**Fiber Photometry MATLAB code used in Flanigan et al. 2022**

GCaMP Signal Processing (470 nm) for two brain regions using 415 nm as an isosbestic control channel:

% for use with arduino generated timestamps, trig 1, 2 roi

FP.rawdata = Z676data0;

m=676

filename = sprintf('SI\_end\_Analyzed\_%d.xls',m)

picz=sprintf('gcamp\_%dzscore',m)

piccorrection =sprintf( 'gcamp%dcorrection',m)

picchopped=sprintf( 'gcamp\_%dchop',m)

motioncorrected= sprintf('gcamp\_%dmotioncorrect',m)

regression= sprintf('gcamp\_%dregression',m)

FP.background1=3225

FP.background2=3065

%fixes uneven length problem

if rem(length(FP.rawdata), 2) == 0

stop=length(FP.rawdata)

else

stop=(length(FP.rawdata)-1)

end

%

% f1 = figure;

% for i = 1:4

% subplot(4,1,i)

% plot(FP.rawdata(:,i))

% end

%f2 deinterleaves and plots each trace individually. written for trig 1

f2 = figure;

subplot(2,3,1)

plot(downsample(FP.rawdata(:,2),2))%takes column 2 data, takes every 2nd data point starting with the first

title('LHb')

subplot(2,3,4)

plot(downsample(FP.rawdata(2:end,2),2))%same thing but starts with the second number

title('LHb')

subplot(2,3,2)

plot(downsample(FP.rawdata(:,3),2))%same thing but starts with the second number

title('BNST')

subplot(2,3,5)%repeats the same with column 4

plot(downsample(FP.rawdata(2:end,3),2))

title('BNST')

%subplot(2,3,3)

%plot(downsample(FP.rawdata(:,4),2))

%title('OFC')

%subplot(2,3,6)

%plot(downsample(FP.rawdata(2:end,4),2))%same thing but starts with the second number

%title('OFC')

% % starts wtih 415

FP.calcium\_dependent(:,1) = downsample(FP.rawdata(2:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.calcium\_dependent(:,2) = downsample(FP.rawdata(2:stop,2),2); % bla

FP.calcium\_dependent(:,3) = downsample(FP.rawdata(2:stop,3),2); % bnst

%FP.calcium\_dependent(:,4) = downsample(FP.rawdata(2:stop,3),2); % ofc

FP.calcium\_dependent(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

FP.isos(:,1) = downsample(FP.rawdata(1:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.isos(:,2) = downsample(FP.rawdata(1:stop,2),2); % bla

FP.isos(:,3) = downsample(FP.rawdata(1:stop,3),2); % bnst

%FP.isos(:,4) = downsample(FP.rawdata(1:stop,3),2); % ofc

FP.isos(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

%starts wtih 470

FP.calcium\_dependent(:,1) = downsample(FP.rawdata(1:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.calcium\_dependent(:,2) = downsample(FP.rawdata(1:stop,2),2); % bla

FP.calcium\_dependent(:,3) = downsample(FP.rawdata(1:stop,3),2); % bnst

%FP.calcium\_dependent(:,4) = downsample(FP.rawdata(1:stop,4),2); % ofc

FP.calcium\_dependent(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

FP.isos(:,1) = downsample(FP.rawdata(2:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.isos(:,2) = downsample(FP.rawdata(2:stop,2),2); % bla

FP.isos(:,3) = downsample(FP.rawdata(2:stop,3),2); % bnst

%FP.isos(:,4) = downsample(FP.rawdata(2:stop,4),2); % ofc

FP.isos(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

%stop=find(FP.calcium\_dependent(:,4)>=1.6); %chop off end of trial if needed, time is in minutes

%stop=find(FP.rawdata(:,1)>=(FP.timestamps(10,1)+150000))

f3 = figure; %plots data agaist time

for i = 1:2 %green first. top will be LHb, bottom is bnst

subplot(2,1,i)

plot(FP.calcium\_dependent(:,4),FP.calcium\_dependent(:,i+1),'b')

hold on

plot(FP.calcium\_dependent(:,4), FP.isos(:,i+1),'m')

xlabel('Time (min)','fontsize',14)

ylabel('Signal Intensity (AU)','fontsize',14)

title('470','fontsize',16)

end

%just subtracts the background for each column from the raw data

FP.backgroundsubtracted(:,1)= FP.calcium\_dependent(:,1); %keeps time in first col

FP.backgroundsubtracted(:,2)= (FP.calcium\_dependent(:,2)-FP.background1);% subtracts branch 1 background from branch 1 data

FP.backgroundsubtracted(:,3)= (FP.calcium\_dependent(:,3)-FP.background2);% right

%FP.backgroundsubtracted(:,4)= (FP.calcium\_dependent(:,4)-FP.background2);

FP.backgroundsubtracted(:,4)= FP.calcium\_dependent(:,4);% keeps modified time in 4th col

FP.backgroundsubtractedisos(:,1)= FP.isos(:,1); %keeps time in first col

FP.backgroundsubtractedisos(:,2)= (FP.isos(:,2)-FP.background1);% subtracts branch 1 background from branch 1 data

FP.backgroundsubtractedisos(:,3)= (FP.isos(:,3)-FP.background2);% right

%FP.backgroundsubtractedisos(:,4)= (FP.isos(:,4)-FP.background2);

FP.backgroundsubtractedisos(:,4)= FP.isos(:,4);% keeps modified time in 4th col

%filer specs

Fstop=5

Fsample=20

order=10

%design filter based on parameters specified above

butterfive = designfilt('lowpassiir', 'FilterOrder', order, 'HalfPowerFrequency', Fstop, 'SampleRate', Fsample, 'DesignMethod', 'butter');

filt=butterfive;

FP.filtered(:,1)= FP.backgroundsubtracted(:,1);

FP.filtered(:,2)= filtfilt(filt,FP.backgroundsubtracted(:,2));

FP.filtered(:,3)= filtfilt(filt,FP.backgroundsubtracted(:,3));

FP.filtered(:,4)= FP.backgroundsubtracted(:,4);

FP.filteredisos(:,1)= FP.backgroundsubtractedisos(:,1);

FP.filteredisos(:,2)= filtfilt(filt,FP.backgroundsubtractedisos(:,2));

FP.filteredisos(:,3)= filtfilt(filt,FP.backgroundsubtractedisos(:,3));

FP.filteredisos(:,4)= FP.backgroundsubtracted(:,4);

start=find(FP.calcium\_dependent(:,4)>=0.25); %chop off beginning of trial if needed, time is in minutes

f4 = figure; %fit signal to bioexp and subtract and normalize470 data

subplot(4,2,1);

temp\_fit = fit(FP.filtered(start:end,4),FP.filtered(start:end,2),'exp2');

plot(temp\_fit,FP.filtered(start:end,4),FP.filtered(start:end,2))

title('Uncorrected LHb 470','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,2);

temp\_fit2 = fit(FP.filtered(start:end,4),FP.filtered(start:end,3),'exp2');

plot(temp\_fit2,FP.filtered(start:end,4),FP.filtered(start:end,3))

title('Uncorrected BNST 470','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,5);

temp\_fit3 = fit(FP.filteredisos(start:end,4),FP.filteredisos(start:end,2),'exp2');

plot(temp\_fit3,FP.filteredisos(start:end,4),FP.filteredisos(start:end,2))

title('Uncorrected LHb isos','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,6);

temp\_fit4 = fit(FP.filteredisos(start:end,4),FP.filteredisos(start:end,3),'exp2');

plot(temp\_fit4,FP.filteredisos(start:end,4),FP.filteredisos(start:end,3))

title('Uncorrected BNST isos','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,3);

plot(FP.filtered(start:end,4),100\*(FP.filtered(start:end,2)-temp\_fit(FP.filtered(start:end,4)))./temp\_fit(FP.filtered(start:end,4)))

title('Corrected LHb 470','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

subplot(4,2,4);

plot(FP.filtered(start:end,4),100\*(FP.filtered(start:end,3)-temp\_fit2(FP.filtered(start:end,4)))./temp\_fit2(FP.filtered(start:end,4)))

title('Corrected BNST 470','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

subplot(4,2,7);

plot(FP.filteredisos(start:end,4),100\*(FP.filteredisos(start:end,2)-temp\_fit3(FP.filteredisos(start:end,4)))./temp\_fit3(FP.filteredisos(start:end,4)))

title('Corrected LHb 415','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

subplot(4,2,8);

plot(FP.filteredisos(start:end,4),100\*(FP.filteredisos(start:end,3)-temp\_fit4(FP.filteredisos(start:end,4)))./temp\_fit4(FP.filteredisos(start:end,4)))

title('Corrected BNST 415','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

savefig(piccorrection)

%gives background subtracted data- the fit of background subtracted

%data/fit. this is df/f

FP.fitsubtract(:,1) = (FP.filtered(start:end,1));

FP.fitsubtract(:,2) = (FP.filtered(start:end,2)-temp\_fit(FP.filtered(start:end,4)))./temp\_fit(FP.filtered(start:end,4));

FP.fitsubtract(:,3) =(FP.filtered(start:end,3)-temp\_fit2(FP.filtered(start:end,4)))./temp\_fit2(FP.filtered(start:end,4));

%FP.fitsubtract(:,4) = (FP.filtered(start:end,4)-temp\_fit2(FP.filtered(start:end,5)))./temp\_fit3(FP.filtered(start:end,5));

FP.fitsubtract(:,4) = FP.calcium\_dependent(start:end,4);

FP.fitsubtractisos(:,1) = (FP.filteredisos(start:end,1));

FP.fitsubtractisos(:,2) = (FP.filteredisos(start:end,2)-temp\_fit3(FP.filteredisos(start:end,4)))./temp\_fit3(FP.filteredisos(start:end,4));

FP.fitsubtractisos(:,3) =(FP.filteredisos(start:end,3)-temp\_fit4(FP.filteredisos(start:end,4)))./temp\_fit4(FP.filteredisos(start:end,4));

%FP.fitsubtractisos(:,4) = (FP.filteredisos(start:end,4)-temp\_fit4(FP.filteredisos(start:end,5)))./temp\_fit4(FP.filteredisos(start:end,5));

FP.fitsubtractisos(:,4) = FP.calcium\_dependent(start:end,4);

FP.zscore(:,1)= FP.fitsubtract(:,1);

FP.zscore(:,2)=zscore(FP.fitsubtract(:,2));

FP.zscore(:,3)=zscore(FP.fitsubtract(:,3));

%FP.zscore(:,4)=zscore(FP.fitsubtract(:,4));

FP.zscore(:,4)=FP.fitsubtract(:,4);

FP.zscorei(:,1)= FP.fitsubtractisos(:,1);

FP.zscorei(:,2)=zscore(FP.fitsubtractisos(:,2));

FP.zscorei(:,3)=zscore(FP.fitsubtractisos(:,3));

%FP.zscorei(:,4)=zscore(FP.fitsubtractisos(:,4));

FP.zscorei(:,4)=FP.fitsubtractisos(:,4);

f7=figure %plots z scored data against time

subplot(2,1,1)

plot(FP.zscore(:,4),FP.zscore(:,2),'b')

hold on

plot(FP.zscorei(:,4),FP.zscorei(:,2),'m')

title('Z-scored LHb','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('Z-score','fontsize',14)

subplot(2,1,2)

plot(FP.zscore(:,4),FP.zscore(:,3),'b')

hold on

plot(FP.zscorei(:,4),FP.zscorei(:,3),'m')

title('Z-scored L','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('Z-score','fontsize',14)

savefig(picz)

% using non negative robust linear regression---DO NOT USE THIS PART IF

% USING A NON GCAMP SENSOR

fitdataR = fit(FP.zscorei(:,2),FP.zscore(:,2),fittype('poly1'),'Robust','on');

fitdataL = fit(FP.zscorei(:,3),FP.zscore(:,3),fittype('poly1'),'Robust','on');

% Plot fit

figure

subplot(2,1,1)

hold on

plot(FP.zscorei(:,2),FP.zscore(:,2),'k.')

plot(fitdataR,'b')

title('Lin reg R')

hold off

subplot(2,1,2)

hold on

plot(FP.zscorei(:,3),FP.zscore(:,3),'k.')

plot(fitdataL,'b')

title('lin reg L')

hold off

savefig(regression)

%fit isos to ca signal

isosfitted(:,1)=fitdataR(FP.zscorei(:,2));

isosfitted(:,2)=fitdataL(FP.zscorei(:,3));

% Plot aligned signals

figure

subplot(2,1,1)

plot(FP.zscore(:,4),isosfitted(:,1),'m')

hold on

plot(FP.zscore(:,4),FP.zscore(:,2),'b')

title('fitted isos and signal R')

hold off

subplot(2,1,2)

plot(FP.zscore(:,4),isosfitted(:,2),'m')

hold on

plot(FP.zscore(:,4),FP.zscore(:,3),'b')

title('fitted isos and signal L')

hold off

FP.corrected(:,1)=FP.zscore(:,1)

FP.corrected(:,2)=(FP.zscore(:,2)-isosfitted(:,1))

FP.corrected(:,3)=(FP.zscore(:,3)-isosfitted(:,2))

%FP.corrected(:,4)=(FP.zscore(:,2)-isosfitted(:,1))

FP.corrected(:,4)=FP.zscore(:,4)

figure

subplot(2,1,1)

plot(FP.corrected(:,4),FP.corrected(:,2),'k')

title('Motion Corrected signal R')

xlabel('Time')

ylabel('z-score')

hold on

subplot(2,1,2)

plot(FP.corrected(:,4),FP.corrected(:,3),'k')

title('Motion Corrected signal L')

xlabel('Time')

ylabel('z-score')

savefig(motioncorrected)

GRAB-5HT Signal Processing (470 nm) for two brain regions:

% for use with arduino generated timestamps, trig 1, 2 roi

FP.rawdata = Z947data0;

FP.timestamps =Z947timestamps;

m=947

filename = sprintf('etohDrinking\_Analyzed\_%d.xls',m)

picz=sprintf('gcamp\_%dzscore',m)

piccorrection =sprintf( 'gcamp%dcorrection',m)

picchopped=sprintf( 'gcamp\_%dchop',m)

motioncorrected= sprintf('gcamp\_%dmotioncorrect',m)

regression= sprintf('gcamp\_%dregression',m)

FP.background1=3225

FP.background2=3065

%fixes uneven length problem

if rem(length(FP.rawdata), 2) == 0

stop=length(FP.rawdata)

else

stop=(length(FP.rawdata)-1)

end

%stop=find(FP.rawdata(:,1)>=(FP.timestamps(10,1)+150000))

%

% f1 = figure;

% for i = 1:4

% subplot(4,1,i)

% plot(FP.rawdata(:,i))

% end

%f2 deinterleaves and plots each trace individually. written for trig 1

f2 = figure;

subplot(2,3,1)

plot(downsample(FP.rawdata(:,2),2))%takes column 2 data, takes every 2nd data point starting with the first

title('LHb')

subplot(2,3,4)

plot(downsample(FP.rawdata(2:end,2),2))%same thing but starts with the second number

title('LHb')

subplot(2,3,2)

plot(downsample(FP.rawdata(:,3),2))%same thing but starts with the second number

title('BNST')

subplot(2,3,5)%repeats the same with column 4

plot(downsample(FP.rawdata(2:end,3),2))

title('BNST')

%subplot(2,3,3)

%plot(downsample(FP.rawdata(:,4),2))

%title('OFC')

%subplot(2,3,6)

%plot(downsample(FP.rawdata(2:end,4),2))%same thing but starts with the second number

%title('OFC')

% % starts wtih 415

FP.calcium\_dependent(:,1) = downsample(FP.rawdata(2:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.calcium\_dependent(:,2) = downsample(FP.rawdata(2:stop,2),2); % bla

FP.calcium\_dependent(:,3) = downsample(FP.rawdata(2:stop,3),2); % bnst

%FP.calcium\_dependent(:,4) = downsample(FP.rawdata(2:stop,3),2); % ofc

FP.calcium\_dependent(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

FP.isos(:,1) = downsample(FP.rawdata(1:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.isos(:,2) = downsample(FP.rawdata(1:stop,2),2); % bla

FP.isos(:,3) = downsample(FP.rawdata(1:stop,3),2); % bnst

%FP.isos(:,4) = downsample(FP.rawdata(1:stop,3),2); % ofc

FP.isos(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

%starts wtih 470

FP.calcium\_dependent(:,1) = downsample(FP.rawdata(1:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.calcium\_dependent(:,2) = downsample(FP.rawdata(1:stop,2),2); % bla

FP.calcium\_dependent(:,3) = downsample(FP.rawdata(1:stop,3),2); % bnst

%FP.calcium\_dependent(:,4) = downsample(FP.rawdata(1:stop,4),2); % ofc

FP.calcium\_dependent(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

FP.isos(:,1) = downsample(FP.rawdata(2:stop,1),2); % takes every 3rd value from column 1 which is time stamps. adjust to start at second or third number to match what you need

FP.isos(:,2) = downsample(FP.rawdata(2:stop,2),2); % bla

FP.isos(:,3) = downsample(FP.rawdata(2:stop,3),2); % bnst

%FP.isos(:,4) = downsample(FP.rawdata(2:stop,4),2); % ofc

FP.isos(:,4) = (FP.calcium\_dependent(:,1)-FP.calcium\_dependent(1,1))./60000

f3 = figure; %plots data agaist time

for i = 1:2 %green first. top will be LHb, bottom is bnst

subplot(2,1,i)

plot(FP.calcium\_dependent(:,4),FP.calcium\_dependent(:,i+1),'b')

hold on

plot(FP.calcium\_dependent(:,4), FP.isos(:,i+1),'m')

xlabel('Time (min)','fontsize',14)

ylabel('Signal Intensity (AU)','fontsize',14)

title('470','fontsize',16)

end

%just subtracts the background for each column from the raw data

FP.backgroundsubtracted(:,1)= FP.calcium\_dependent(:,1); %keeps time in first col

FP.backgroundsubtracted(:,2)= (FP.calcium\_dependent(:,2)-FP.background1);% subtracts branch 1 background from branch 1 data

FP.backgroundsubtracted(:,3)= (FP.calcium\_dependent(:,3)-FP.background2);% right

%FP.backgroundsubtracted(:,4)= (FP.calcium\_dependent(:,4)-FP.background2);

FP.backgroundsubtracted(:,4)= FP.calcium\_dependent(:,4);% keeps modified time in 4th col

FP.backgroundsubtractedisos(:,1)= FP.isos(:,1); %keeps time in first col

FP.backgroundsubtractedisos(:,2)= (FP.isos(:,2)-FP.background1);% subtracts branch 1 background from branch 1 data

FP.backgroundsubtractedisos(:,3)= (FP.isos(:,3)-FP.background2);% right

%FP.backgroundsubtractedisos(:,4)= (FP.isos(:,4)-FP.background2);

FP.backgroundsubtractedisos(:,4)= FP.isos(:,4);% keeps modified time in 4th col

%filer specs

Fstop=5

Fsample=20

order=10

%design filter based on parameters specified above

butterfive = designfilt('lowpassiir', 'FilterOrder', order, 'HalfPowerFrequency', Fstop, 'SampleRate', Fsample, 'DesignMethod', 'butter');

filt=butterfive;

FP.filtered(:,1)= FP.backgroundsubtracted(:,1);

FP.filtered(:,2)= filtfilt(filt,FP.backgroundsubtracted(:,2));

FP.filtered(:,3)= filtfilt(filt,FP.backgroundsubtracted(:,3));

FP.filtered(:,4)= FP.backgroundsubtracted(:,4);

FP.filteredisos(:,1)= FP.backgroundsubtractedisos(:,1);

FP.filteredisos(:,2)= filtfilt(filt,FP.backgroundsubtractedisos(:,2));

FP.filteredisos(:,3)= filtfilt(filt,FP.backgroundsubtractedisos(:,3));

FP.filteredisos(:,4)= FP.backgroundsubtracted(:,4);

start=find(FP.calcium\_dependent(:,4)>=0.75); %chop off beginning of trial if needed, time is in minutes

f4 = figure; %fit signal to bioexp and subtract and normalize470 data

subplot(4,2,1);

temp\_fit = fit(FP.filtered(start:end,4),FP.filtered(start:end,2),'exp2');

plot(temp\_fit,FP.filtered(start:end,4),FP.filtered(start:end,2))

title('Uncorrected LHb 470','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,2);

temp\_fit2 = fit(FP.filtered(start:end,4),FP.filtered(start:end,3),'exp2');

plot(temp\_fit2,FP.filtered(start:end,4),FP.filtered(start:end,3))

title('Uncorrected BNST 470','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,5);

temp\_fit3 = fit(FP.filteredisos(start:end,4),FP.filteredisos(start:end,2),'exp2');

plot(temp\_fit3,FP.filteredisos(start:end,4),FP.filteredisos(start:end,2))

title('Uncorrected LHb isos','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,6);

temp\_fit4 = fit(FP.filteredisos(start:end,4),FP.filteredisos(start:end,3),'exp2');

plot(temp\_fit4,FP.filteredisos(start:end,4),FP.filteredisos(start:end,3))

title('Uncorrected BNST isos','fontsize',16)

ylabel('F (au)','fontsize',14)

subplot(4,2,3);

plot(FP.filtered(start:end,4),100\*(FP.filtered(start:end,2)-temp\_fit(FP.filtered(start:end,4)))./temp\_fit(FP.filtered(start:end,4)))

title('Corrected LHb 470','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

subplot(4,2,4);

plot(FP.filtered(start:end,4),100\*(FP.filtered(start:end,3)-temp\_fit2(FP.filtered(start:end,4)))./temp\_fit2(FP.filtered(start:end,4)))

title('Corrected BNST 470','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

subplot(4,2,7);

plot(FP.filteredisos(start:end,4),100\*(FP.filteredisos(start:end,2)-temp\_fit3(FP.filteredisos(start:end,4)))./temp\_fit3(FP.filteredisos(start:end,4)))

title('Corrected LHb 415','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

subplot(4,2,8);

plot(FP.filteredisos(start:end,4),100\*(FP.filteredisos(start:end,3)-temp\_fit4(FP.filteredisos(start:end,4)))./temp\_fit4(FP.filteredisos(start:end,4)))

title('Corrected BNST 415','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('dF/F %','fontsize',14)

savefig(piccorrection)

%gives background subtracted data- the fit of background subtracted

%data/fit. this is df/f

FP.fitsubtract(:,1) = (FP.filtered(start:end,1));

FP.fitsubtract(:,2) = (FP.filtered(start:end,2)-temp\_fit(FP.filtered(start:end,4)))./temp\_fit(FP.filtered(start:end,4));

FP.fitsubtract(:,3) =(FP.filtered(start:end,3)-temp\_fit2(FP.filtered(start:end,4)))./temp\_fit2(FP.filtered(start:end,4));

%FP.fitsubtract(:,4) = (FP.filtered(start:end,4)-temp\_fit2(FP.filtered(start:end,5)))./temp\_fit3(FP.filtered(start:end,5));

FP.fitsubtract(:,4) = FP.calcium\_dependent(start:end,4);

FP.fitsubtractisos(:,1) = (FP.filteredisos(start:end,1));

FP.fitsubtractisos(:,2) = (FP.filteredisos(start:end,2)-temp\_fit3(FP.filteredisos(start:end,4)))./temp\_fit3(FP.filteredisos(start:end,4));

FP.fitsubtractisos(:,3) =(FP.filteredisos(start:end,3)-temp\_fit4(FP.filteredisos(start:end,4)))./temp\_fit4(FP.filteredisos(start:end,4));

%FP.fitsubtractisos(:,4) = (FP.filteredisos(start:end,4)-temp\_fit4(FP.filteredisos(start:end,5)))./temp\_fit4(FP.filteredisos(start:end,5));

FP.fitsubtractisos(:,4) = FP.calcium\_dependent(start:end,4);

FP.zscore(:,1)= FP.fitsubtract(:,1);

FP.zscore(:,2)=zscore(FP.fitsubtract(:,2));

FP.zscore(:,3)=zscore(FP.fitsubtract(:,3));

%FP.zscore(:,4)=zscore(FP.fitsubtract(:,4));

FP.zscore(:,4)=FP.fitsubtract(:,4);

FP.zscorei(:,1)= FP.fitsubtractisos(:,1);

FP.zscorei(:,2)=zscore(FP.fitsubtractisos(:,2));

FP.zscorei(:,3)=zscore(FP.fitsubtractisos(:,3));

%FP.zscorei(:,4)=zscore(FP.fitsubtractisos(:,4));

FP.zscorei(:,4)=FP.fitsubtractisos(:,4);

f7=figure %plots z scored data against time

subplot(2,1,1)

plot(FP.zscore(:,4),FP.zscore(:,2),'b')

hold on

plot(FP.zscorei(:,4),FP.zscorei(:,2),'m')

title('Z-scored LHb','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('Z-score','fontsize',14)

subplot(2,1,2)

plot(FP.zscore(:,4),FP.zscore(:,3),'b')

hold on

plot(FP.zscorei(:,4),FP.zscorei(:,3),'m')

title('Z-scored L','fontsize',16)

xlabel('Time (m)','fontsize',14)

ylabel('Z-score','fontsize',14)

savefig(picz)