Table of Contents

Variables

```
field=[field];  % Extracellualr field
Vm=[Vm];        % Whole Cell
tb=[tb];        % time base
```

```
Undefined function or variable 'field'.

Error in master_phase_locking_analysis (line 3)
field=[field];  % Extracellualr field
```

Bandpass filter

```
cutout=[];
filtered=[];
filtered.params.Fs=1/mean(diff(tb(1:50)));%20000
filtered.params.Nyquist=filtered.params.Fs/2;
% bandpass filter to separate gamma
filtered.params.bp.wn_low= [1 10];
filtered.params.bp.wn_low=filtered.params.bp.wn_low./
filtered.params.Nyquist;

[filtered.params.bp.B_low,  filtered.params.bp.A_low]   =
 butter(1,filtered.params.bp.wn_low,'bandpass');

%
 fvtool(filtered.params.bp.B_low,filtered.params.bp.A_low,'Fs',filtered.params.Fs)

for trial_id=1:size(field,2)
        clear temp;
        temp=field(:,trial_id);
        filtered.field(:,trial_id)      =
 filter(filtered.params.bp.B_low,filtered.params.bp.A_low,temp);
end
```

Perievent histogram

```
cutout=[];
cutout.reject_boundaries=[1 numel(field)];
```

```
cutout.spike_pos_Vm=zeros(500,size(Vm,2));
 cutout.spike_pos_Vm(cutout.spike_pos_Vm==0)=NaN;
cutout.spike_times_Vm=zeros(500,size(Vm,2));
 cutout.spike_times_Vm(cutout.spike_times_Vm==0)=NaN;
temp1=[]; temp2=[];
% figure ; hold on
for trial_id=1:size(Vm,2) % find spike times
    [temp1, temp2]=
 findpeaks(Vm(:,trial_id),'MINPEAKHEIGHT',-10); %amp, time
    temp1(temp2<cutout.reject_boundaries(1) |
 temp2>cutout.reject_boundaries(2))=[];
    temp2(temp2<cutout.reject_boundaries(1) |
 temp2>cutout.reject_boundaries(2))=[];
    cutout.spike_pos_Vm(1:numel(temp1),trial_id)=temp1;
    cutout.spike_times_Vm(1:numel(temp2),trial_id)=temp2;
%     plot(tb,Vm(:,trial_id));
%
 scatter(tb(cutout.spike_times_Vm(:,trial_id)),cutout.spike_pos_Vm(:,trial_id)))
end
cutout.spike_pos_Vm(cutout.spike_pos_Vm==0)=NaN;
 cutout.spike_times_Vm(cutout.spike_times_Vm==0)=NaN;

% cutout field around spike times (for Vm data)
cutout.window_size=500;
cutout.Vm_extract=[];cutout.field_extract=[];
for trial_id=1:size(Vm,2)
   time_temp= cutout.spike_times_Vm(:,trial_id);
 time_temp(isnan(time_temp))=[];
   amp_temp = cutout.spike_pos_Vm(:,trial_id);
 amp_temp(isnan(time_temp))=[];
    for spike_id =1:numel(time_temp)
        cutout.Vm_extract{trial_id}
(:,spike_id)=Vm(time_temp(spike_id)-
cutout.window_size:time_temp(spike_id)+cutout.window_size, trial_id);

        %cutout.field_extract{trial_id}
(:,spike_id)=field(time_temp(spike_id)-
cutout.window_size:time_temp(spike_id)+cutout.window_size, trial_id);
        cutout.field_extract{trial_id}
(:,spike_id)=filtered.field(time_temp(spike_id)-
cutout.window_size:time_temp(spike_id)+cutout.window_size, trial_id);
    end
end
cutout.Vm_extract=cell2mat(cutout.Vm_extract);
 cutout.field_extract=cell2mat(cutout.field_extract);

% run hilbert transform
for spike_id =1:size(cutout.field_extract,2)

 cutout.hilbert_field(:,spike_id)=(angle(hilbert(cutout.field_extract(:,spike_id))
% cutout.hilbert =
 atan2(imag(hilbert(cutout.field_extract(:,1))),real(hilbert(cutout.field_extract(
pi;
end
```

```
cutout.hilbert_phase=cutout.hilbert_field(501,:);
phase.mean=nanmean(cutout.hilbert_phase);
phase.sem=nansem(cutout.hilbert_phase);

cutout.hilbert_field_mean=nanmean(cutout.hilbert_field,2);
cutout.hilbert_field_sem=nansem(cutout.hilbert_field,2)
```

Plot spike phase

```
figure ; hold on
plot(tb(1:5000),1000*field(1:5000),'k')
plot(tb(1:5000),Vm(1:5000),'b')

tb_temp=((1:1001)/filtered.params.Fs)*1000;
max(tb_temp)
% axis_limits=([0 max(tb_temp) -Inf Inf]);
axis_limits=([0 Inf -Inf Inf]);
temp=[];
for idx=1:size(cutout.field_extract,2)
    temp(:,idx)=cutout.field_extract(:,idx)-
mean(cutout.field_extract(:,idx));
end

cutout.field_mean=nanmean(temp,2);
cutout.field_sem=nansem(temp,2);
figure; hold on
subplot(4,1,1); hold on %Vm whole- cell
    plot(tb_temp,cutout.Vm_extract,'b')
    axis(axis_limits)
    xlabel('time (samples)')
subplot(4,1,2); hold on % Field each rep
    plot(tb_temp,1000*temp+90,'k')
    axis(axis_limits)
subplot(4,1,3); hold on % Field mean±SEM
    plot(tb_temp,cutout.field_mean,'k','LineWidth',1.5)
    ciplot((cutout.field_mean)-(cutout.field_sem),...
           (cutout.field_mean)+(cutout.field_sem),...
           tb_temp,'k')
    axis(axis_limits)
subplot(4,1,4); hold on
    % plot(cutout.hilbert_field*180/pi,'g')
    plot(tb_temp,cutout.hilbert_field_mean*180/pi,'g','LineWidth',1.5)
    ciplot((cutout.hilbert_field_mean-cutout.hilbert_field_sem)*180/
pi,...
           (cutout.hilbert_field_mean+cutout.hilbert_field_sem)*180/
pi,...
           tb_temp,'g')
    axis(axis_limits)



    figure;
```

```
rose(cutout.hilbert_phase,18)
% clean up
clear amp_temp idx spike_id temp temp1 temp2 trial_id time_temp
```

Population phase rose

```
h1=rose(WT.MeanPhase,36)
    x1 = get(h1,'Xdata');
    y1 = get(h1,'Ydata');
    g1   = patch(x1,y1,'b');
    set(g1,'EdgeAlpha',0,'EdgeColor',[0 0
 1],'FaceAlpha',0.8,'FaceColor',[0 0 1])

hold on

h2=rose(KO.MeanPhase,36)
    x2 = get(h2,'Xdata');
    y2 = get(h2,'Ydata');
    g2   = patch(x2,y2,'r');
    set(g2,'EdgeAlpha',0,'EdgeColor',[1 0
 0],'FaceAlpha',0.8,'FaceColor',[1 0 0])

%       set(gca,'View',[-90 90])
      set(gca,'XDir','reverse');
```

Cut and align AP to input waveforms

```
cutout_Inj=[];
cutout_Inj.reject_boundaries=[6500 43280];
cutout_Inj.peak_pos_Inj=zeros(30,size(Vm,2));
 cutout_Inj.peak_pos_Inj(cutout_Inj.peak_pos_Inj==0)=NaN;
cutout_Inj.peak_times_Inj=zeros(30,size(Vm,2));
 cutout_Inj.peak_times_Inj(cutout_Inj.peak_times_Inj==0)=NaN;
temp1=[]; temp2=[];
temp_Iinj=field;
for trial_id=1:size(Vm,2)
    temp_Iinj(:,trial_id)=zscore(temp_Iinj(:,trial_id));
end
temp_Iinj([1:6500 43280:size(temp_Iinj,1)],:)=NaN;
% figure ; hold on
for trial_id=1:size(Vm,2)
    [temp1 temp2]=
 findpeaks(temp_Iinj(:,trial_id),'MINPEAKHEIGHT',1,'MINPEAKDISTANCE',500); %amp,
 time
    temp1(temp2<cutout_Inj.reject_boundaries(1) |
 temp2>cutout_Inj.reject_boundaries(2))=[];
    temp2(temp2<cutout_Inj.reject_boundaries(1) |
 temp2>cutout_Inj.reject_boundaries(2))=[];
    cutout_Inj.peak_pos_Inj(1:numel(temp1),trial_id)=temp1;
    cutout_Inj.peak_times_Inj(1:numel(temp2),trial_id)=temp2;
%     plot(tb,Vm(:,trial_id));
%
 scatter(tb(cutout.peak_times_Inj(:,trial_id)),cutout.peak_pos_Inj(:,trial_id)))
```

```
end

%
cutout_Inj.window_size=1000;
cutout_Inj.Vm_extract=[];cutout_Inj.field_extract=[];
for trial_id=1:size(Vm,2)
   time_temp= cutout_Inj.peak_times_Inj(:,trial_id);
 time_temp(isnan(time_temp))=[];
   amp_temp = cutout_Inj.peak_pos_Inj(:,trial_id);
 amp_temp(isnan(time_temp))=[];
    for spike_id =1:numel(time_temp)
        cutout_Inj.Vm_extract{trial_id}
(:,spike_id)=Vm(time_temp(spike_id)-
cutout_Inj.window_size:time_temp(spike_id)+cutout_Inj.window_size,
 trial_id);
        cutout_Inj.field_extract{trial_id}
(:,spike_id)=field(time_temp(spike_id)-
cutout_Inj.window_size:time_temp(spike_id)+cutout_Inj.window_size,
 trial_id);
    end
end

cutout_Inj.Vm_extract=cell2mat(cutout_Inj.Vm_extract);
 cutout_Inj.field_extract=cell2mat(cutout_Inj.field_extract);

for spike_id =1:size(cutout_Inj.field_extract,2)

 cutout_Inj.hilbert_field(:,spike_id)=(angle(hilbert(cutout_Inj.field_extract(:,sp
% cutout.hilbert =
 atan2(imag(hilbert(cutout.field_extract(:,1))),real(hilbert(cutout.field_extract(
pi;
end

cutout_Inj.hilbert_phase=cutout_Inj.hilbert_field(1000,:);
cutout_Inj.phasemean=nanmean(cutout_Inj.hilbert_phase)
cutout_Inj.phasesem=nansem(cutout_Inj.hilbert_phase)
% plot
temp=[];
for idx=1:size(cutout_Inj.field_extract,2)
    temp(:,idx)=cutout_Inj.field_extract(:,idx)-
mean(cutout_Inj.field_extract(:,idx));
end
figure;
% subplot(2,1,1);
hold on
line([1000, 1000],[-2,9],'Color',[0.5 0.5
 0.5 ],'LineStyle','-','LineWidth',2)
plot(zscore(mean(temp,2)),'k','lineWidth',4)
plot(zscore(mean(cutout_Inj.hilbert_field,2)*180/
pi),'g','LineWidth',4)
plot(3+zscore(cutout_Inj.Vm_extract),'r')
% axis([200 1800 -2 10])
axis([666 1341 -2 10])
% subplot(2,1,2); hold on
```

```
% figure;
% rose(cutout_Inj.hilbert_phase,36)
% clean up
 box off ; axis off
set(gca,'XTick',[]);set(gca,'YTick',[])
clear amp_temp idx spike_id temp temp1 temp2 trial_id time_temp temp_Iinj
```

Published with MATLAB® R2018a