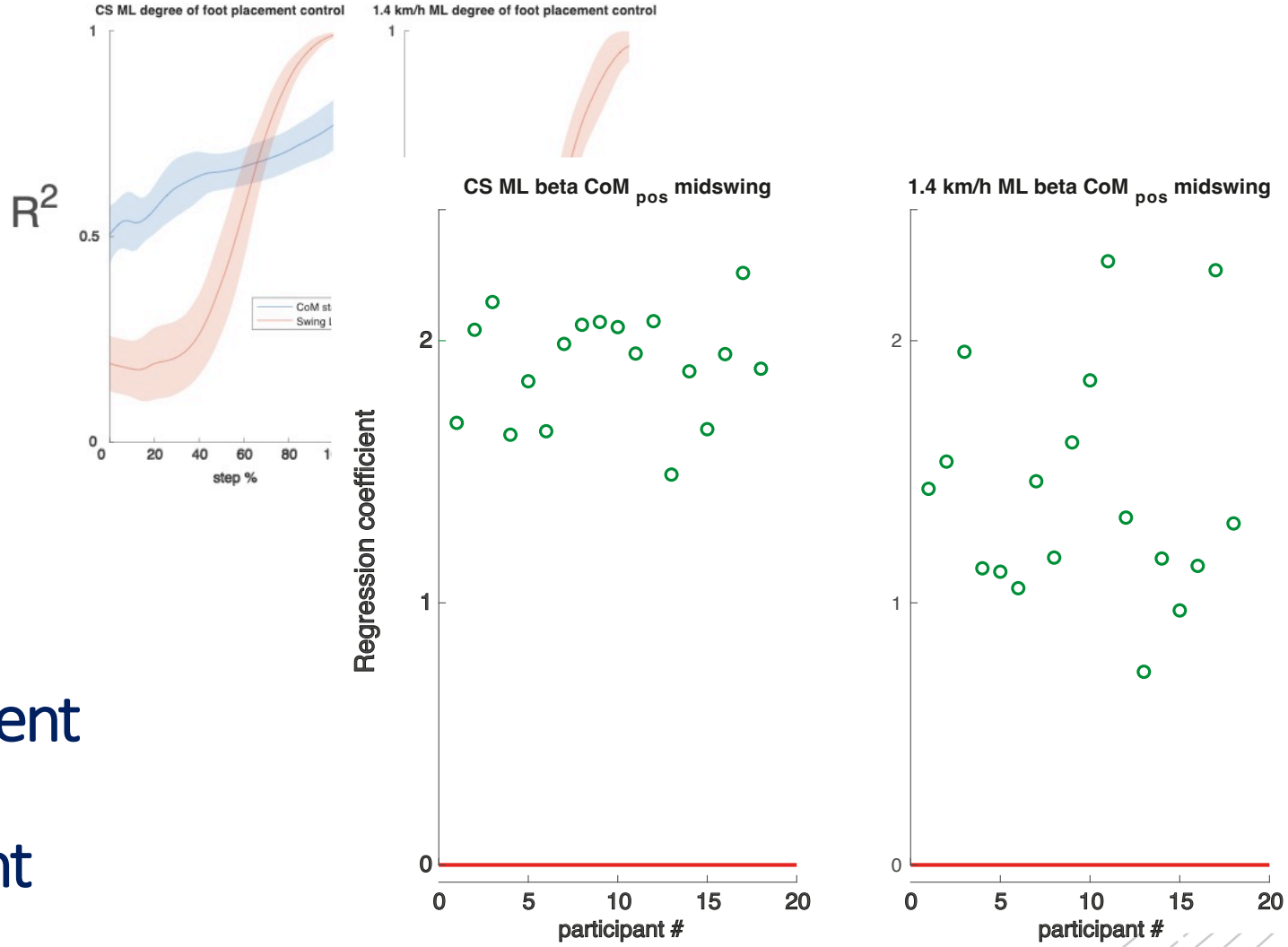
```
OUT.Combined_pct.data(i_pred_sample) = stats.rsquare;
```

```
[OUT,intermediates] =
foot_placement_model_function_step(COM,Rfoot,Lfoot,events,fsopto,pred_samples,order,removeorigin,centerdata)
```

## Foot placement model

## Outcome measures

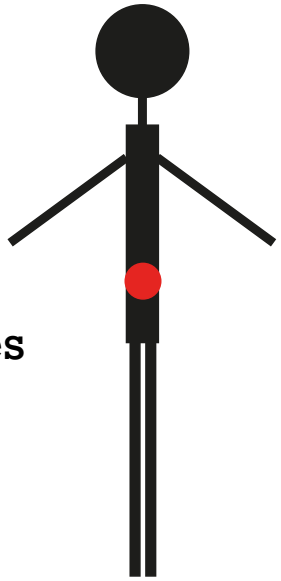
More lateral CoM displacement
leads to
more lateral foot placement



## Foot placement model

# Analysis pipeline

- Measuring kinematic data: Pelvis, heel markers
- Computing gait events
- Preparing the predictors (ML/AP CoM kinematic state)
- Preparing the dependent variable (step width/step length)
- Performing the linear regression
- Interpreting the outcome measures



# Foot placement model

## Key functions

- Gait events:
  - calc_events_zeni
  - check_signal (*VU3Dmodel*)
- Foot placement model
  - foot_placement_model_function_step_ML
  - foot placement_model_fuction_step_AP
    - calc_derivative (*VU3Dmodel*)
    - normalizetimebase_step
    - regstats

MATLAB®

# Foot placement model

## Scripts

- main1_gaitevents
  - Compare two methods of gait event detection
  - Why would we use Zeni for Parkinson's patients?
- main2_checkdata
- main3_FPmodel & main4_Fpplots
  - Compare ML and AP results
  - Look at different outcome measures ($R^2$, gains, foot placement error)
  - Look at the results for the predictor (CoM kinematic state) at different percentages of the swing phase.
  - At which percentage of the step cycle is the CoM kinematic state the most meaningful as a predictor?
  - Which outcome measure do you expect to improve for Parkinson's patients?

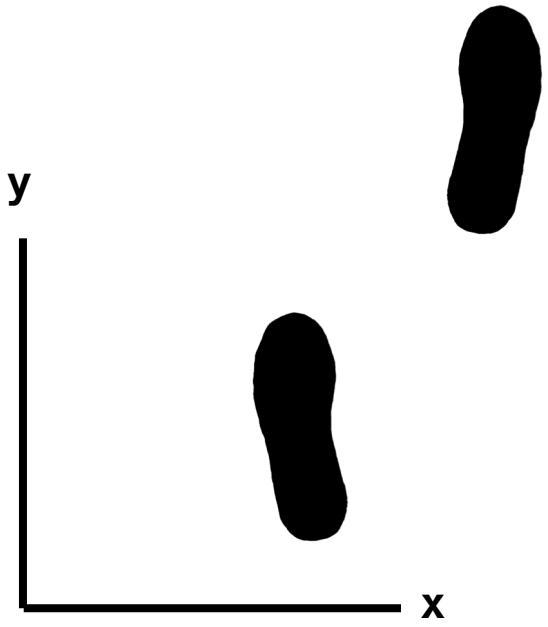
## Final note

# Coordinate system

- Stick to the consortium agreement
- Use same coordinate system for all subjects and trials
- Make a note of the coordinate system you define

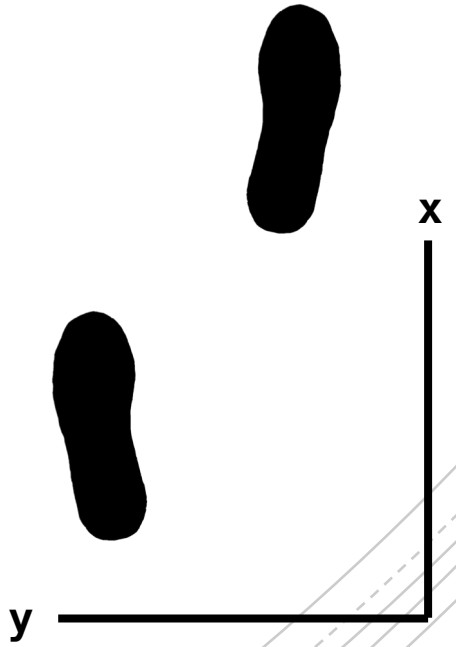
## Today's dataset:

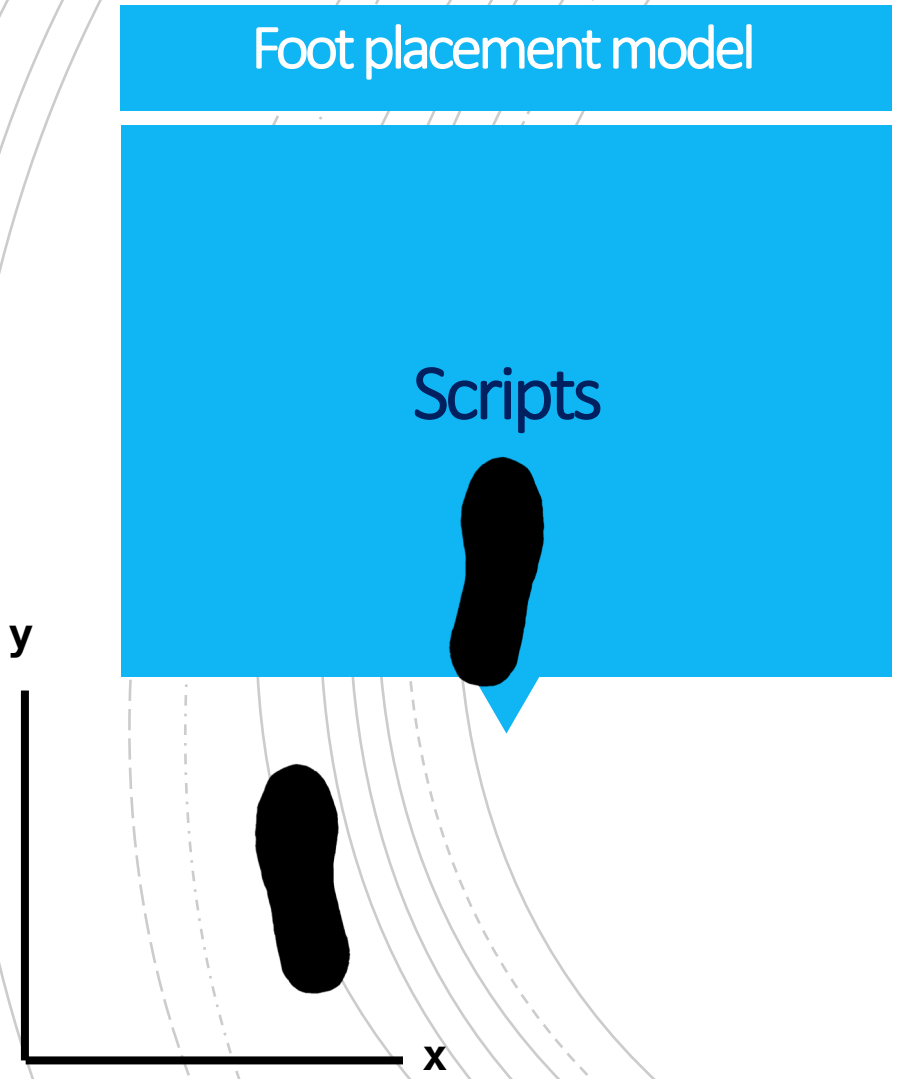
- x: mediolateral (right +)
- y: anteroposterior (forward +)
- z: vertical (upward+)




## Consortium agreement:

- x: anteroposterior (forward +)
- y: mediolateral (left +)
- z: vertical (upward +)






- main1_gaitevents
  - Compare two methods of gait event detection
  - Why would we use Zeni for Parkinson's patients?
- main2_checkdata
- main3_FPmodel & main4_Fpplots
  - Compare ML and AP results
  - Look at different outcome measures ($R^2$, gains, foot placement error)
  - Look at the results for the predictor (CoM kinematic state) at different percentages of the swing phase.
  - At which percentage of the step cycle is the CoM kinematic state the most meaningful as a predictor?
  - Which outcome measure do you expect to improve for Parkinson's patients?

JPND research
EU Joint Programme – Neurodegenerative Disease Research

Amsterdam team: prof. dr. Jaap van Dieën, dr. Sjoerd Bruijn, dr. Bernadette van Wijk
dr. Moira van Leeuwen

# Foot Placement Control & Corticomuscular Coherence

## Concepts &

## Analysis Pipeline

Amsterdam Movement Sciences
VU VRIJE UNIVERSITEIT AMSTERDAM
Amsterdam UMC University Medical Centers

JPND research
EU Joint Programme – Neurodegenerative Disease Research

Amsterdam team: prof. dr. Jaap van Dieën, dr. Sjoerd Bruijn, dr. Bernadette van Wijk
dr. Moira van Leeuwen

Foot Placement Control
&

# Corticomuscular Coherence

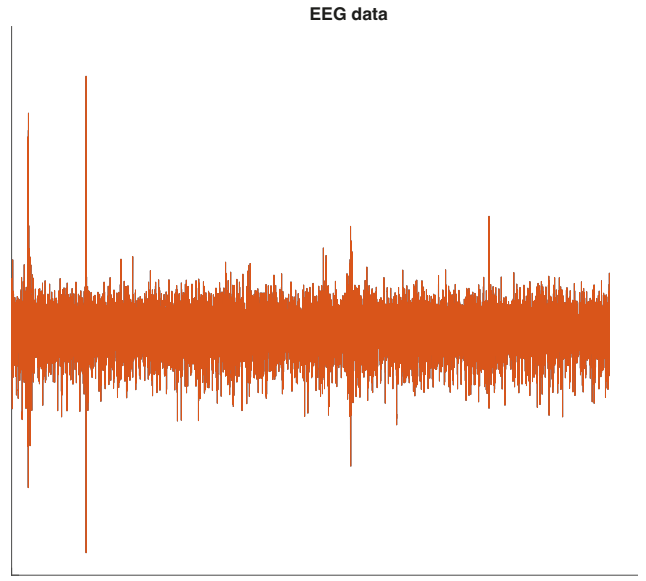
## Concepts &

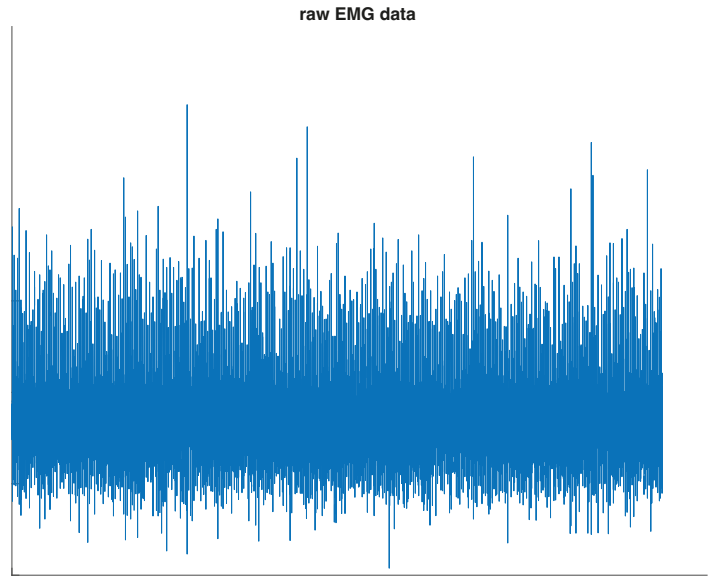
## Analysis Pipeline

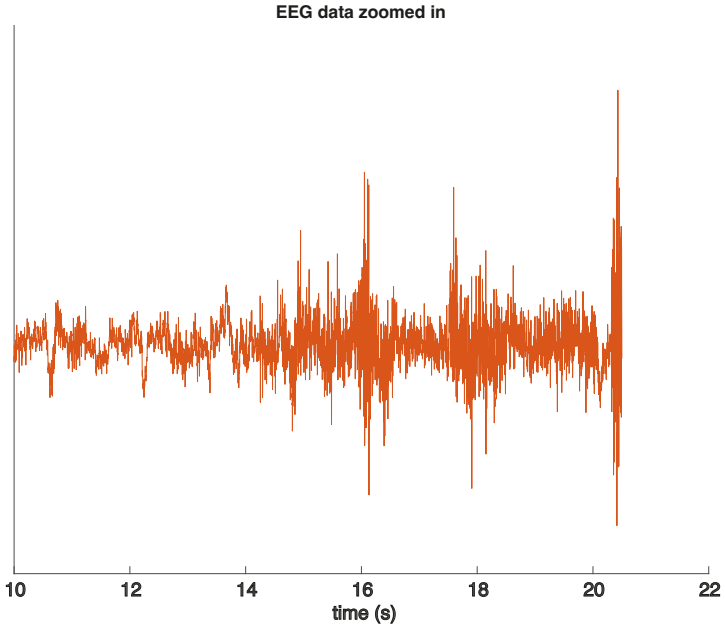
Amsterdam Movement Sciences
VU VRIJE UNIVERSITEIT AMSTERDAM
Amsterdam UMC University Medical Centers

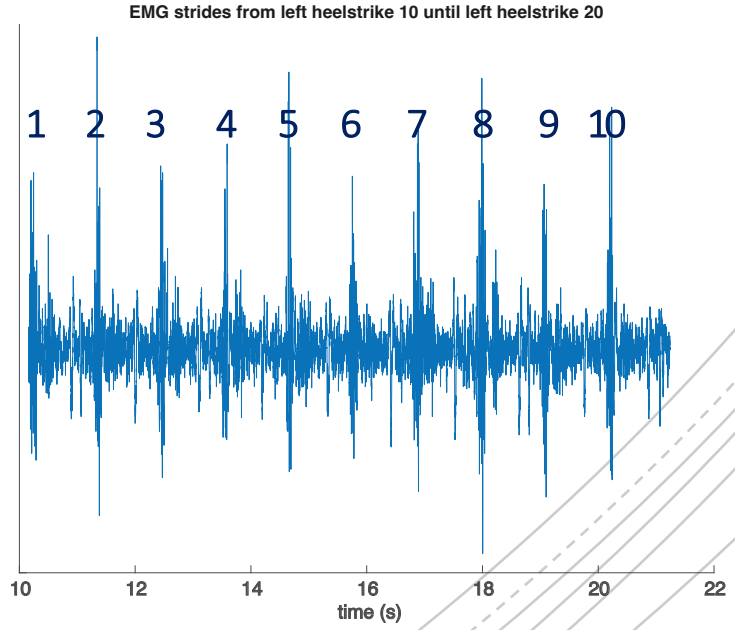
# Corticomuscular coherence

EMG
&
EEG data



plot_stride_av_EMG_ft

```
muscle_name = 
norm_val = 100;

plot_stride_av_EMG_ft(muscle_name,emg_data,events_EMG,norm_val)
```

## Corticomuscular coherence

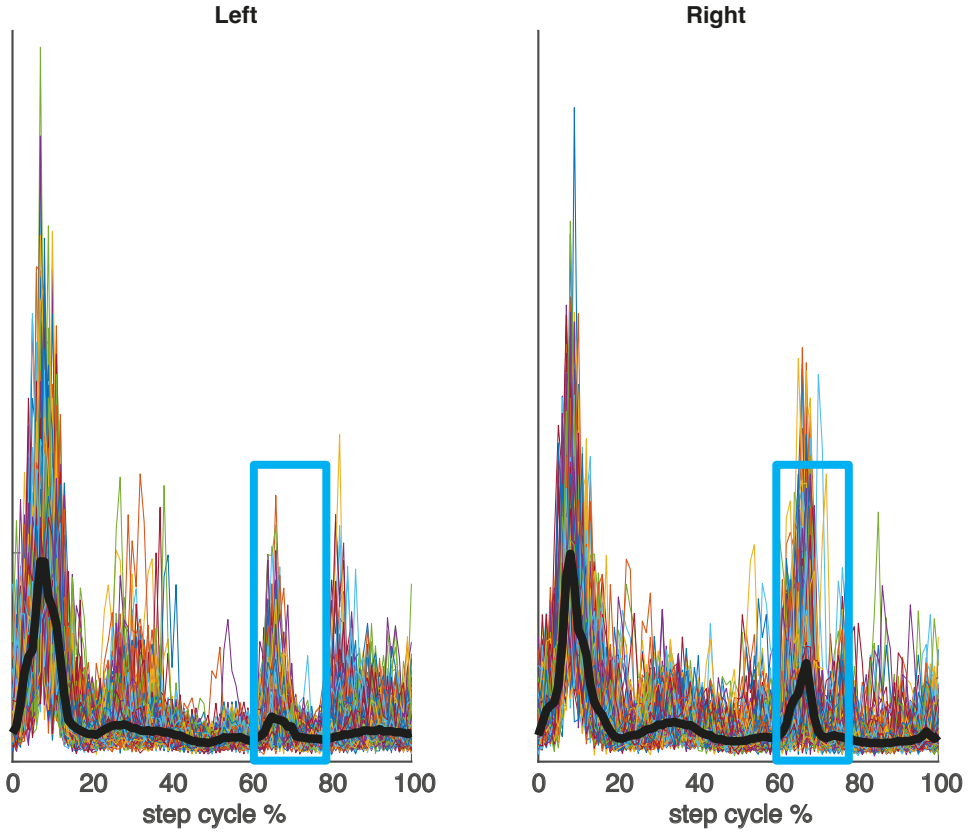
# Quick check: EMG

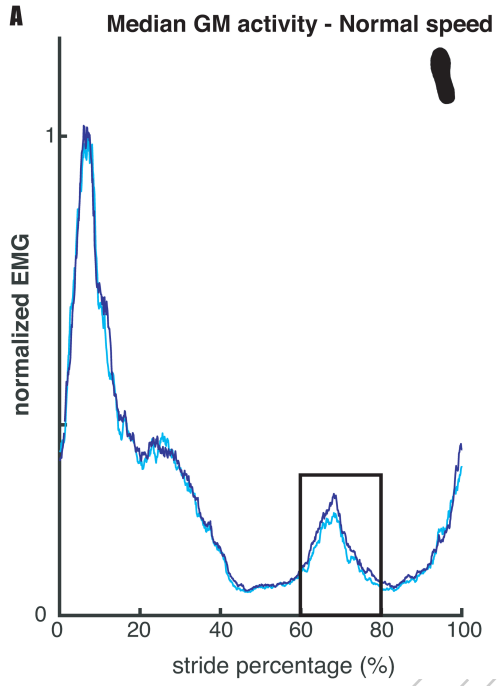
```
[EMG] = EMG_preprocessing(EMG_R,events_EMG.fs,20,50); % conform Rankin et al. (2014)
```

High pass filter → Rectification →Low pass filter

```
normalize_gait_events
```

Time normalization from heelstrike to ipsilateral heelstrike



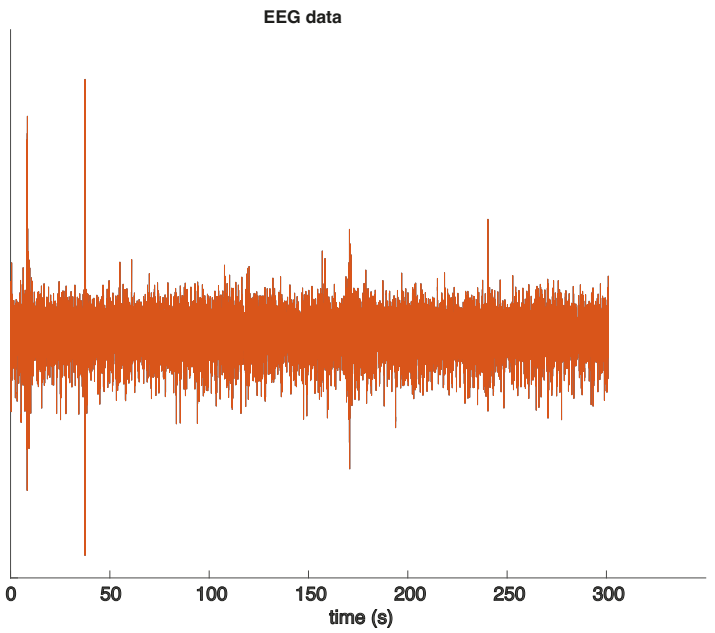



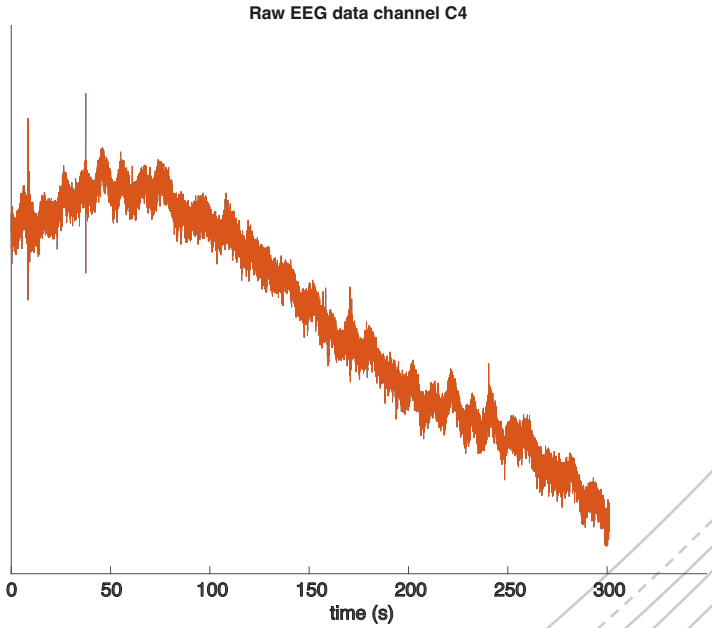

van Leeuwen et al. 2020

- Preprocessing
- Identifying artefacts ICA

## Corticomuscular coherence

Quick check: EEG

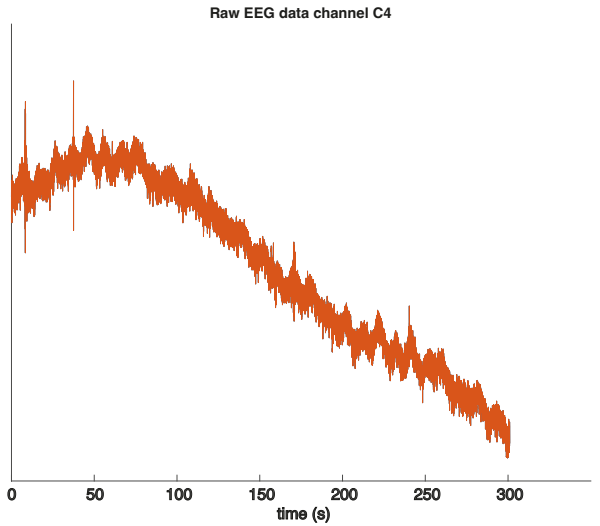


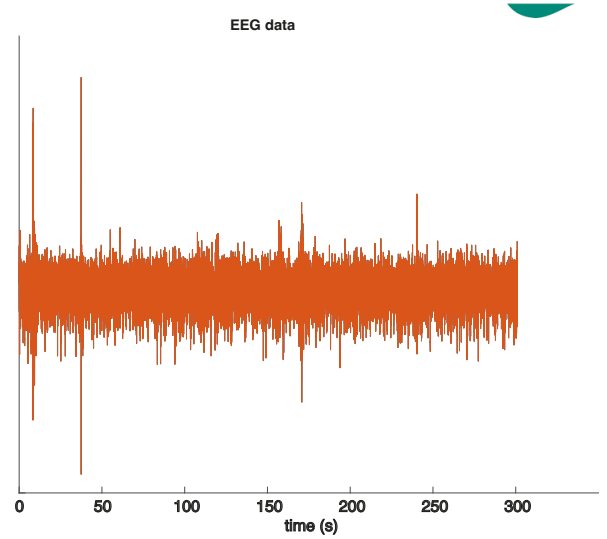



ft_preprocessing

## Corticomuscular coherence

EEG preprocessing



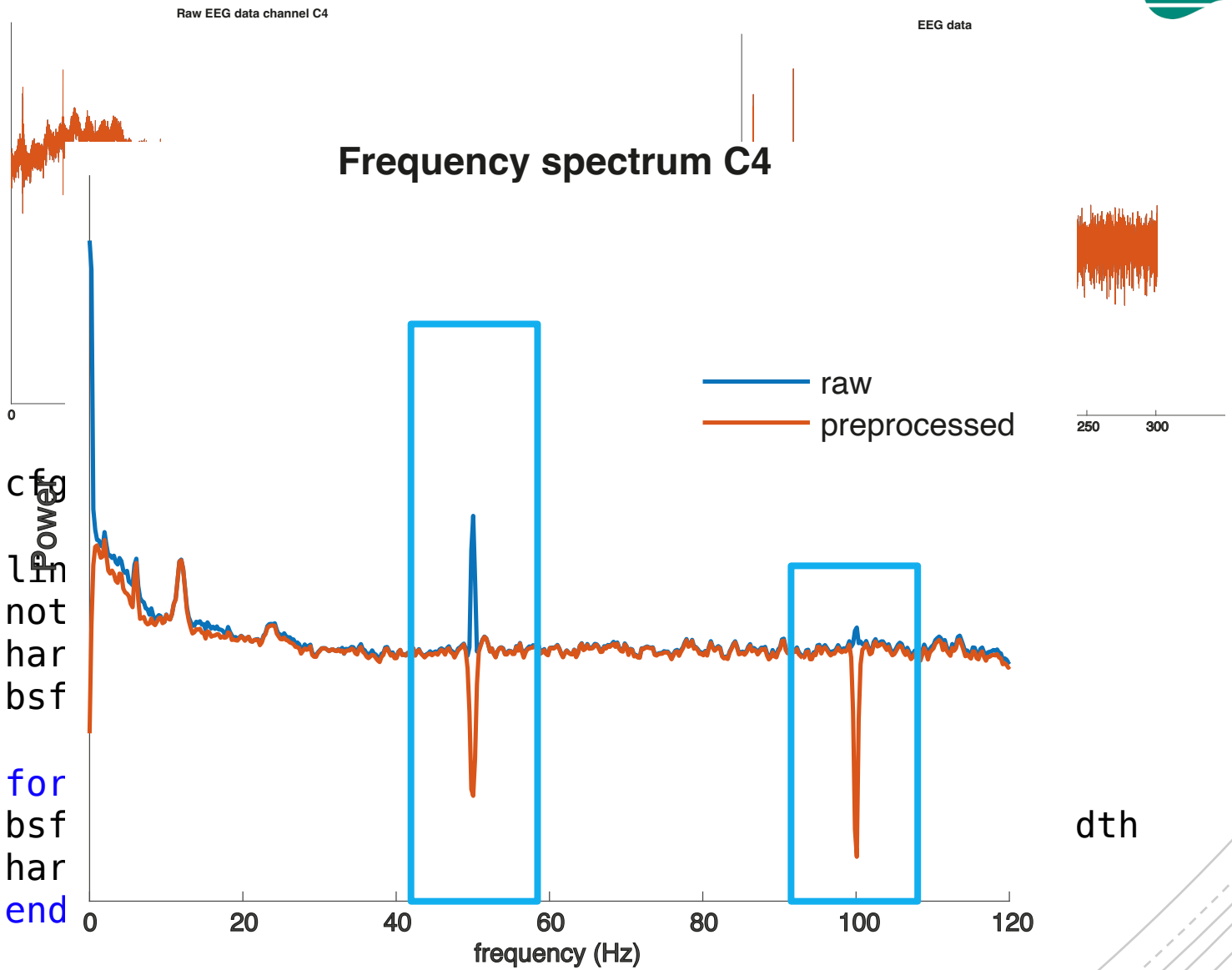




```
% Filters

cfg.bpfilter= 'yes'
cfg.bpfiltord = 2;


cfg.bpfreq = [0.5 220];

% Demean & detrend

cfg.demean = 'yes';
cfg.detrend = 'yes';
```

ft_preprocessing

## Corticomuscular coherence

# EEG preprocessing




```
cfg
lin
not
har
bsf

for
bsf
har
end
```

```
dth
```

```
cfg.bsfreq = bsfreq;
```

Corticomuscular coherence

# EEG
# Bad channels

- "Bad channels with a SD of their activation exceeding the mean SD activation plus 3 standard deviations" (Klapprott & Debener 2024)
- Single-channel variance exceeding 3 times the standard deviation, or 10 times the maximum of all channels) (Zandvoort 2019)
- Repairing bad channels based on interpolation neighboring electrodes

`ft_componentanalysis`

Corticomuscular coherence

ICA
artefact removal

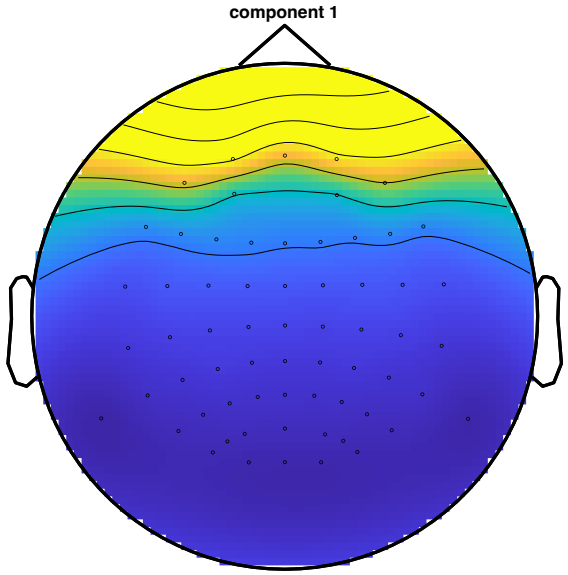
- Eye movement artefacts
- Muscle activity artefacts
- (line noise artefacts)

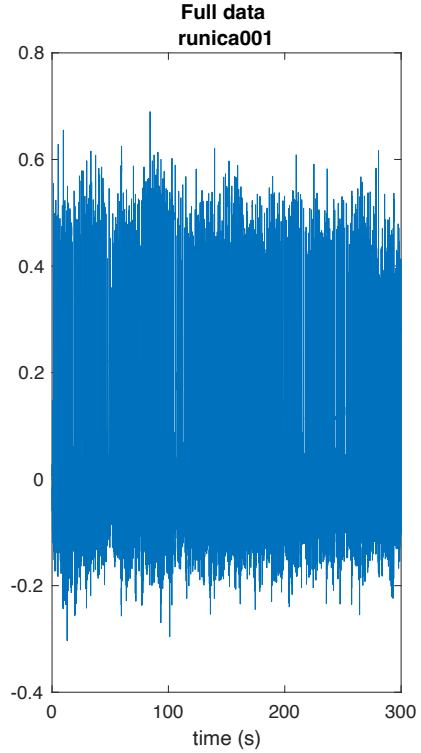
- EEG components

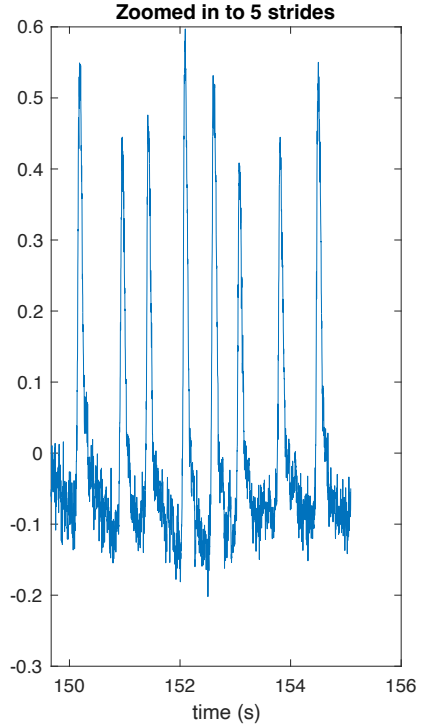
```
artifactChannels={ 'Fp1','Fp2','Fpz','AF7','AF3','AF4','AF8' };
%frontal
```

## Corticomuscular coherence

ICA
Eye artefacts



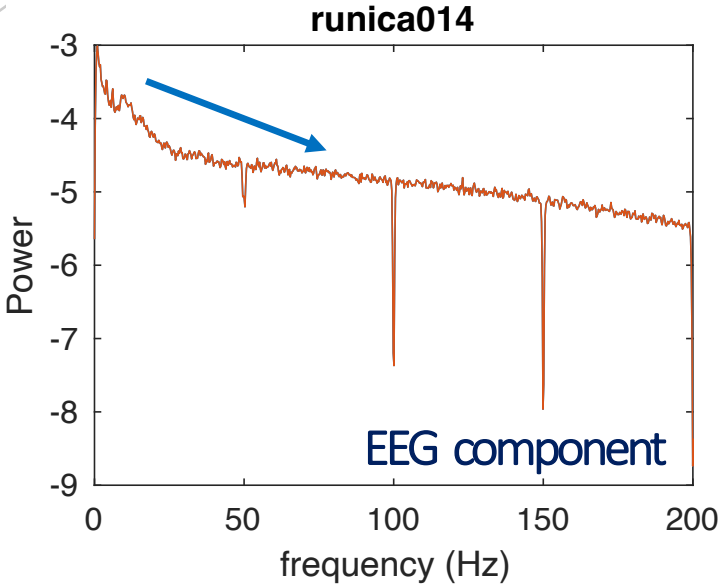



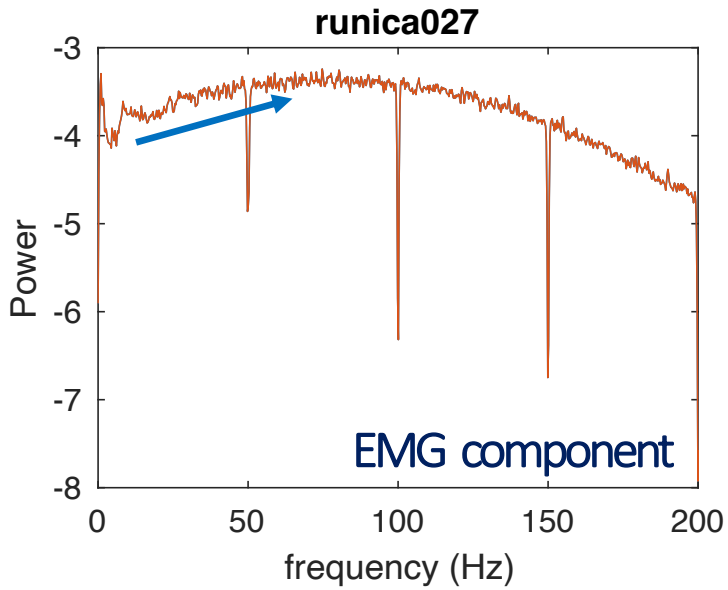




```
eye_artefact_comp1 = find(median_freq<5);
```

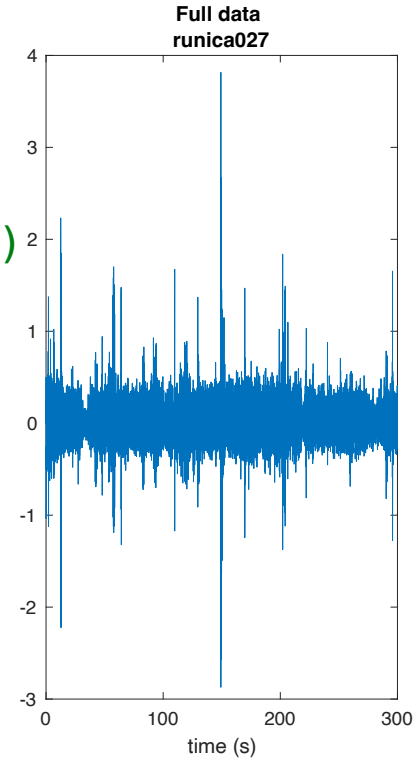
## Corticomuscular coherence

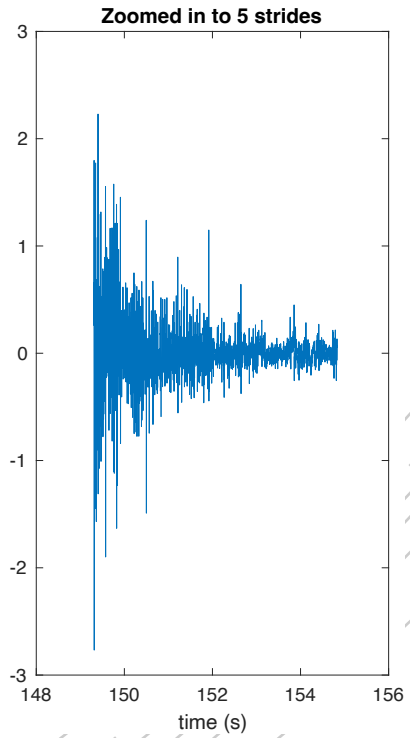
# ICA
# Muscle activity artefacts



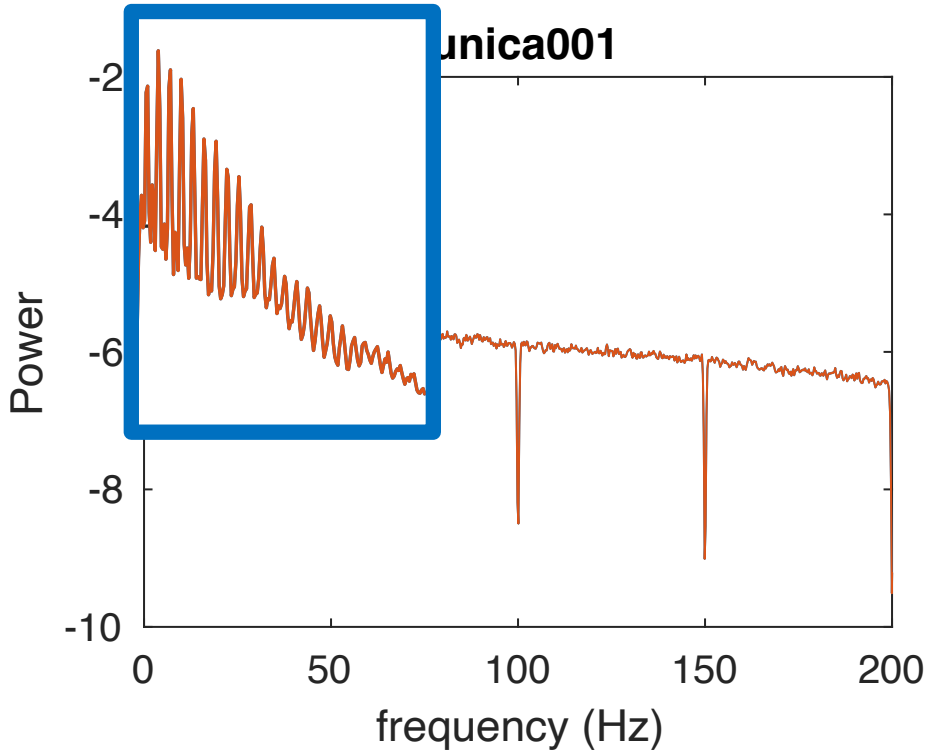

```
% Zandvoort et al (2019)
muscle_artefact_comp =
find(median_freq>60);
```



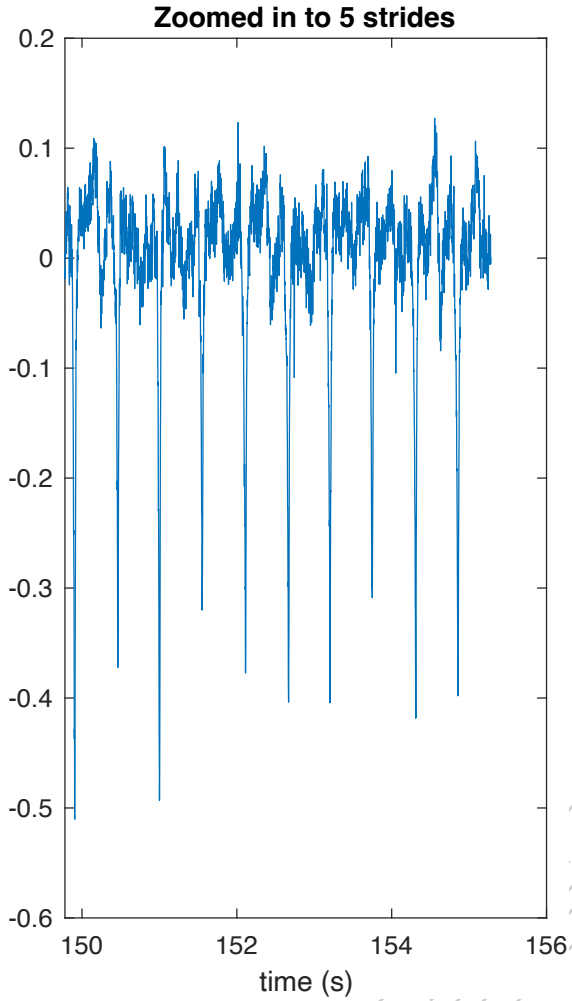



## Corticomuscular coherence

ICA
Movement
artefacts

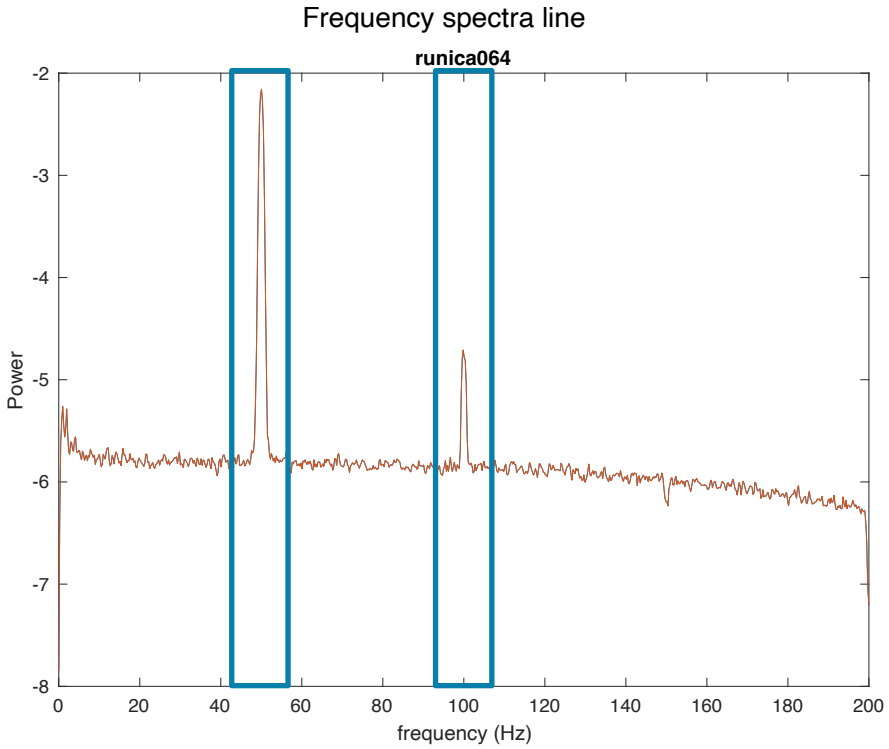


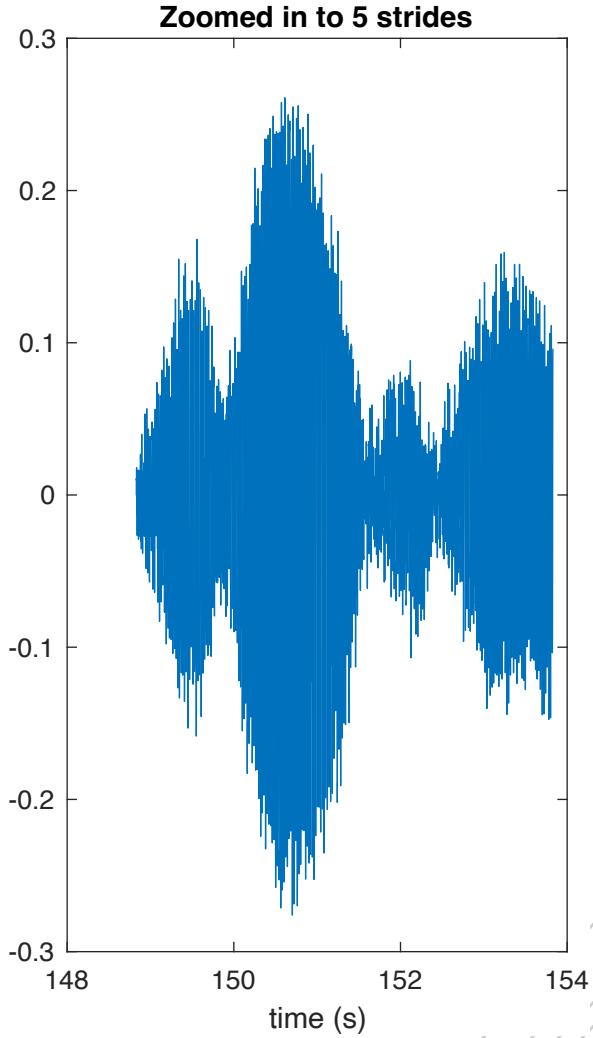

Stride frequency and harmonics



# Corticomuscular coherence

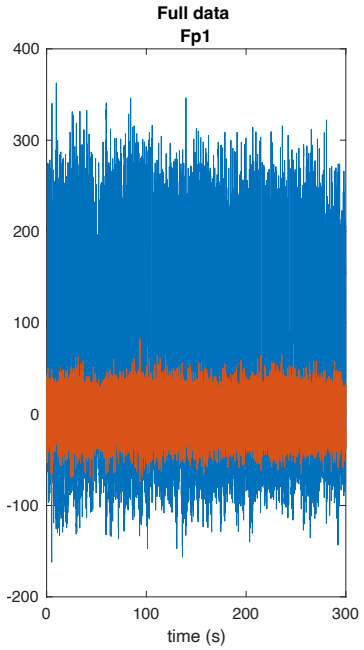
ICA
Line noise

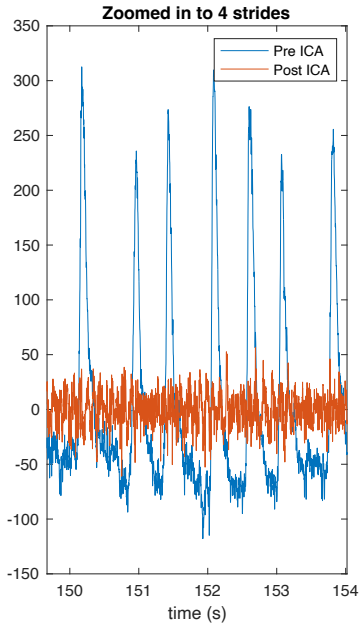


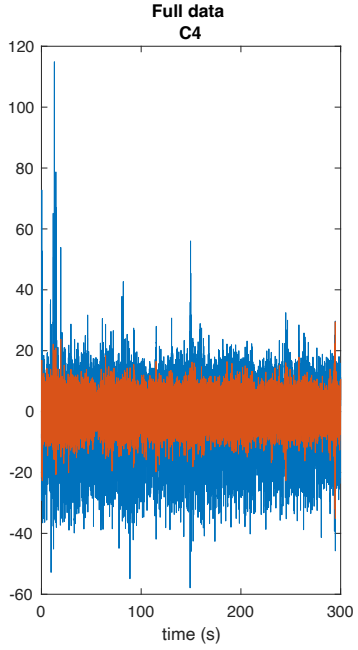



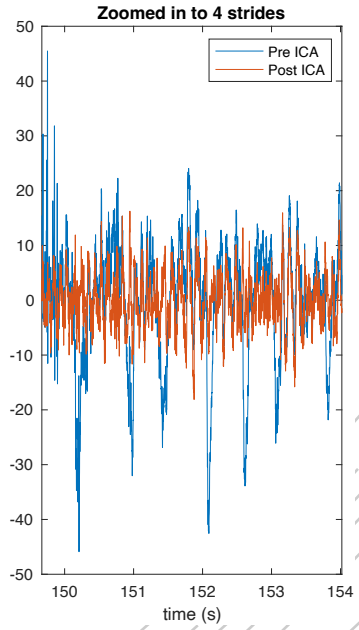
# Corticomuscular coherence

## ICA
## effect on EEG



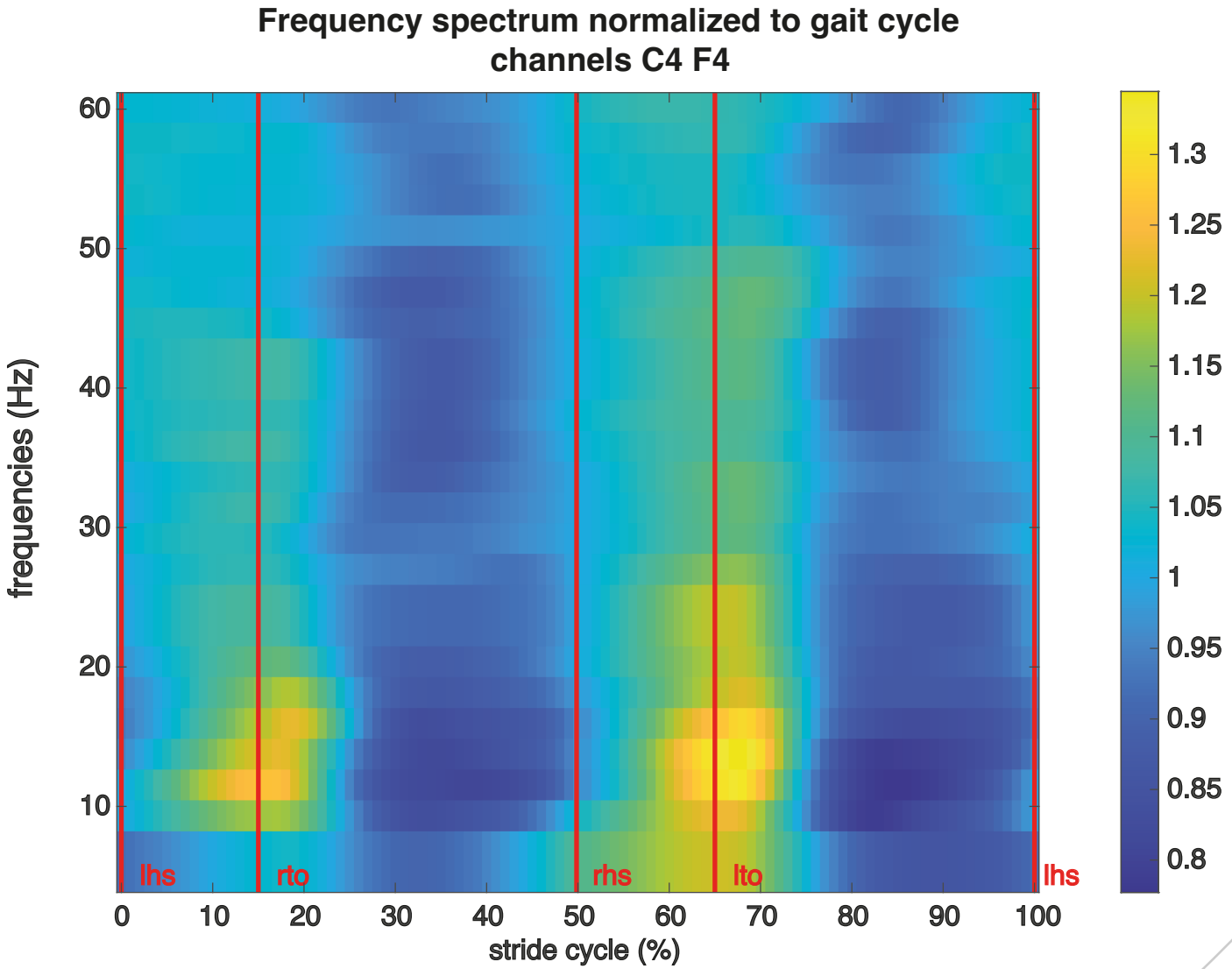



`ft_freqanalysis`

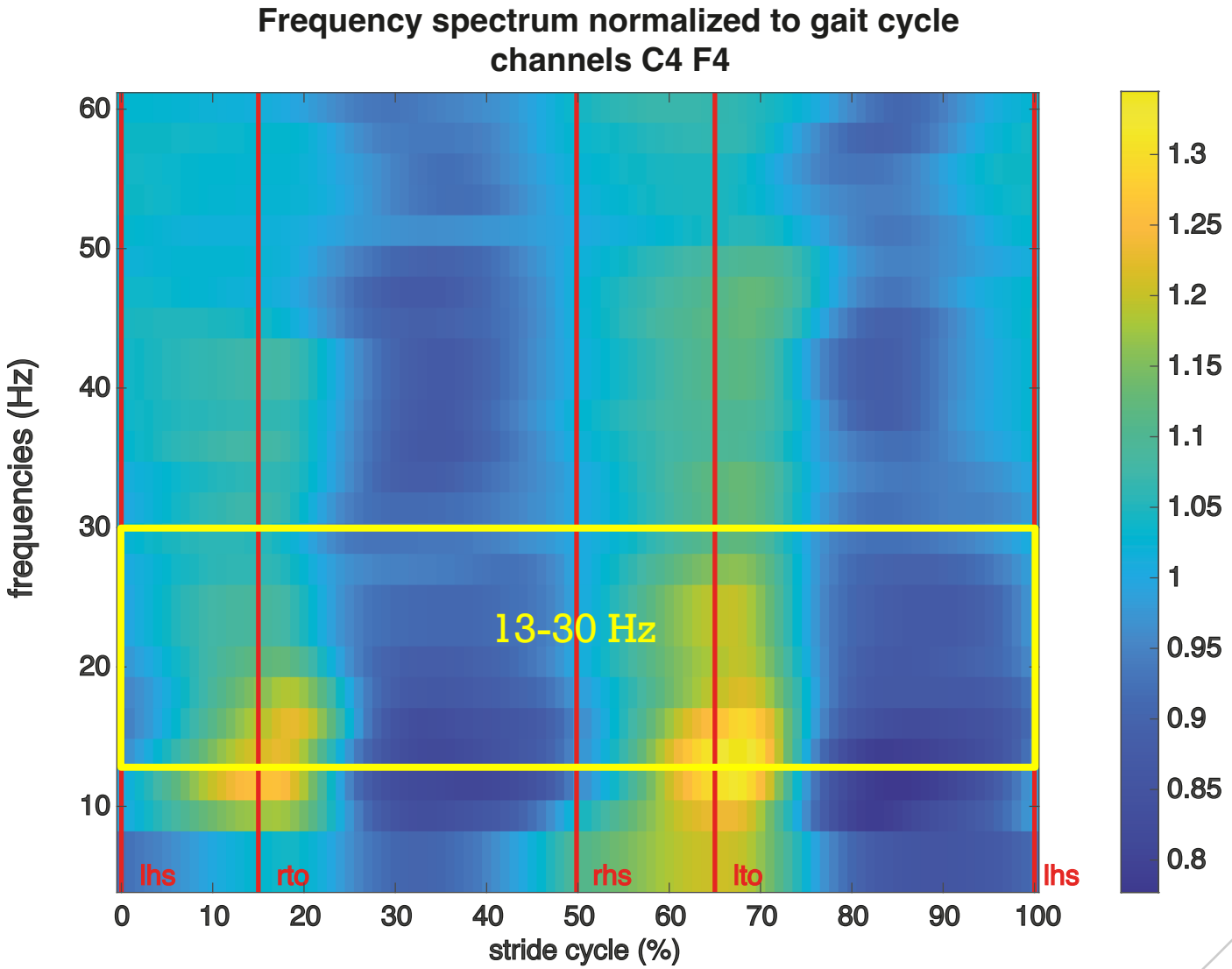
## Corticomuscular coherence

## Power spectrum



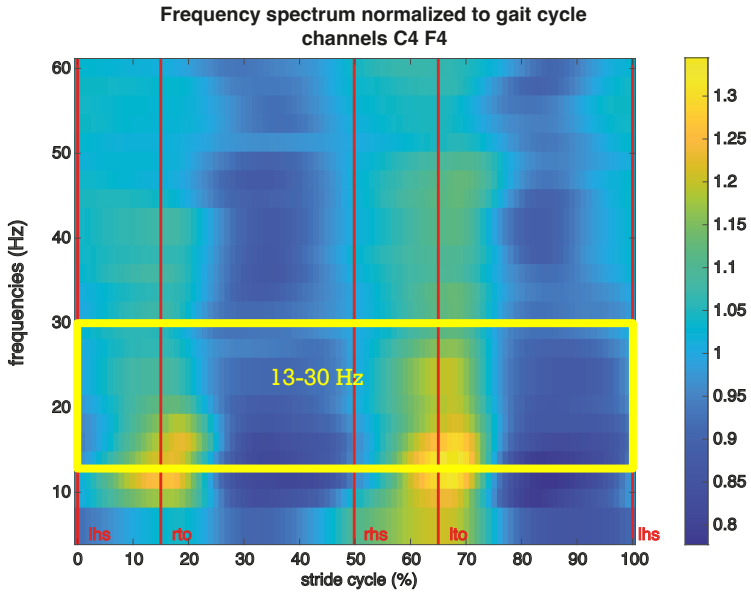
## Corticomuscular coherence

# Beta band



# Corticomuscular coherence

## Beta band



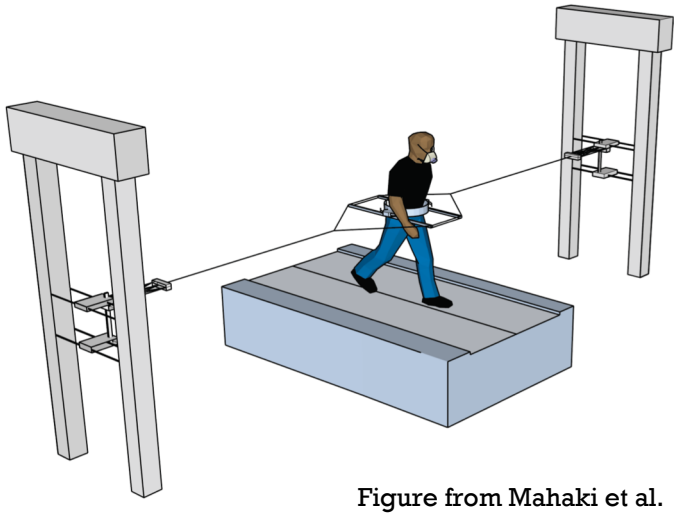
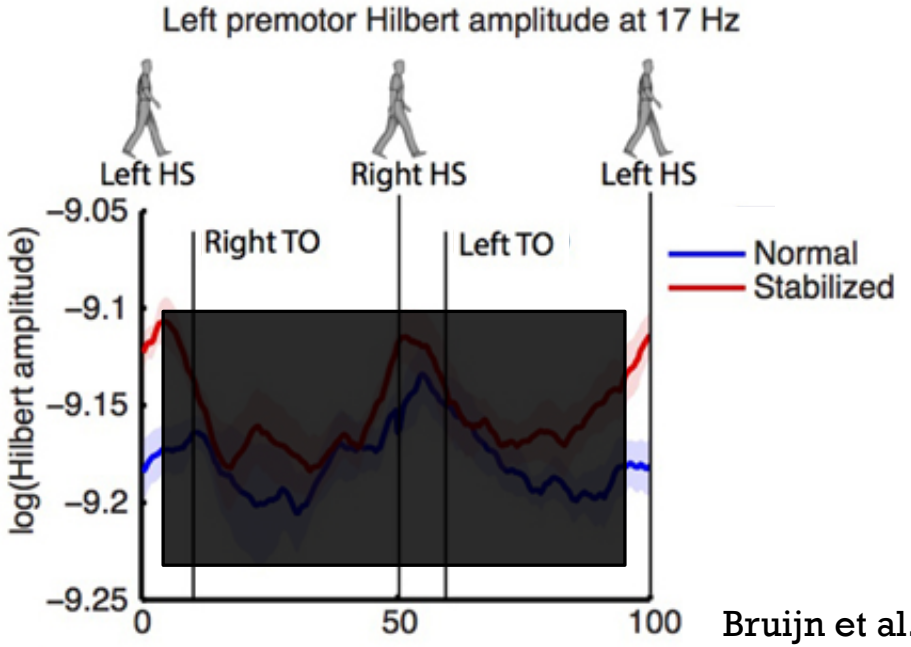
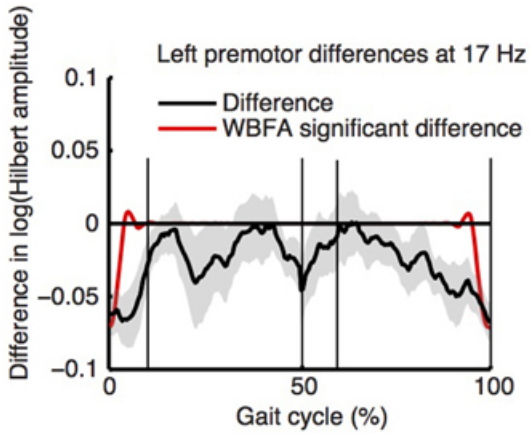



Figure from Mahaki et al.



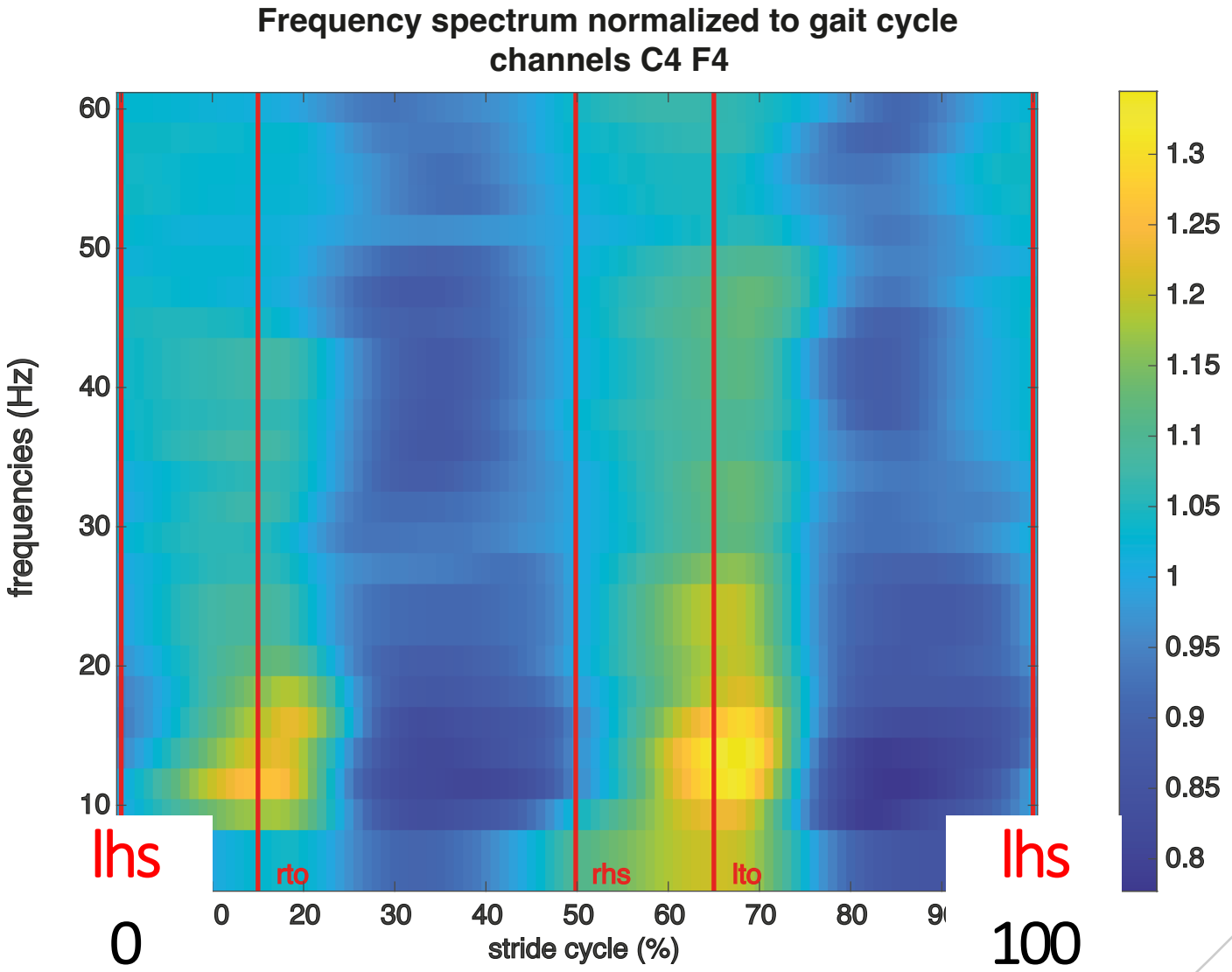




Bruijn et al. 2015

`time_normalize_power`

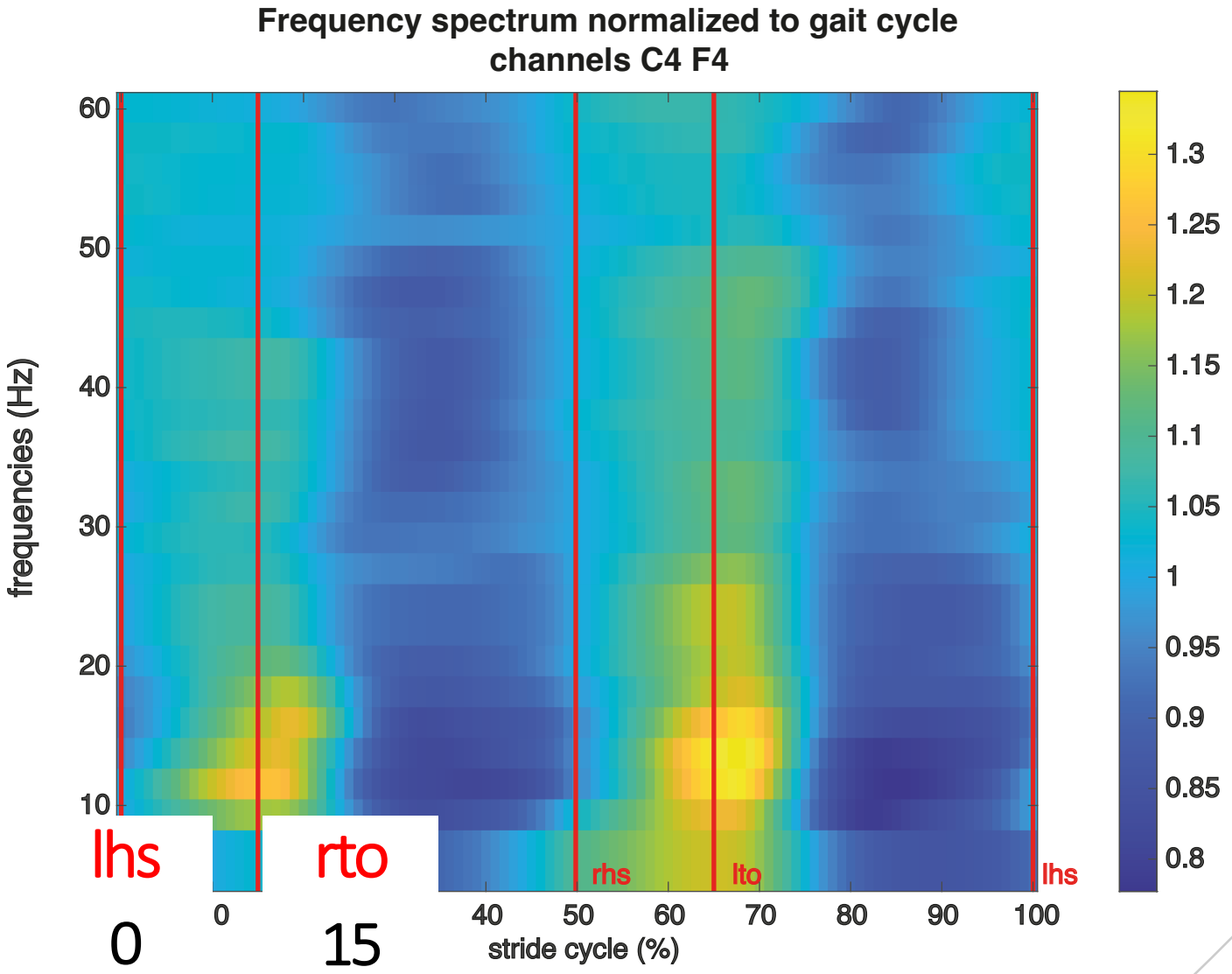
Corticomuscular coherence

Power spectrum



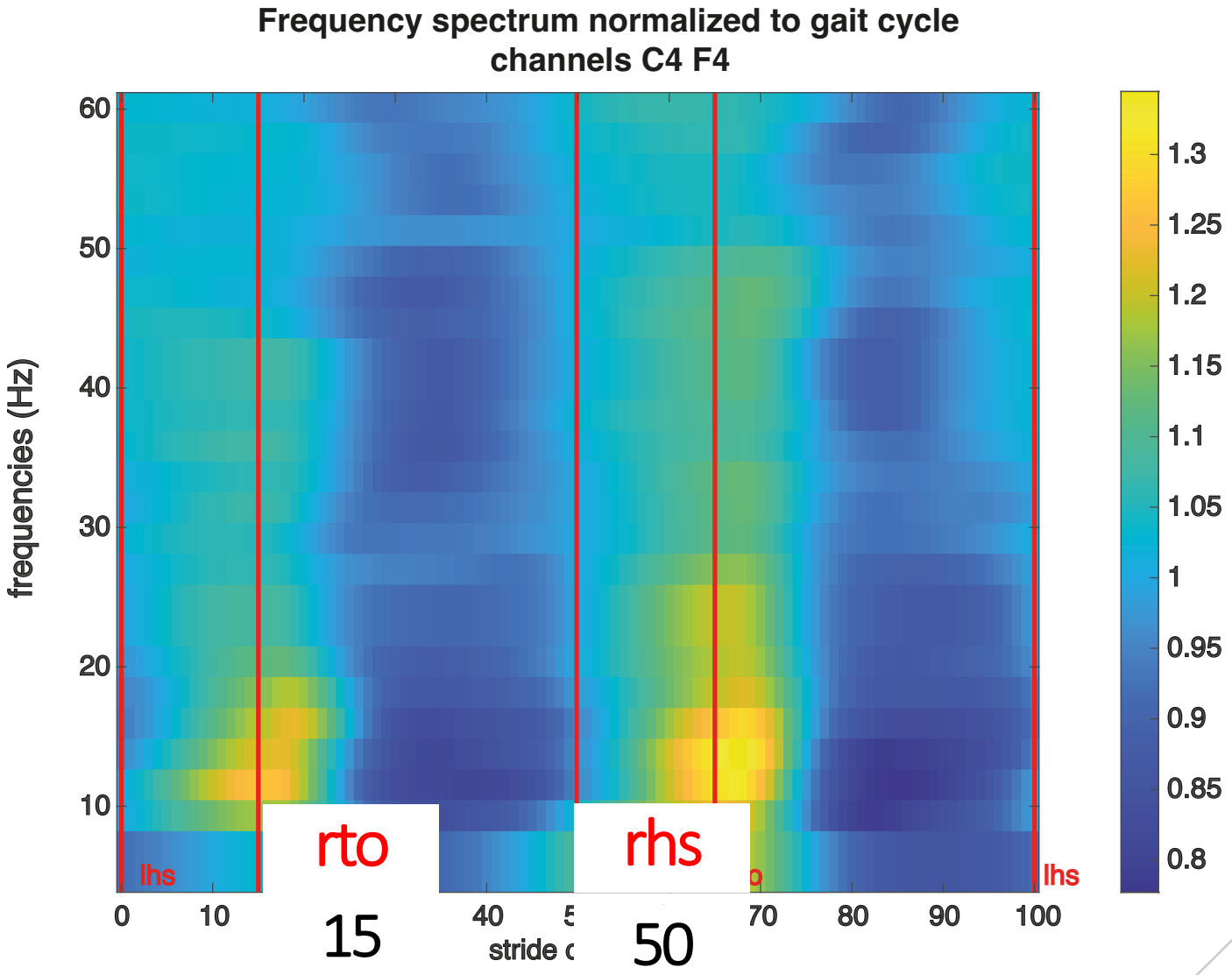
## Corticomuscular coherence

## Power spectrum



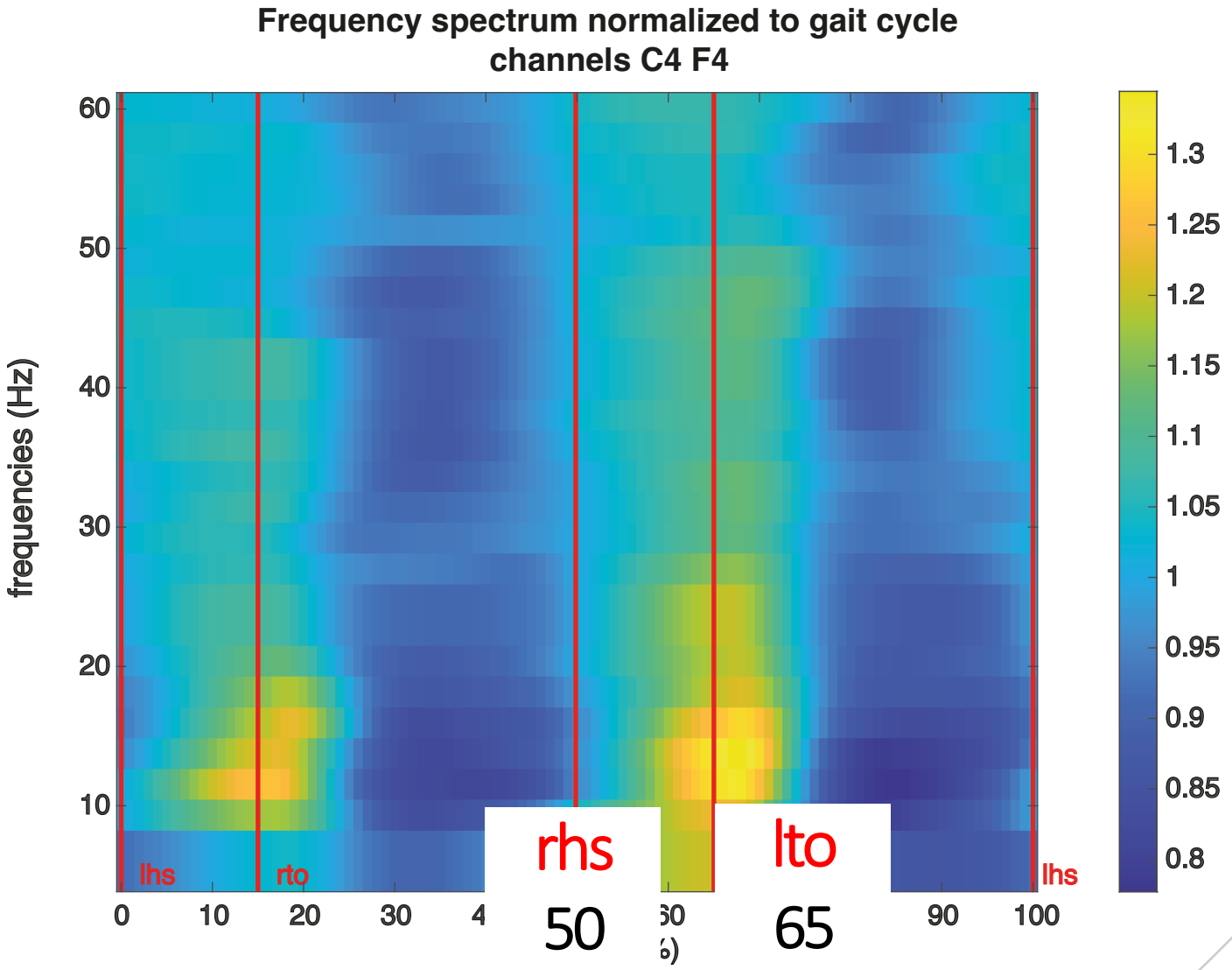
## Corticomuscular coherence

# Power spectrum



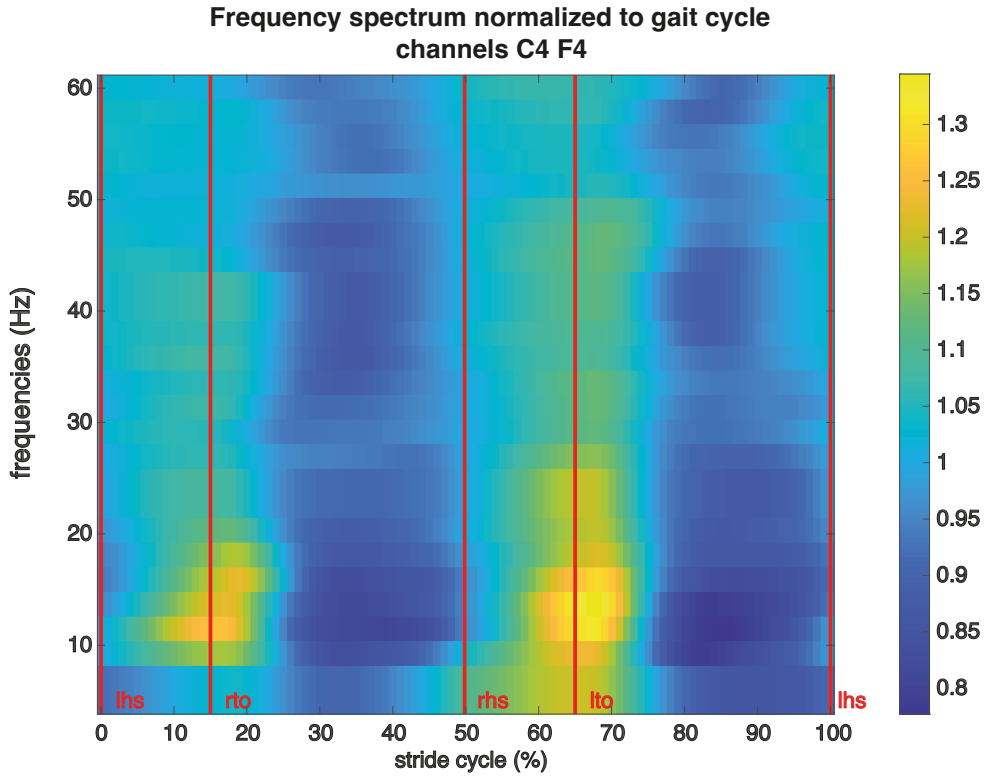
## Corticomuscular coherence

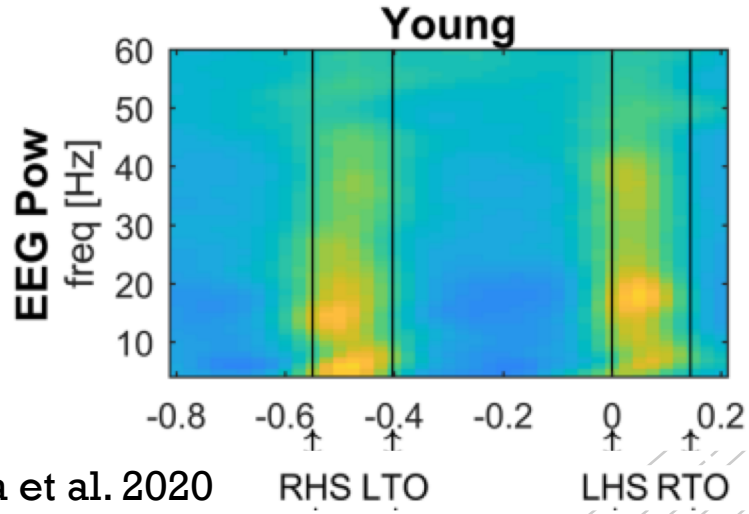
# Power spectrum



## Corticomuscular coherence

# Power spectrum

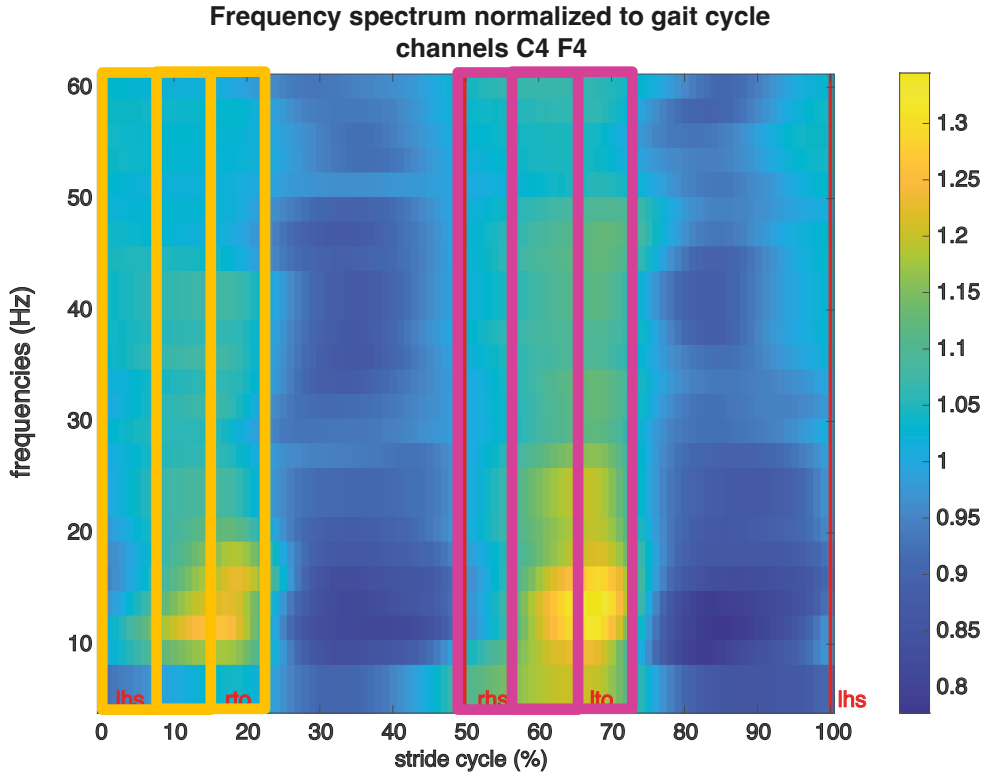


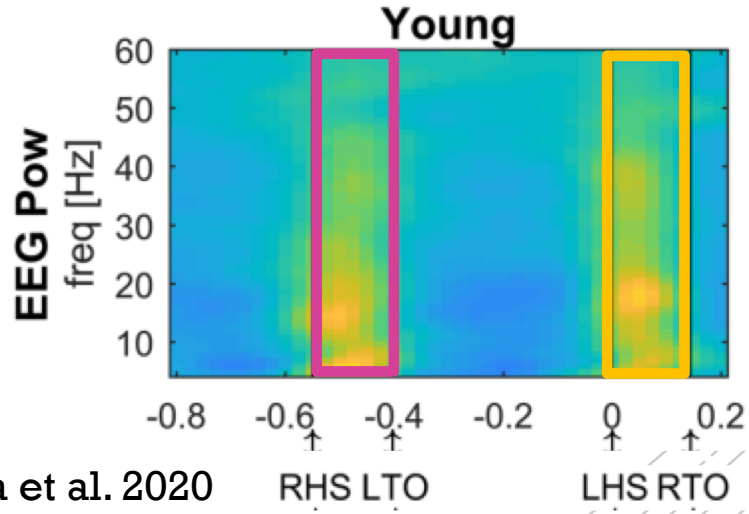




Boonstra et al. 2020

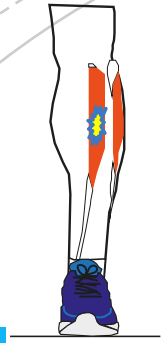
## Corticomuscular coherence

# Power spectrum



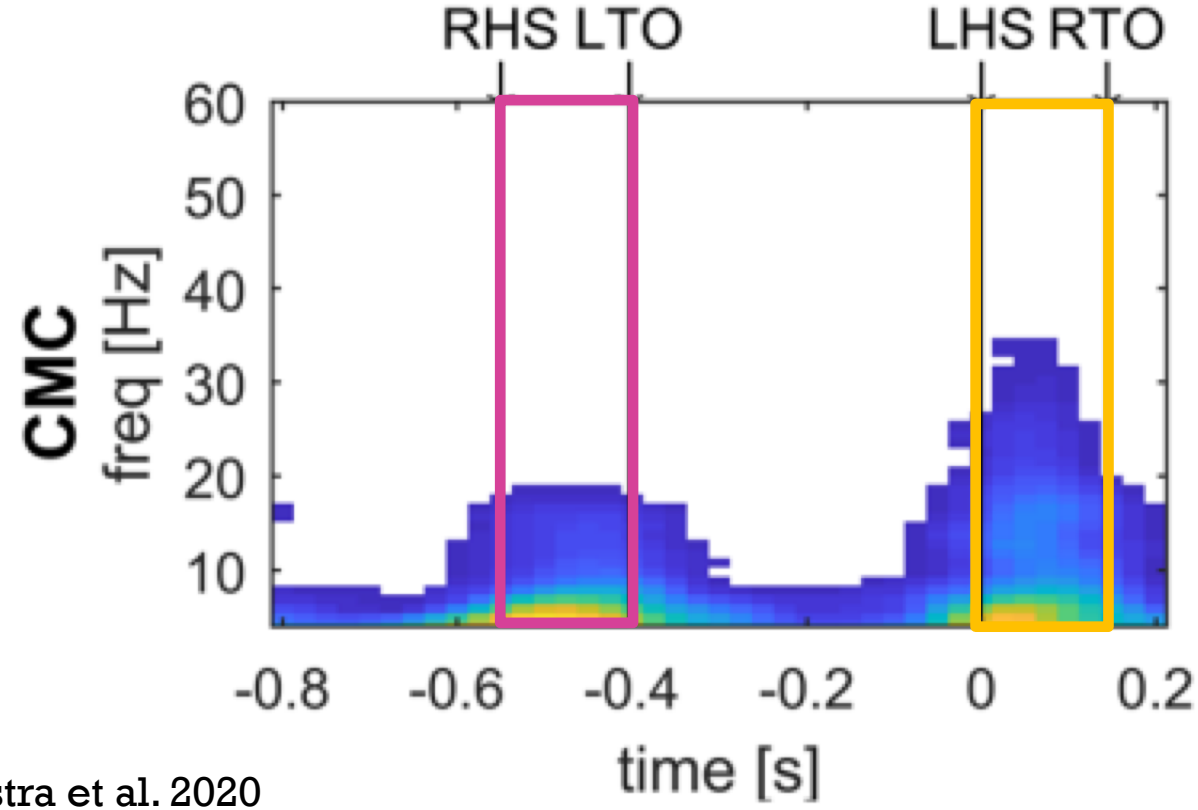

Boonstra et al. 2020



# Tibialis Anterior C4- F4

Corticomuscular coherence

Corticomuscular coherence




Boonstra et al. 2020

## Corticomuscular coherence

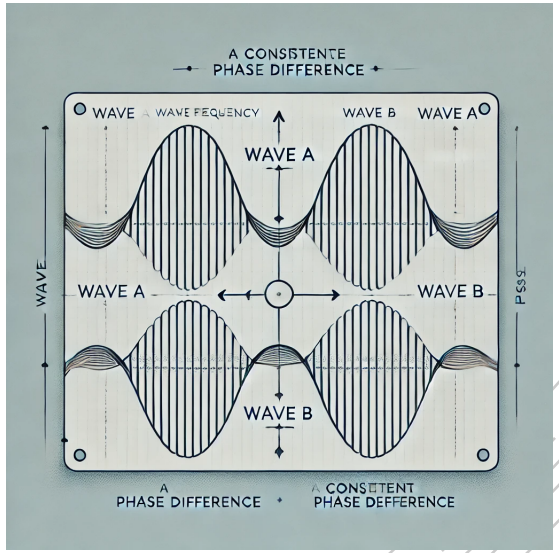
Corticomuscular coherence

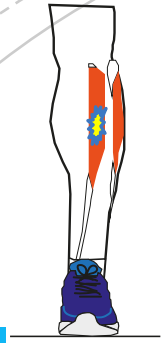
$$Coh_{S1,S2}(f) = \frac{|P_{S1,S2}(f)|^2}{|P_{S1}(f)| \times |P_{S2}(f)|}$$


**Functional connectivity**

**Cortical control over muscle activity**

**Higher coherence**
**More/better stability control?**

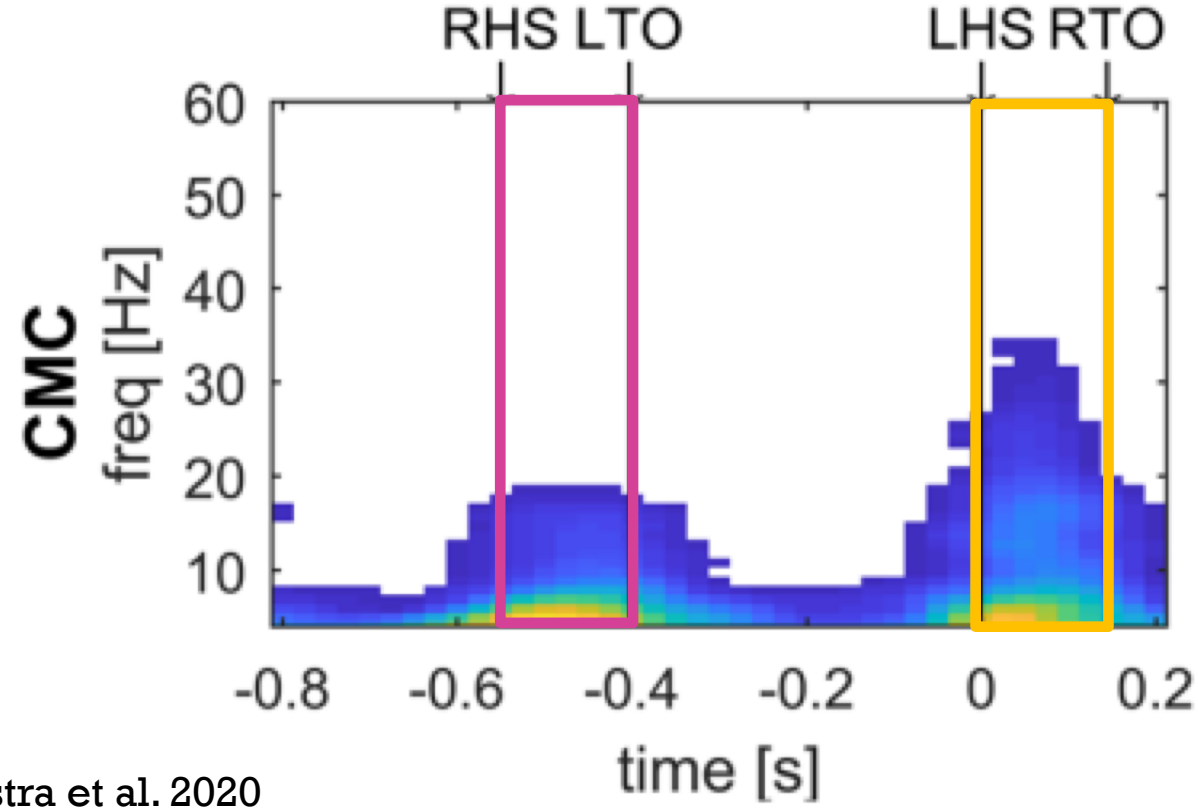




# Tibialis Anterior C4- F4

Corticomuscular coherence

Corticomuscular coherence

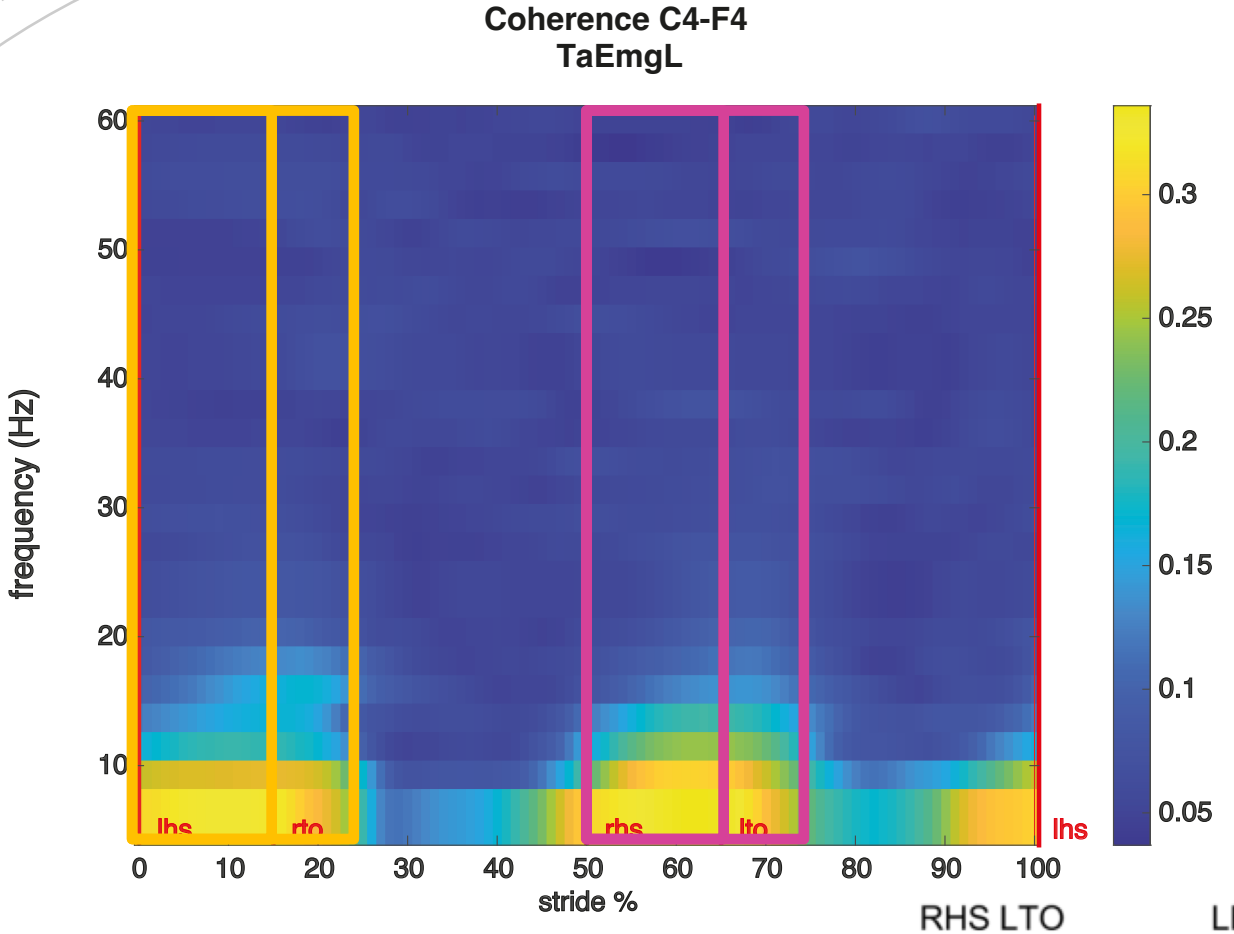


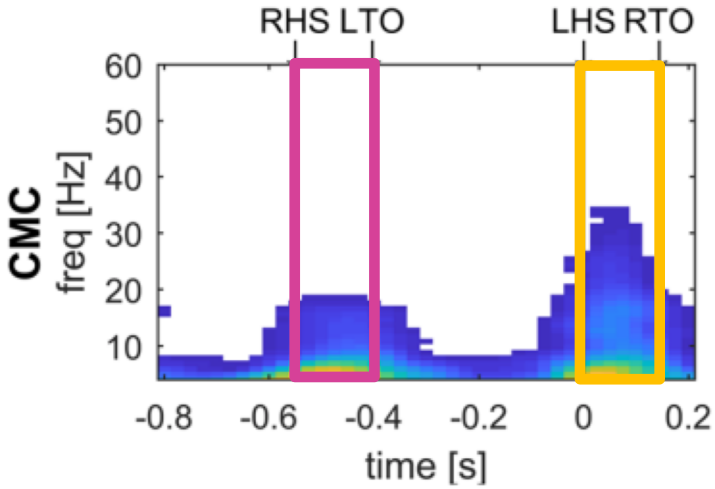

Boonstra et al. 2020

`ft_connectivityanalysis`

## Corticomuscular coherence

Corticomuscular coherence




Boonstra et al. 2020

## Corticomuscular coherence

~~Quick check: EEG~~

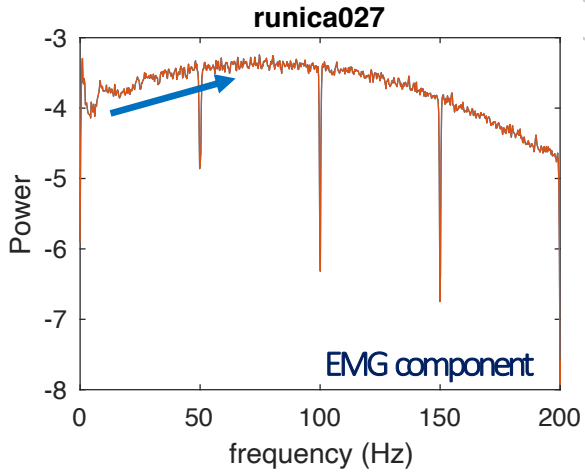
# Corticomuscular coherence

Long

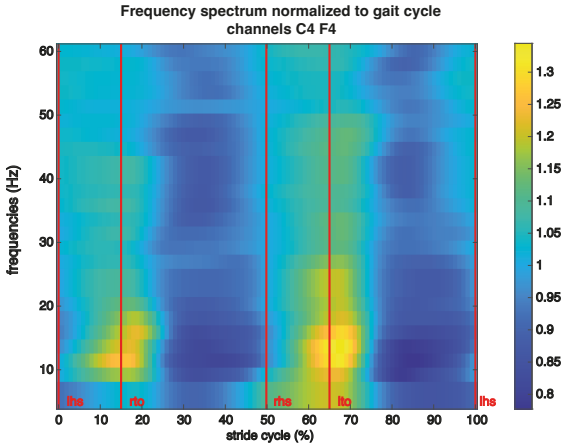
~~Quick~~ check:

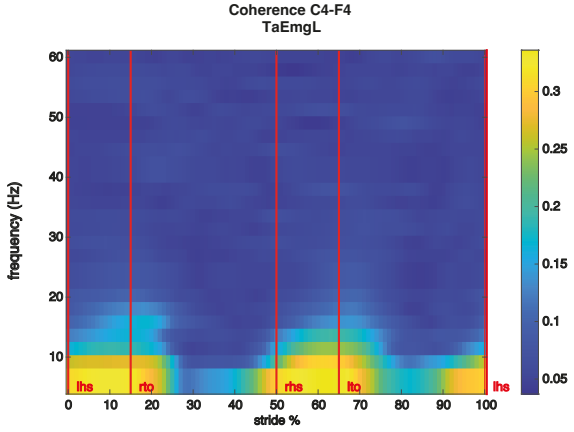
EEG

- Preprocessing
- ICA artefact removal

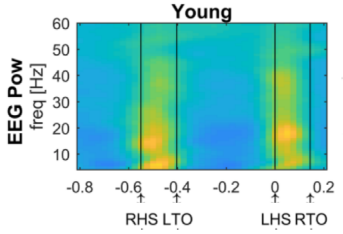


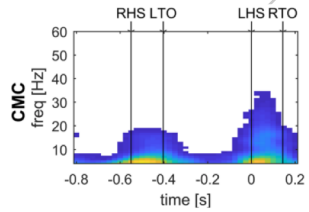

- Computing frequency power spectrum
- Computing coherence spectrum




- Comparing to the literature



Corticomuscular coherence

## Next steps

- Improving data cleaning and preprocessing
- Computing results for muscles of interest
- Source localization
- Statistics

Corticomuscular coherence

# Next steps

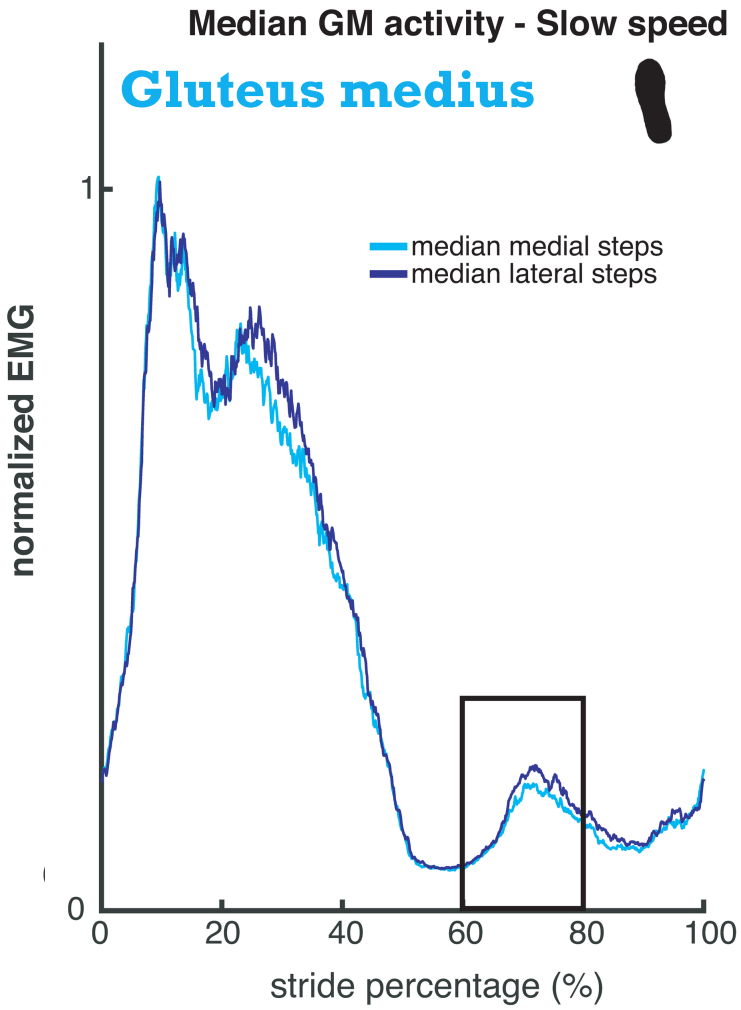
- Improving data cleaning and preprocessing
- Computing results for muscles of interest
- Source localization
- Statistics

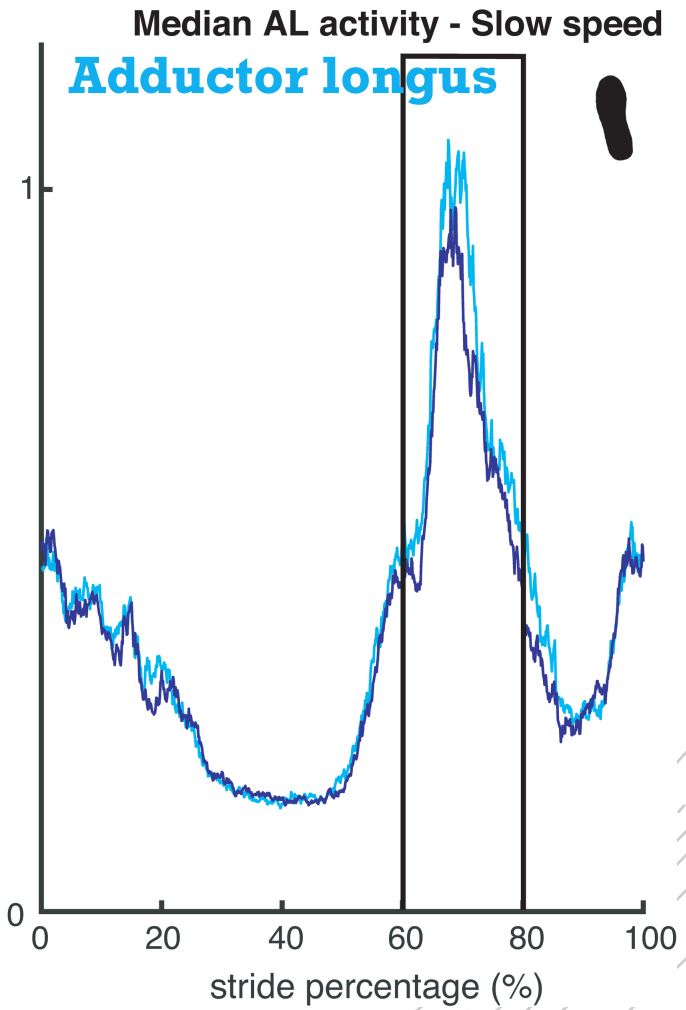
ML

$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

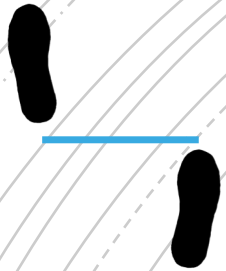
## Corticomuscular coherence

## Muscles of interest



AP

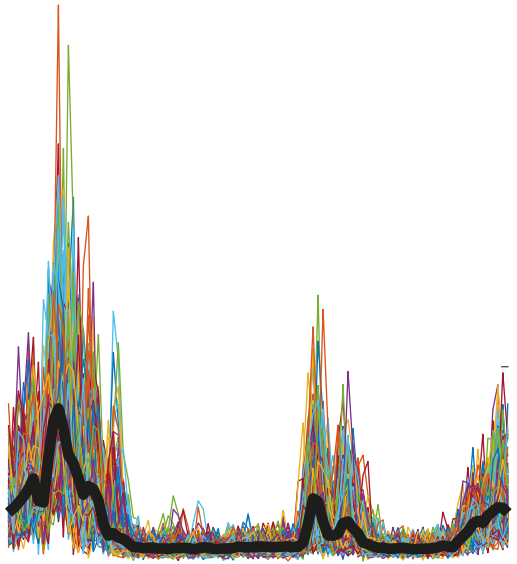


$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

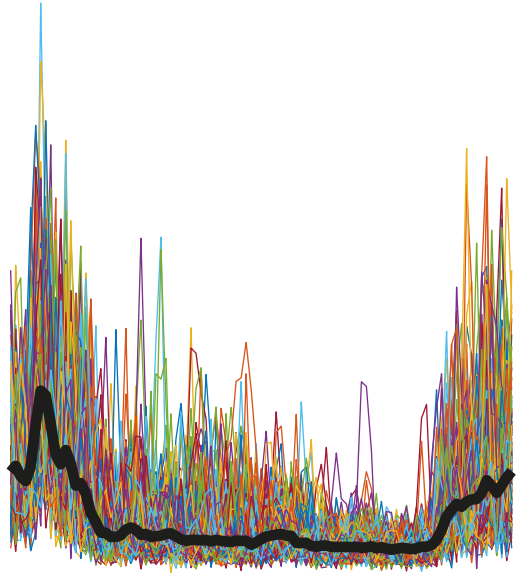
## Corticomuscular coherence

## Muscles of interest

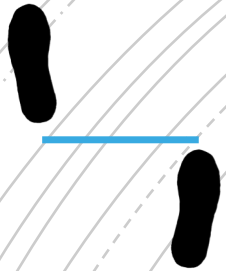
### Rectus femoris



### Biceps femoris



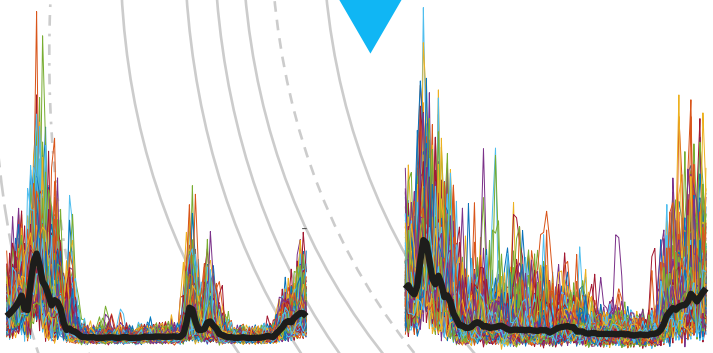
AP



$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

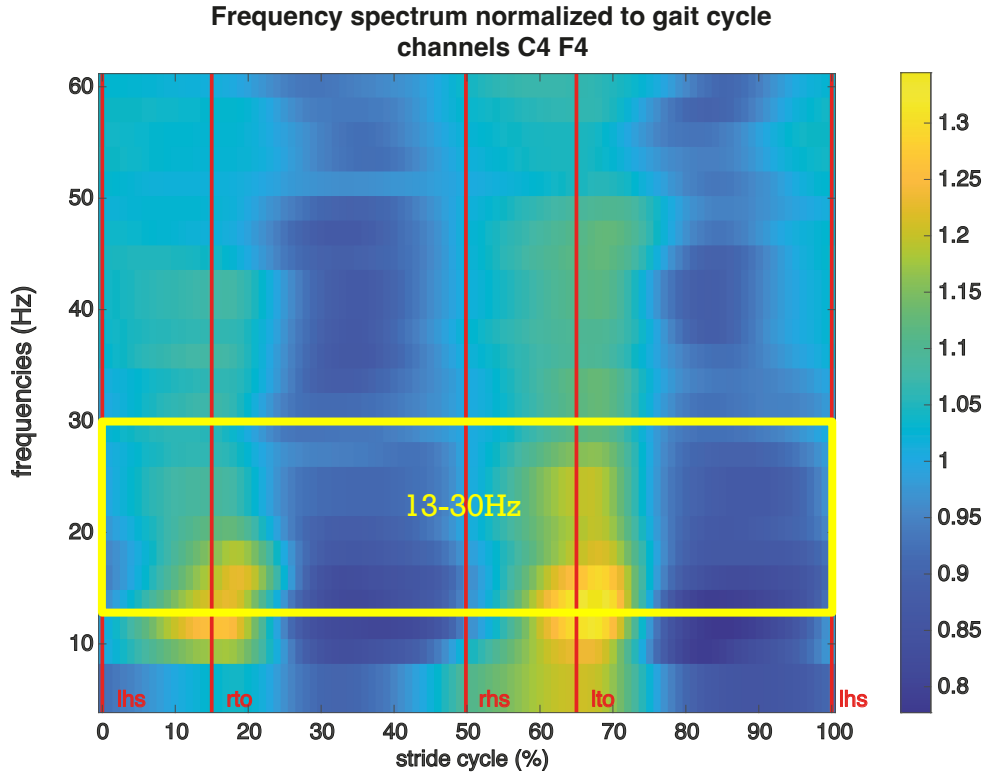
## Corticomuscular coherence

## Muscles of interest

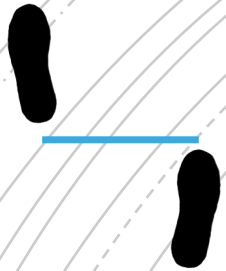


### Rectus femoris

### Biceps femoris



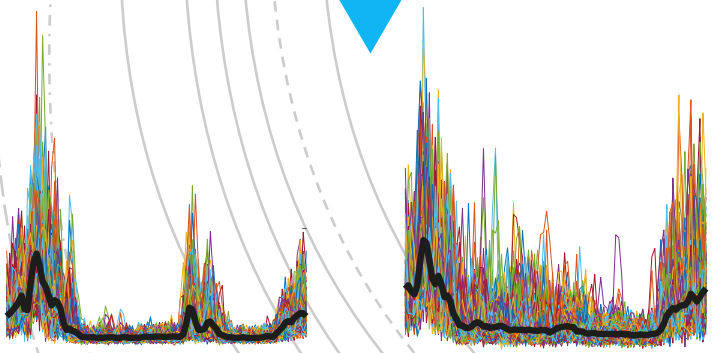
AP



$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$

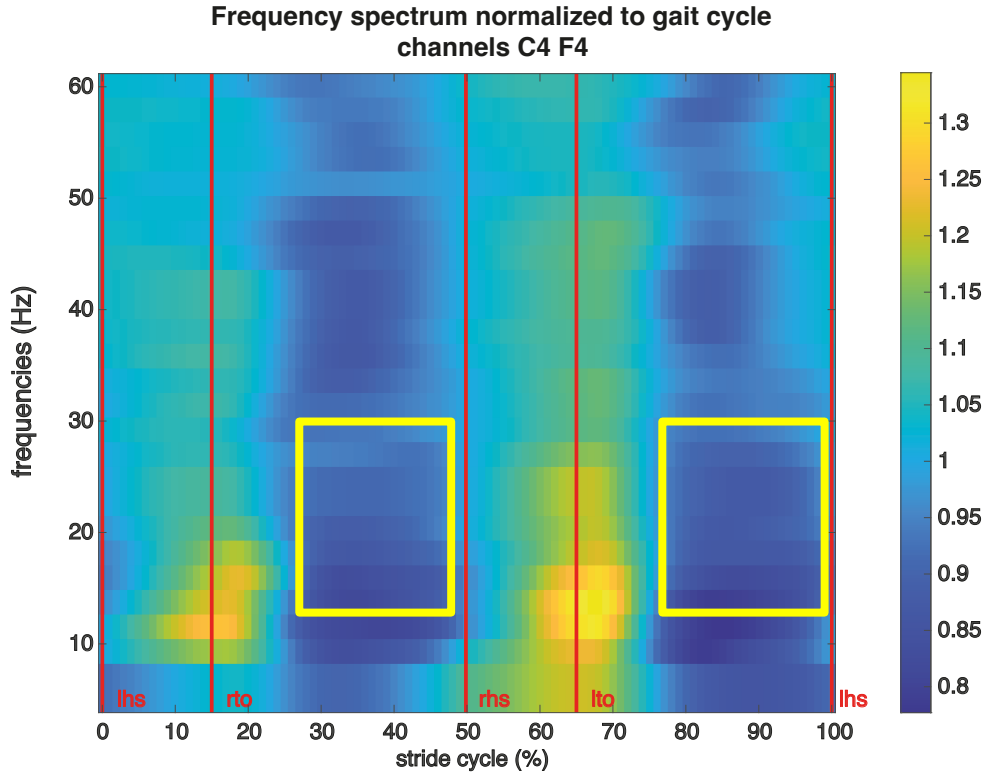
## Corticomuscular coherence

## Muscles of interest

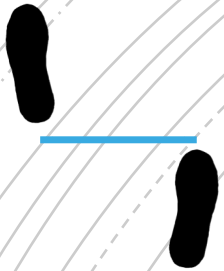


### Rectus femoris

### Biceps femoris



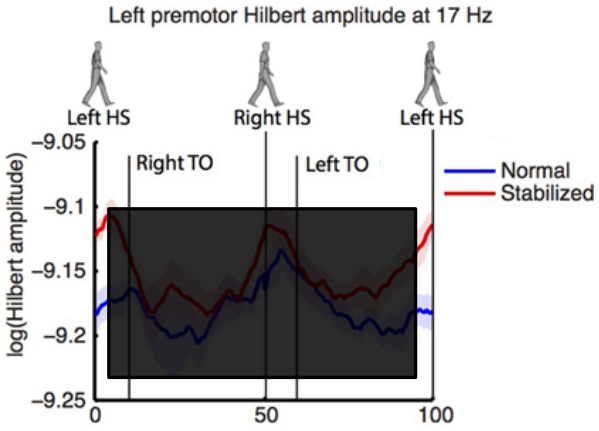
AP

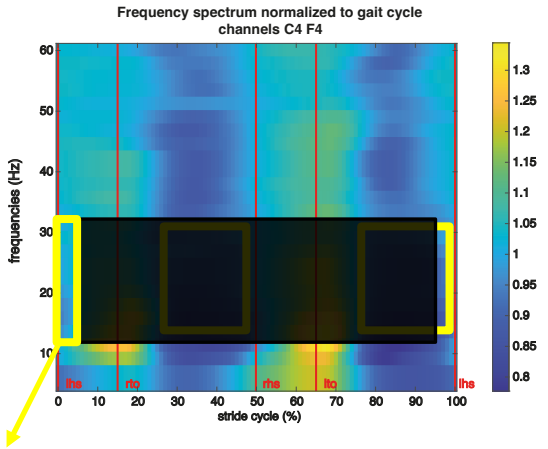


## Corticomuscular coherence

## Muscles of interest

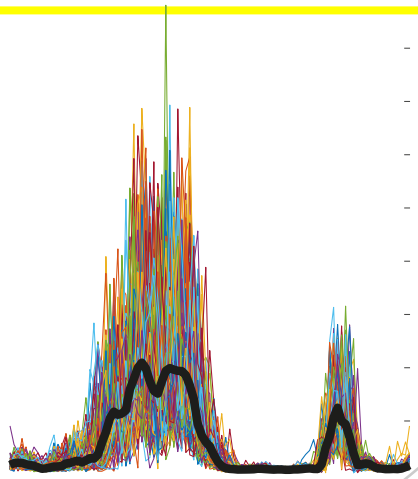
$$FP = \beta_{CoM\_pos}(i) * CoM_{pos}(i) + \beta_{CoM\_vel}(i) * CoM_{vel}(i) + \varepsilon$$



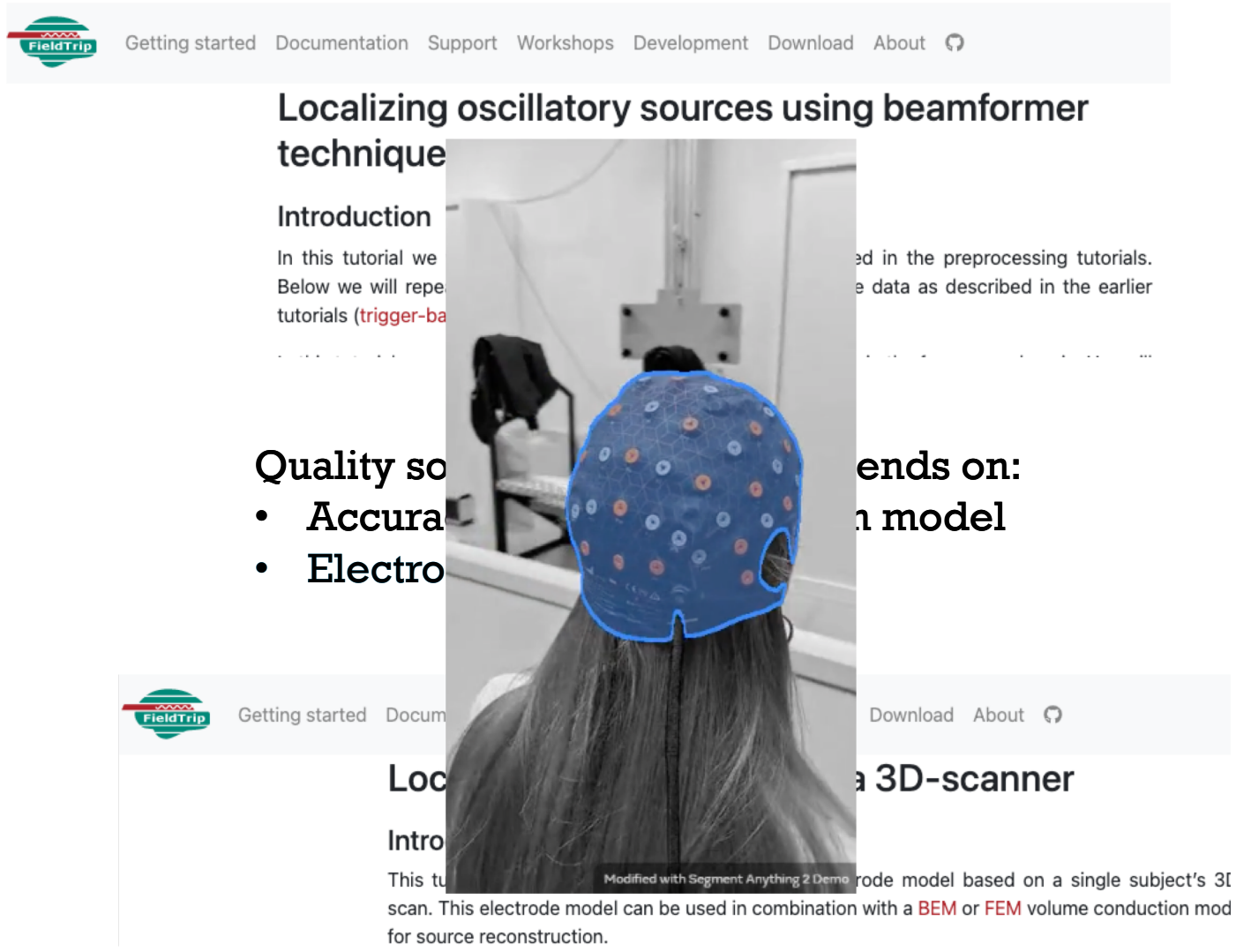

**Double stance phase → preparing pushoff**

**Gastrocnemius**



## Corticomuscular coherence

# Source localization



JPND research
EU Joint Programme – Neurodegenerative Disease Research

Amsterdam team: prof. dr. Jaap van Dieën, dr. Sjoerd Bruijn, dr. Bernadette van Wijk
dr. Moira van Leeuwen

# Foot Placement Control & Corticomuscular Coherence

## Concepts &

## Analysis Pipeline

Amsterdam Movement Sciences
VU VRIJE UNIVERSITEIT AMSTERDAM
Amsterdam UMC University Medical Centers