Table of Contents


```
function SumPPICIBin(varargin)
```

get user input and set up file names

```
n = 1;
while n <= length(varargin)
    switch varargin{n}
        case 'filePrefix'
            filePrefix = varargin{n+1}; n=n+2;
        case 'sp'
            sp = varargin{n+1}; n=n+2;
        case 'sdir'
            sdir = varargin{n+1}; n=n+2;
        case 'concatFiles'
            concatFiles = varargin{n+1}; n=n+2;
        case 'countType'
            countType = varargin{n+1}; n=n+2;
        case 'effort'
            effort = varargin{n+1}; n=n+2;
        case 'refTime'
            refStartTime = varargin{n+1}; n=n+2;
        otherwise
            error('Bad optional argument: "%s"', varargin{n});
    end
end
```

get default parameters

```
p = sp_setting_defaults('sp',sp,'analysis','SumPPICIBin');
```

Undefined function or variable 'sp'.

Error in SumPPICIBin (line 27)

```
p = sp_setting_defaults('sp',sp,'analysis','SumPPICIBin');
```

Concatenate variables

```
PPall = []; TTall = []; ICIall = []; % initialize matrices
for idsk = 1 : length(concatFiles)
    % Load file
    fprintf('Loading %d/%d file %s
\n',idsk,length(concatFiles),fullfile(sdir,concatFiles{idsk}))
    load(fullfile(sdir,concatFiles{idsk}));

    PPall = [PPall; MPP];   % group peak-to-peak
    TTall = [TTall; MTT];   % group start times

    % Inter-Click Interval
    ici = diff(MTT)*24*60*60*1000; % in ms
    ICIall = [ICIall;ici];  % group inter-click interval
end
```

Convert times to bin vector times

```
vTT = datevec(TTall);
tbin = datetime([vTT(:,1:4), floor(vTT(:,5)/p.binDur)*p.binDur, ...
    zeros(length(vTT),1)]);
```

create table and get click counts and max pp per bin

```
data = timetable(tbin,TTall,PPall);
binData =
 varfun(@max,data,'GroupingVariable','tbin','InputVariable','PPall');
binData.Properties.VariableNames{'GroupCount'} = 'Count'; % #clicks
 per bin
binData.Properties.VariableNames{'max_PPall'} = 'maxPP';

positiveCounts = sum(binData.Count);
positiveBins = length(binData.Count);
```

group effort in bins

```
effort.diffSec = seconds(effort.End-effort.Start) ;
effort.bins = effort.diffSec/(60*p.binDur);
effort.roundbin = round(effort.diffSec/(60*p.binDur));

secMonitEffort = sum(effort.diffSec);
binMonitEffort = sum(effort.roundbin);
```

get average of detection by effort

```
NktTkt = positiveCounts/secMonitEffort;
```

```
NktTktbin = positiveBins/binMonitEffort;
disp(['Nkt/Tkt for click counting: ',num2str(NktTkt)]);
disp(['Nkt/Tkt for group counting: ',num2str(NktTktbin)]);
```

group data by days and weeks

```
Click = retime(binData(:,1),'daily','sum'); % #click per day
Bin = retime(binData(:,1),'daily','count'); % #bin per day
dayData = synchronize(Click,Bin);
weekData = retime(dayData,'weekly','mean');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Create plots, binlog and pplog files

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Plot Inter-Click Interval

```
iciIdx = find(ICIall > p.iciRange(1) & ICIall < p.iciRange(2));
% statistics
miciSel = mean(ICIall(iciIdx));
sdiciSel = std(ICIall(iciIdx));
moiciSel = mode(ICIall(iciIdx));
meiciSel = median(ICIall(iciIdx));

figure(1); set(1,'name','Inter-Click Interval')
h1 = gca;
centerIci = p.iciRange(1):1:p.iciRange(2);
[nhist] = histc(ICIall(iciIdx),centerIci);
bar(h1,centerIci,nhist, 'barwidth', 1, 'basevalue', 1)
xlim(h1,p.iciRange);
title(h1,sprintf('%s N=
%d',filePrefix,length(ICIall)), 'Interpreter', 'none')
xlabel(h1,'Inter-Click Interval (ms)')
ylabel(h1,'Counts')
% create labels and textbox
mnlabel = sprintf('Mean = %0.2f', miciSel);
stdlabel = sprintf('Std = %0.2f', sdiciSel);
melabel = sprintf('Median = %0.2f', meiciSel);
molabel = sprintf('Mode = %0.2f', moiciSel);
annotation('textbox',[0.58 0.75 0.1 0.1],'String',
{mnlabel,stdlabel,...
    melabel,molabel});

% save ici data and figure
icifn = [filePrefix,'_',p.speName,'_ici'];
% saveas(h1,fullfile(sdir,icifn),'png')
```

Plot Peak-to-peak

statistics

```
mpp = mean(PPall);
sdpp = std(PPall);
mopp = mode(PPall);
mepp = median(PPall);

% Plot histogram
figure(2); set(2,'name','Received Levels')
h2 = gca;
center = p.threshRL:1:p.p1Hi;
[nhist] = histc(PPall,center);
bar(h2,center, nhist, 'barwidth', 1, 'basevalue', 1)
title(h2,sprintf('%s N=
%d',filePrefix,length(PPall)), 'Interpreter', 'none')
xlabel(h2,'Peak-Peak Amplitude (dB)')
xlim(h2,[p.threshRL, p.p1Hi])
% create labels and textbox
mnlabel = sprintf('Mean = %0.2f', mpp);
stdlabel = sprintf('Std = %0.2f', sdpp);
melabel = sprintf('Median = %0.2f', mepp);
molabel = sprintf('Mode = %0.2f', mopp);
annotation('textbox',[0.58 0.75 0.1 0.1],'String',
{mnlabel,stdlabel,...
    melabel,molabel});

% Save plot
ppfn = [filePrefix,'_',p.speName,'_pp'];
% saveas(h2,fullfile(sdir,ppfn),'png')
```

Percent log histogram PP click counting

```
figure(3); set(3,'name','Received Levels in logarithmic scale')
h3 = gca;
nper = nhist*100/length(PPall); % percentage
bar(h3,center,nper, 'barwidth', 1, 'basevalue',
 1,'EdgeColor','k','FaceColor','w');
set(h3,'YScale','log')
xlim(h3,[min(center)-0.5,max(center)+0.5]) % center is in the middle
 of the bin
title(h3,sprintf('%s N=
%d',filePrefix,length(PPall)), 'Interpreter', 'none')
ylabel(h3,'Percent of detections')
xlabel(h3,'Peak-Peak Amplitude (dB)');

% Save plot and pplog values in a mat file
pplog = [filePrefix,'_',p.speName,'_pplog'];
% save(fullfile(sdir,pplog),'center','nper')
% saveas(h3,fullfile(sdir,pplog),'png')
```

Plot weekly mean of detections

```
figure(4); set(4,'name','Weekly presence')
h4(1) = subplot(2,1,1);
h4(2) = subplot(2,1,2);
```

```
bar(h4(1),weekData.tbin,weekData.Count_Click)
bar(h4(2),weekData.tbin,weekData.Count_Bin)
set(h4(1),'xticklabel', '');
ylabel(h4(1),{'Weekly mean';'of clicks per day'})
ylabel(h4(2),{'Weekly mean';['of ', num2str(p.binDur), ' min bins per
 day']})
title(h4(1),'Click Counting')
title(h4(2),'Group Counting');
axis (h4(1),'tight')
axis (h4(2),'tight')

% define step according to number of weeks
if length(weekData.tbin) > 53 && length(weekData.tbin) <= 104 % 2
 years
    step = calmonths(1);
elseif length(weekData.tbin) > 104 && length(weekData.tbin) <= 209 % 4
 years
    step = calmonths(2);
elseif length(weekData.tbin) > 209 && length(weekData.tbin) <= 313 % 6
 years
    step = calmonths(3);
elseif length(weekData.tbin) > 313 && length(weekData.tbin) <= 417 % 8
 years
    step = calmonths(4);
elseif length(weekData.tbin) >= 417 % more than 8 years
    step = calyears(1);
end
% define tick steps only if more than 1 year of data
if length(weekData.tbin) > 53
    xtickformat('MMMyy')
    xticks(weekData.tbin(1):step:weekData.tbin(end))
    xtickangle(45)
end

% publish('summary','pdf')
% %%
% %plot diel pattern
% figure(fignum); fignum = fignum +1;
% sutc = 5;% shift to local midnight
% dm = mod(diel - sutc,24);
% dcenter = 0.5:1:23.5;
% [dhist] = hist(dm,dcenter);
% hd = bar(dcenter, dhist, 'barwidth', 1, 'basevalue', 1);
% set(hd,'EdgeColor','k','FaceColor','w')
% xlim([0,24]);
% ylim([0,1.05*max(dhist)])
% ylabel(gca, 'Number of Bins','FontSize',16)
% xlabel(gca,['Time of Day (UTC-',num2str(sutc),')'],'FontSize',16);
% dstr=[p.speName,' ',site,' Number Bins= ',num2str(length(diel))];
% title(dstr);
% dfn = [site,'_',p.speName,'_diel.pdf'];
% fnd = fullfile(detpn,dfn);
% saveas(gcf,fnd,'pdf')
%
```

```
% %plot pp bin data
% figure(fignum); fignum = fignum +1; %linear histogram
% %hist(pp,100);
% lbin = length(bin);
% mbin = mean(bin);
% sdbin = std(bin);
% %[n, center] = hist(pp,100);
% [nbhist] = hist(bin,center);
% h2 = bar(center, nbhist, 'barwidth', 1, 'basevalue', 1);
% set(h2,'EdgeColor','k','FaceColor','w')
% xlim([min(center),max(center)])
% ylim([0,1.05*max(nbhist)])
% ylabel(gca, 'Number of bins','FontSize',16)
% xlabel(gca,'Peak-Peak Amplitude (dB)','FontSize',16);
% binstr=[p.speName,' ',site,' ','Mean= ',num2str(mbin),...
%     'dB  StDev= ',num2str(sdbin),' Number= ',num2str(length(bin))];
% title(binstr)
%
% figure(fignum); fignum = fignum +1; %percent log histogram PP click
% nbper = nbhist*100/lbin; % percentage
% h3 = bar(center,nbper, 'barwidth', 1, 'basevalue', 1);
% set(h3,'EdgeColor','k','FaceColor','w')
% %plot(center,n*100/lpp);
% set(gca,'YScale','log')
% xlim([min(center),max(center)])
% ylim([.1,50])
% set(gca,'FontSize',12)
% title(binstr)
% ylabel(gca, 'Percent of bins','FontSize',16)
% xlabel(gca,'Peak-Peak Amplitude (dB)','FontSize',16);
% binlog = [site,'_',p.speName,'_binlog.mat'];
% fnblog = fullfile(detpn,binlog);
% save(fnblog,'center','nbper')
% binfn = [site,'_',p.speName,'_bin.pdf'];
% fn5 = fullfile(detpn,ppfn);
% saveas(gcf,fn5,'pdf')

end
```

Published with MATLAB® R2016b