modification pipeline 23 - RMS detecting spindles

June 28, 2016

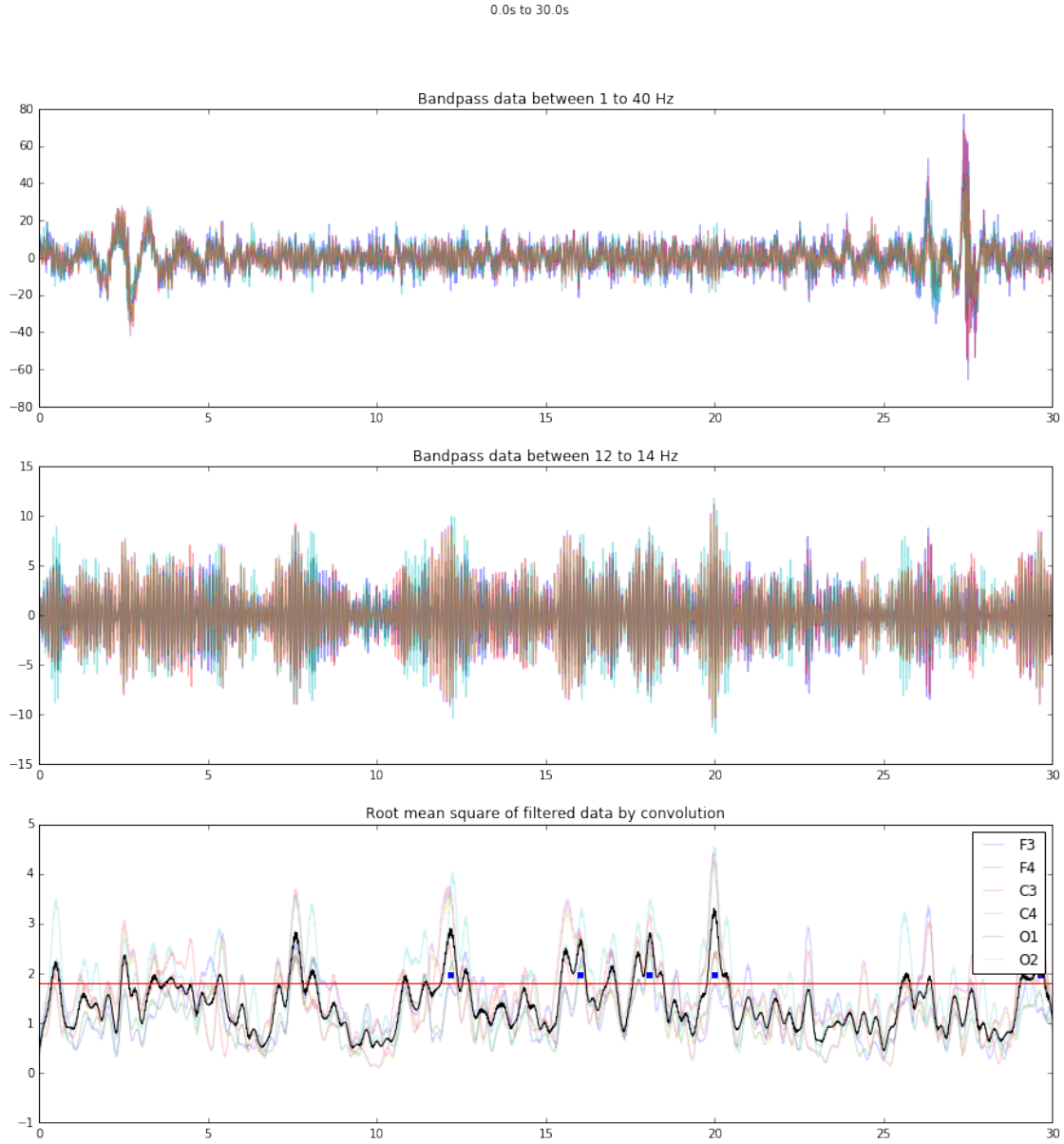
In [1]:
```
import eegPinelineDesign
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")
import mne
```

```
try:
    eegPinelineDesign.change_file_directory('C:/Users/ning/Downloads/training set')
except:
    pass
EDFfiles, Annotationfiles = eegPinelineDesign.split_type_of_files()

file_to_read='suj13_l2nap_day2.vhdr'
channelList = ['F3','F4','C3','C4','O1','O2','LOc', 'ROc']
raw = eegPinelineDesign.load_data(file_to_read,channelList,1,40)
# select channels of interests
channelList = ['F3','F4','C3','C4','O1','O2']
raw.pick_channels(channelList)
#raw_narrow.plot_psd(fmax=30,tmin=200,tmax=2000)


epochs = np.arange(0,raw.last_samp,30*1000)
epochSegment={};RMS={};peak_time=[];
for epch in range(len(epochs)-1):
    start,stop = (epochs[epch],epochs[epch+1])
    epochSegment[epch]=[]
    RMS[epch]=[]
    temp=[];temp2=[]
    fig=plt.figure(epch+1)
    fig.suptitle(str(epochs[epch]/1000)+'s to '+str(epochs[epch+1]/1000)+'s')
    fig.set_figheight(15)
    fig.set_figwidth(15)
    ###############################################################################
    for iii,names in enumerate(channelList):
        tempSegment,tempTime = raw[iii,start:stop]
        filter_Segment = mne.filter.band_pass_filter(tempSegment[0,:],1000,10,14)
        #take filtered segments to a list
        temp.append(filter_Segment)
```

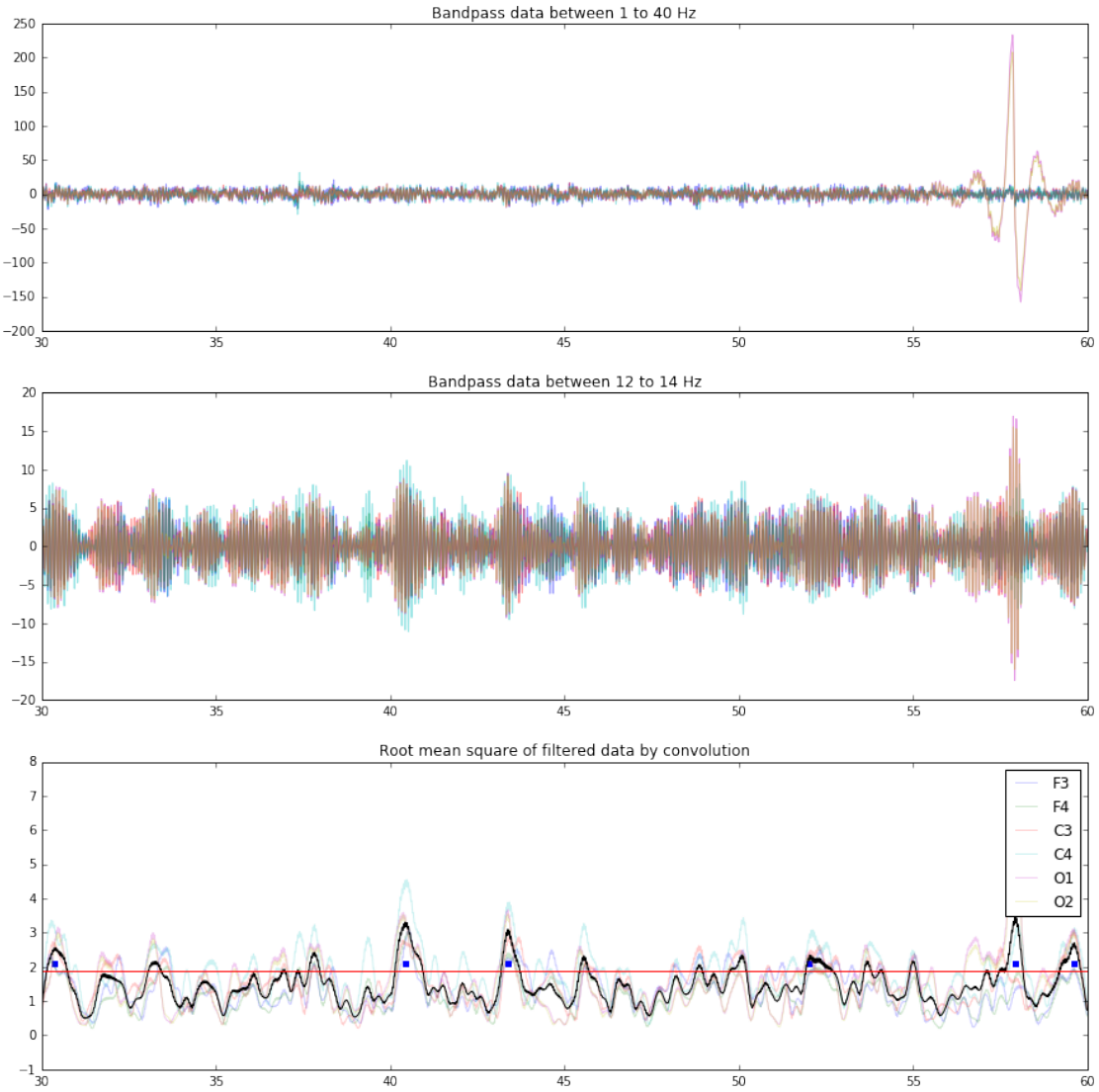
```
            ax1=plt.subplot(311)
            ax1.plot(tempTime,tempSegment[0,:],alpha=.4,label=names)
            ax1.set_title('Bandpass data between 1 to 40 Hz')
            ax2=plt.subplot(312)
            ax2.plot(tempTime,filter_Segment,alpha=.4,label=names)
            ax2.set_title('Bandpass data between 12 to 14 Hz')
            ax=plt.subplot(313)
            # convolve over 200 ms window over 30s of data
            RMS_Segment = eegPinelineDesign.window_rms(filter_Segment,200)
            temp2.append(RMS_Segment)
            ax.plot(tempTime,RMS_Segment,alpha=0.2,label=names)
            ax.set_title('Root mean square of filtered data by convolution')
        ###############################################################################
        ax.legend()
        epochSegment[epch]=np.array(temp)
        RMS[epch]=np.array(temp2)
        # temp data contains only 30s of data
        tempEpoch=np.array(temp)
        tempRMS=np.array(temp2)
        # calculate the means and standard deviation
        mean_tempEpoch=tempEpoch.mean(axis=0)
        mean_tempRMS=tempRMS.mean(axis=0)
        ax.plot(tempTime,mean_tempRMS,color='k')
        ax.set_xlim([tempTime[0],tempTime[-1]])
        # algorithm for peak search: 1-amplitude 2 - duration
        mph=mean_tempRMS.mean() + 0.8 * mean_tempRMS.std()
        ax.axhline(mph,color='r')
        pass_=mean_tempRMS > (mean_tempRMS.mean() + 0.8 * mean_tempRMS.std())
        up = np.where(np.diff(pass_.astype(int))>0)
        down = np.where(np.diff(pass_.astype(int))<0)
        if (up[0].shape > down[0].shape) or (up[0].shape < down[0].shape):
            size = np.min([up[0].shape,down[0].shape])
            up = up[0][:size]
            down = down[0][:size]
        C = np.vstack((up,down))
        for pairs in C.T:
            if 0.5 < (tempTime[pairs[1]] - tempTime[pairs[0]]) < 2.5:
                TimePoint = np.mean([tempTime[pairs[1]],tempTime[pairs[0]]])
                SegmentForPeakSearching = mean_tempRMS[pairs[0]:pairs[1]]
                temp_temp_time = tempTime[pairs[0]:pairs[1]]
                ints_temp = np.argmax(SegmentForPeakSearching)
                peak_time.append(temp_temp_time[ints_temp])
                ax.scatter(temp_temp_time[ints_temp],mph+0.1*mph,marker='s',color='blue')
```

```
['64chlocs.elp', 'after constraint by sleeping stage.png', 'AP-128-X2.bvef', 'before constraint by slee
Extracting parameters from suj13_l2nap_day2.vhdr...
Setting channel info structure...
Reading 0 ... 1805099  =      0.000 ...  1805.099 secs...
Band-pass filtering from 1 - 2e+02 Hz
Fitting ICA to data using 8 channels.
Please be patient, this may take some time
Inferring max_pca_components from picks.
Using all PCA components: 8
    Searching for artifacts...
```

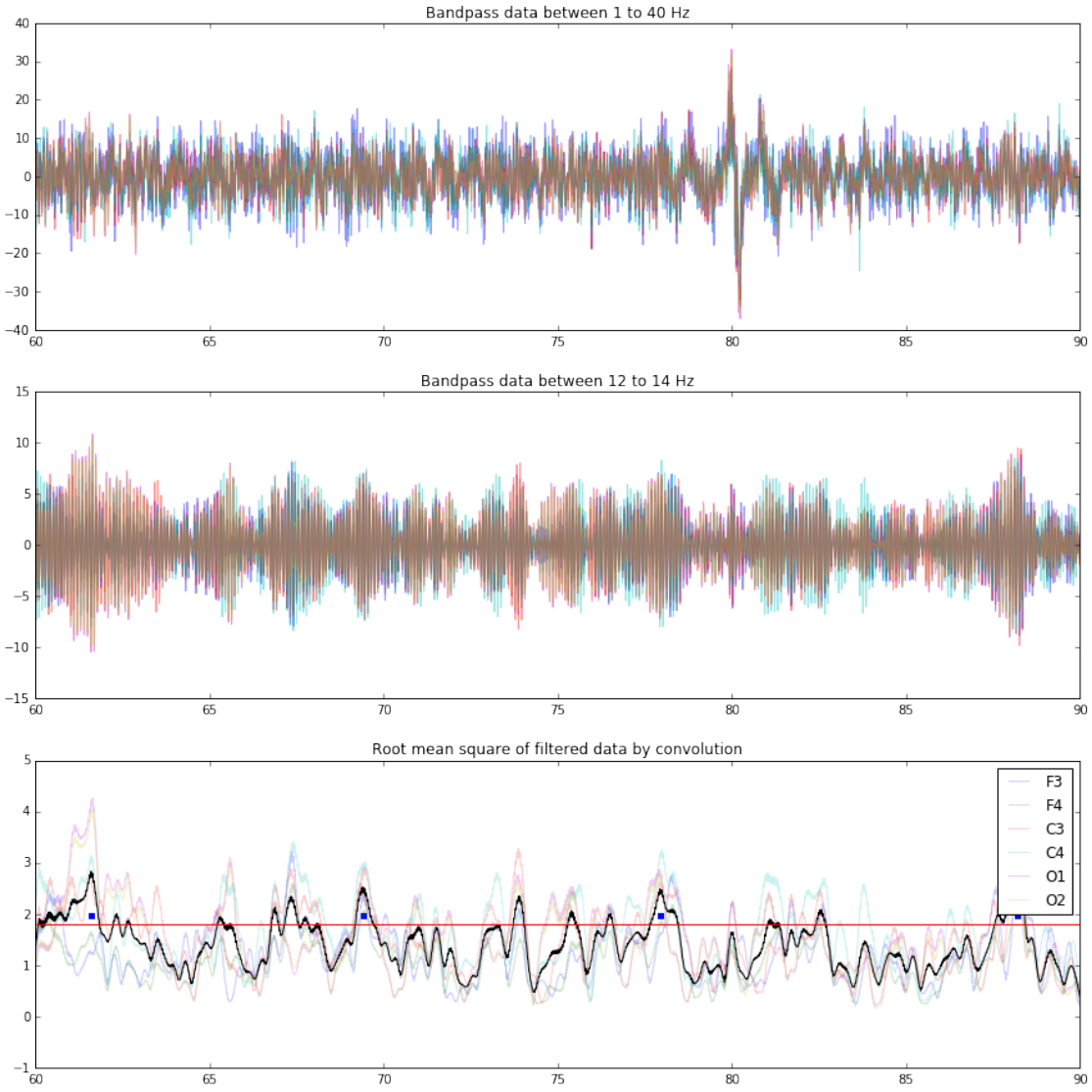
```
    found 2 artifacts by EOG 00
    found 2 artifacts by EOG 01
    found 1 artifact by skewness
    found 1 artifact by kurtosis
    found 1 artifact by variance
Artifact indices found:
    2, 6, 2, 6, 4, 4, 1
    Removing duplicate indices...
Ready.
Transforming to ICA space (8 components)
Zeroing out 4 ICA components
Band-pass filtering from 1 - 40 Hz
```



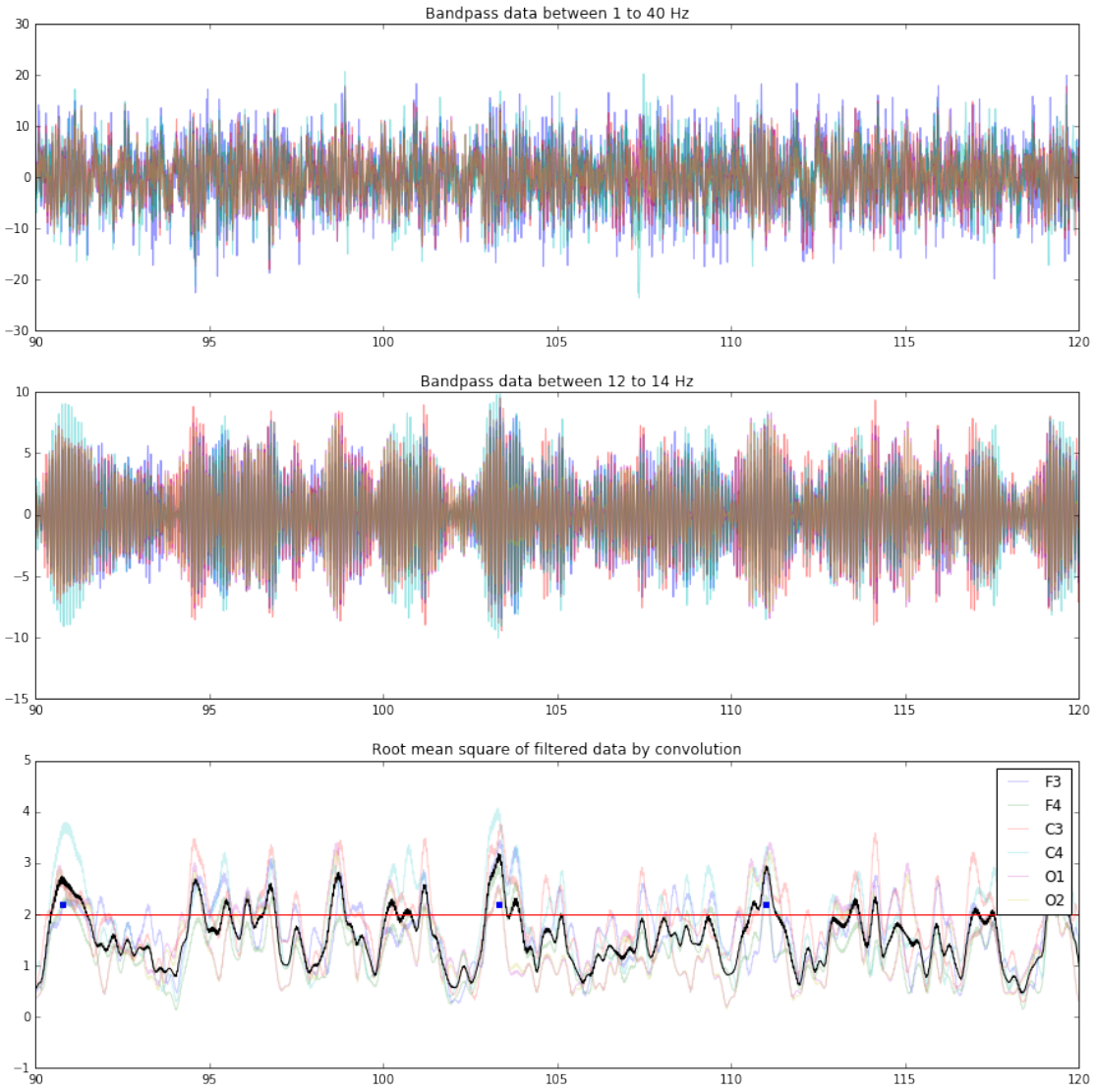
30.0s to 60.0s


Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

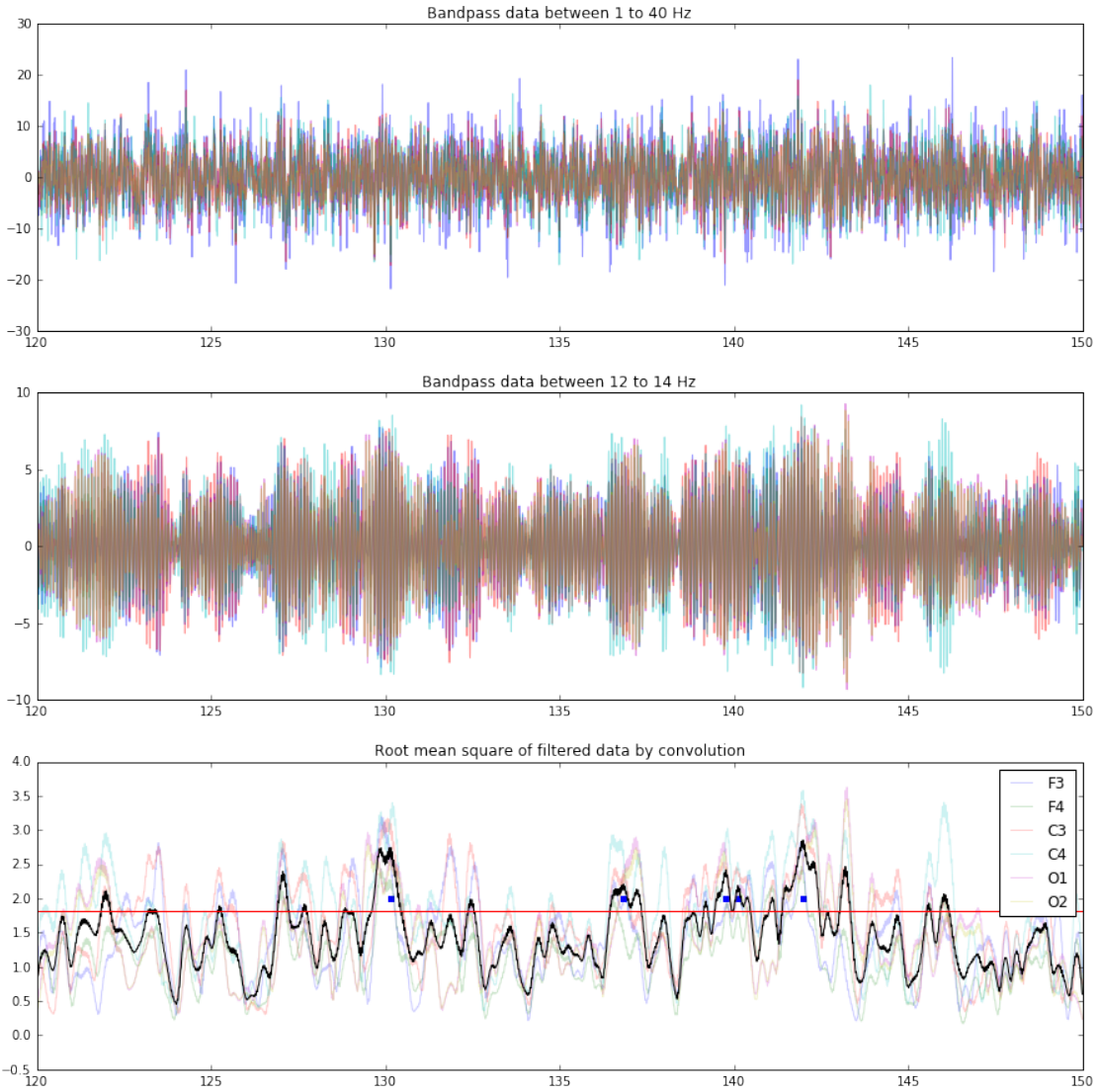
60.0s to 90.0s


Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

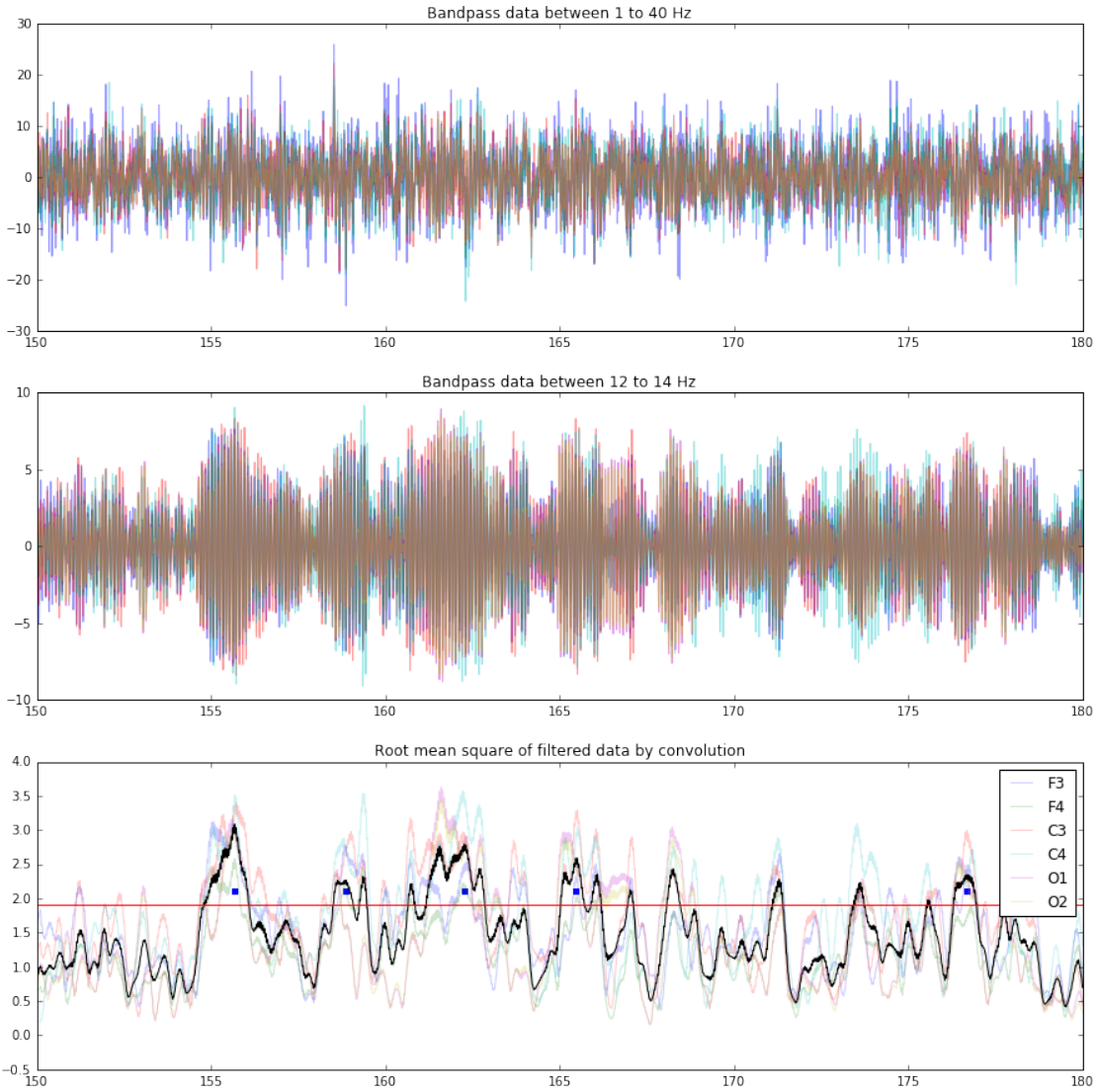
90.0s to 120.0s



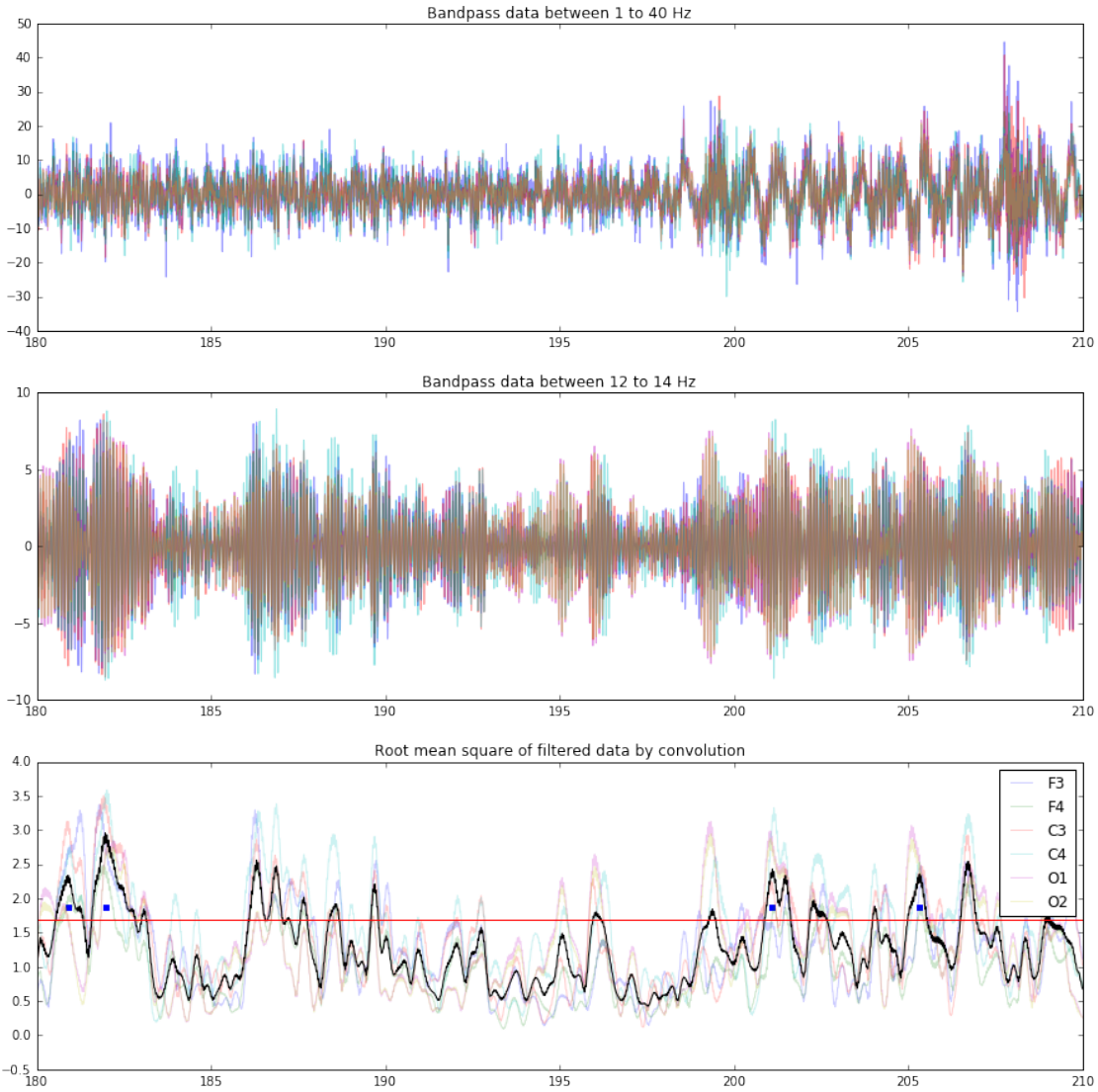
120.0s to 150.0s

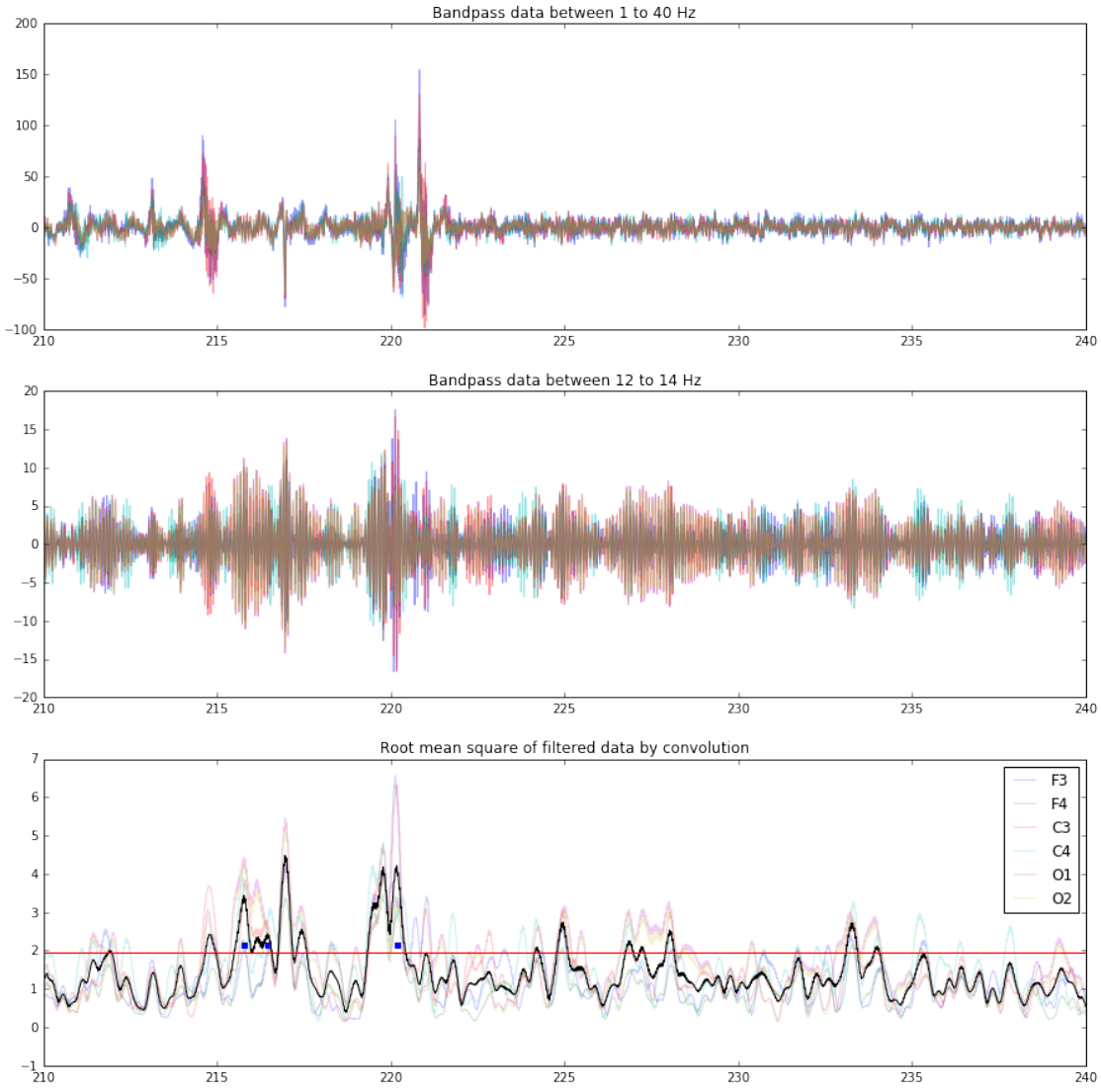


150.0s to 180.0s

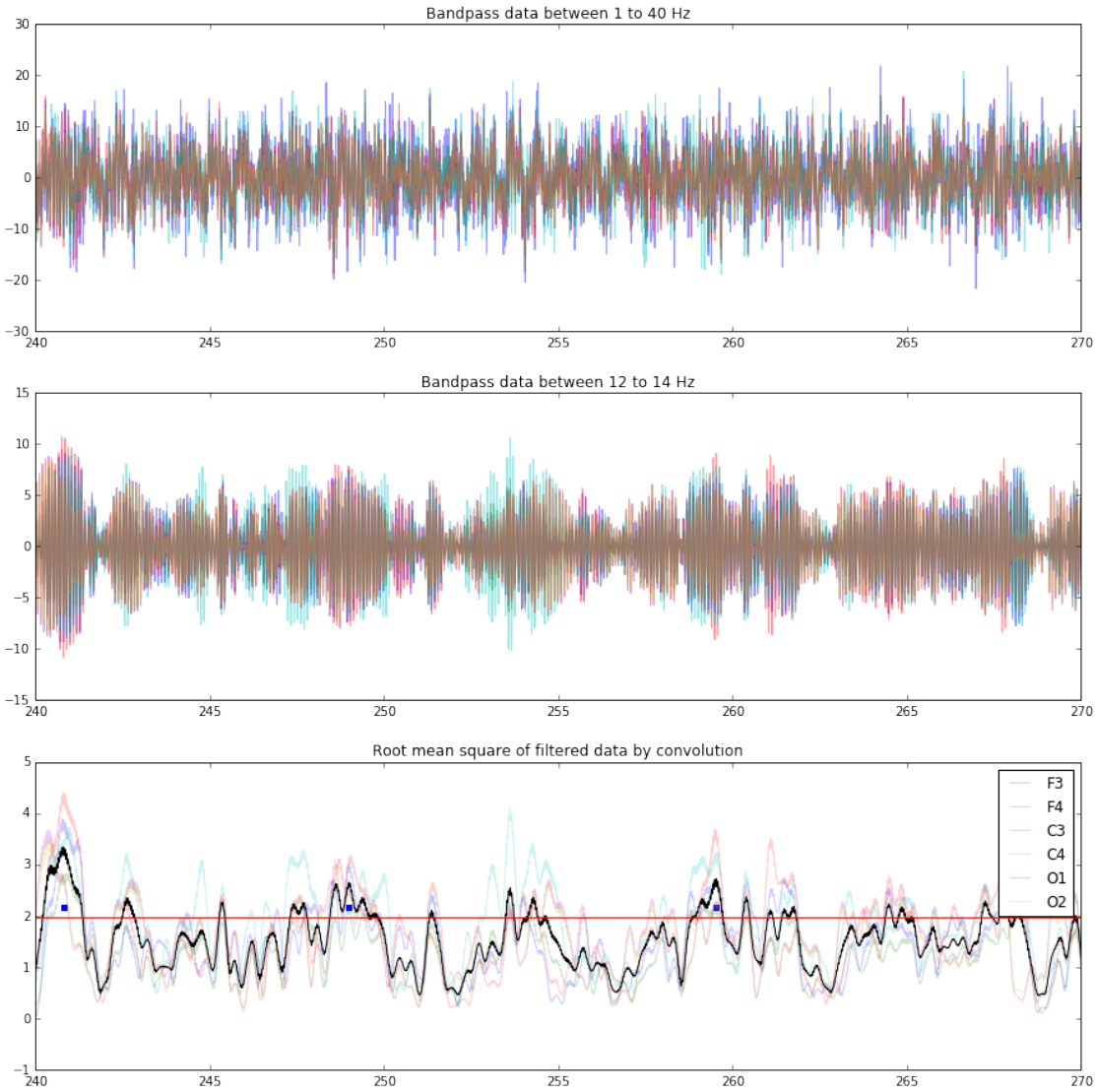


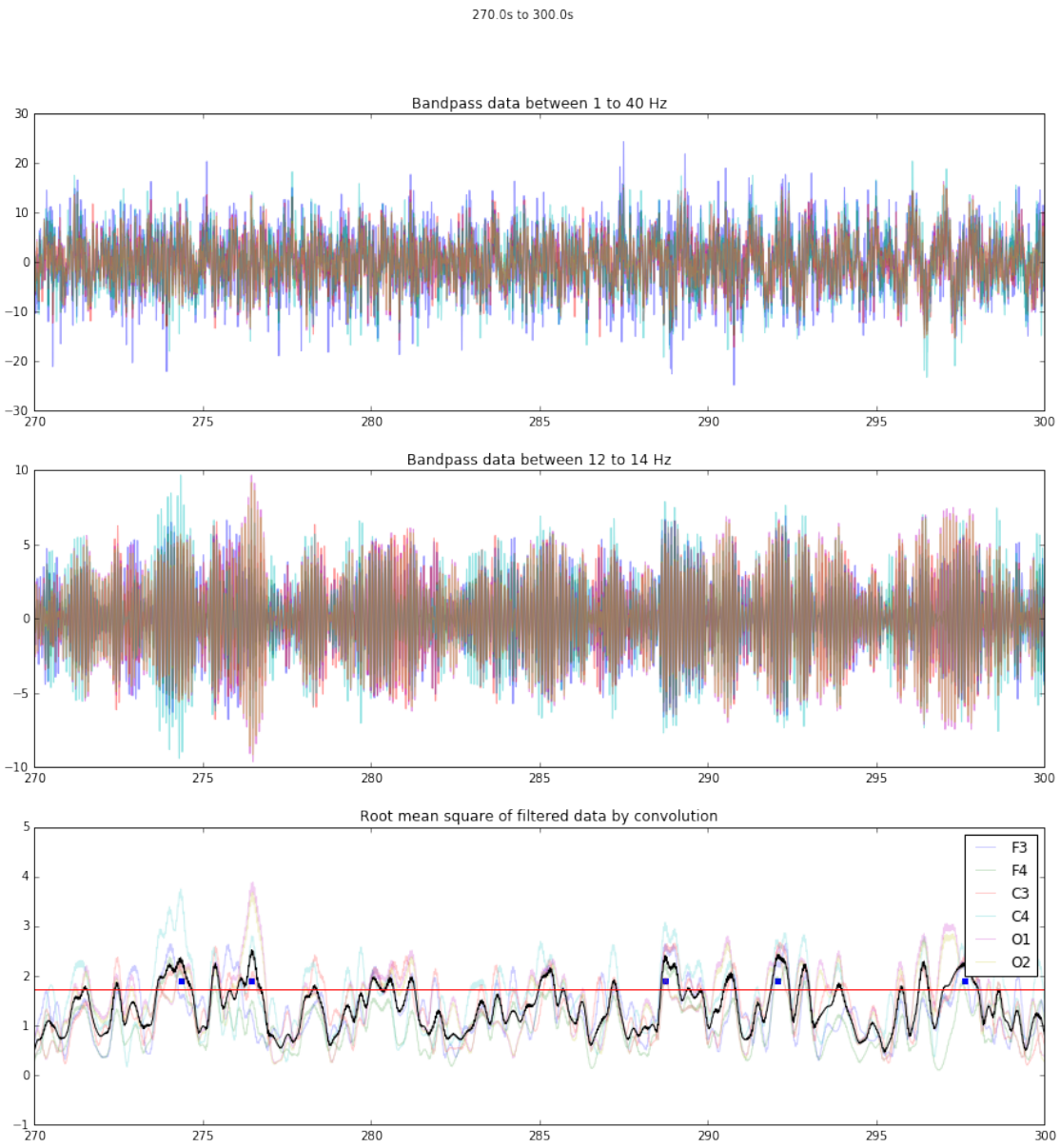
180.0s to 210.0s



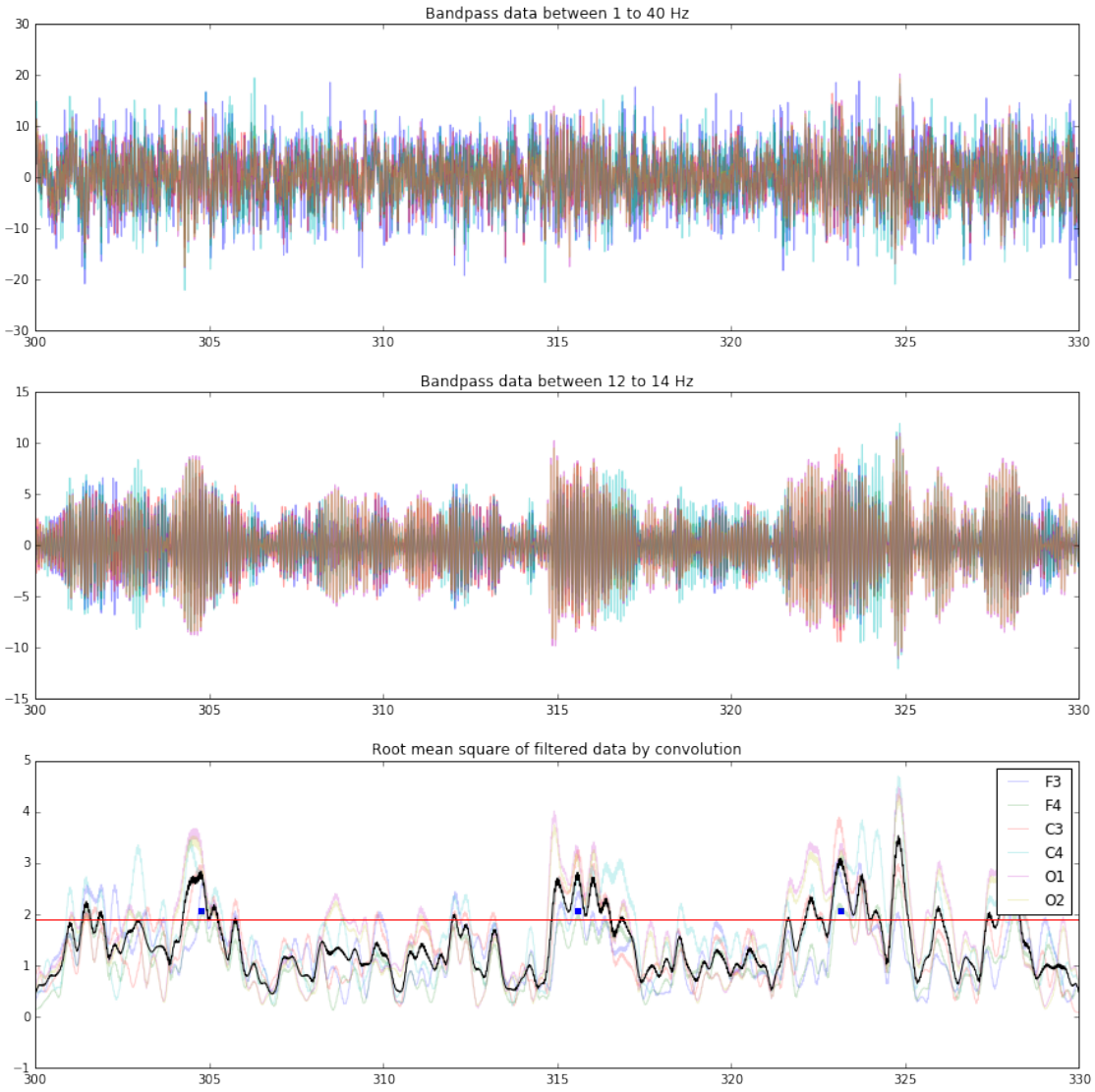

210.0s to 240.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

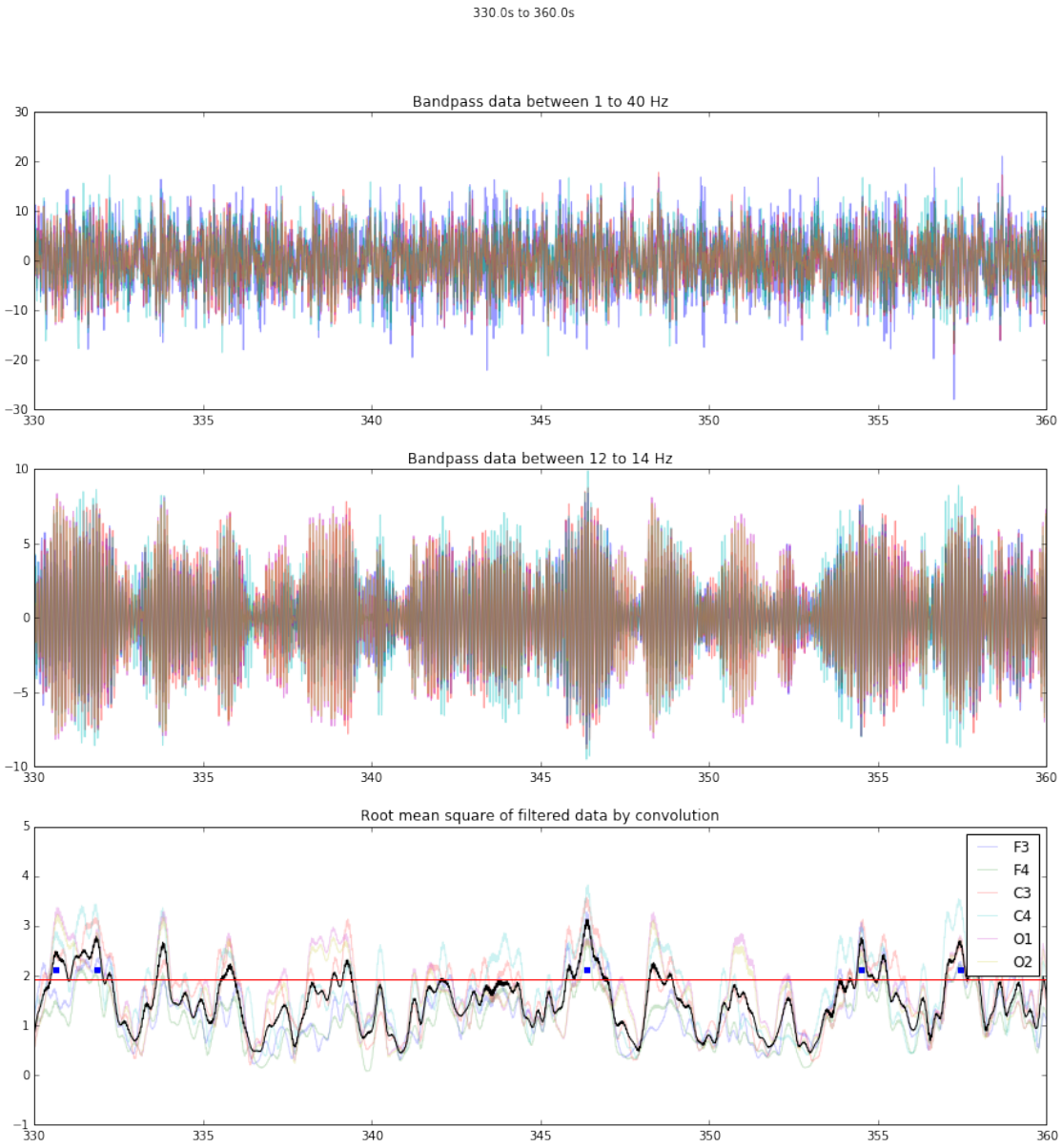
240.0s to 270.0s



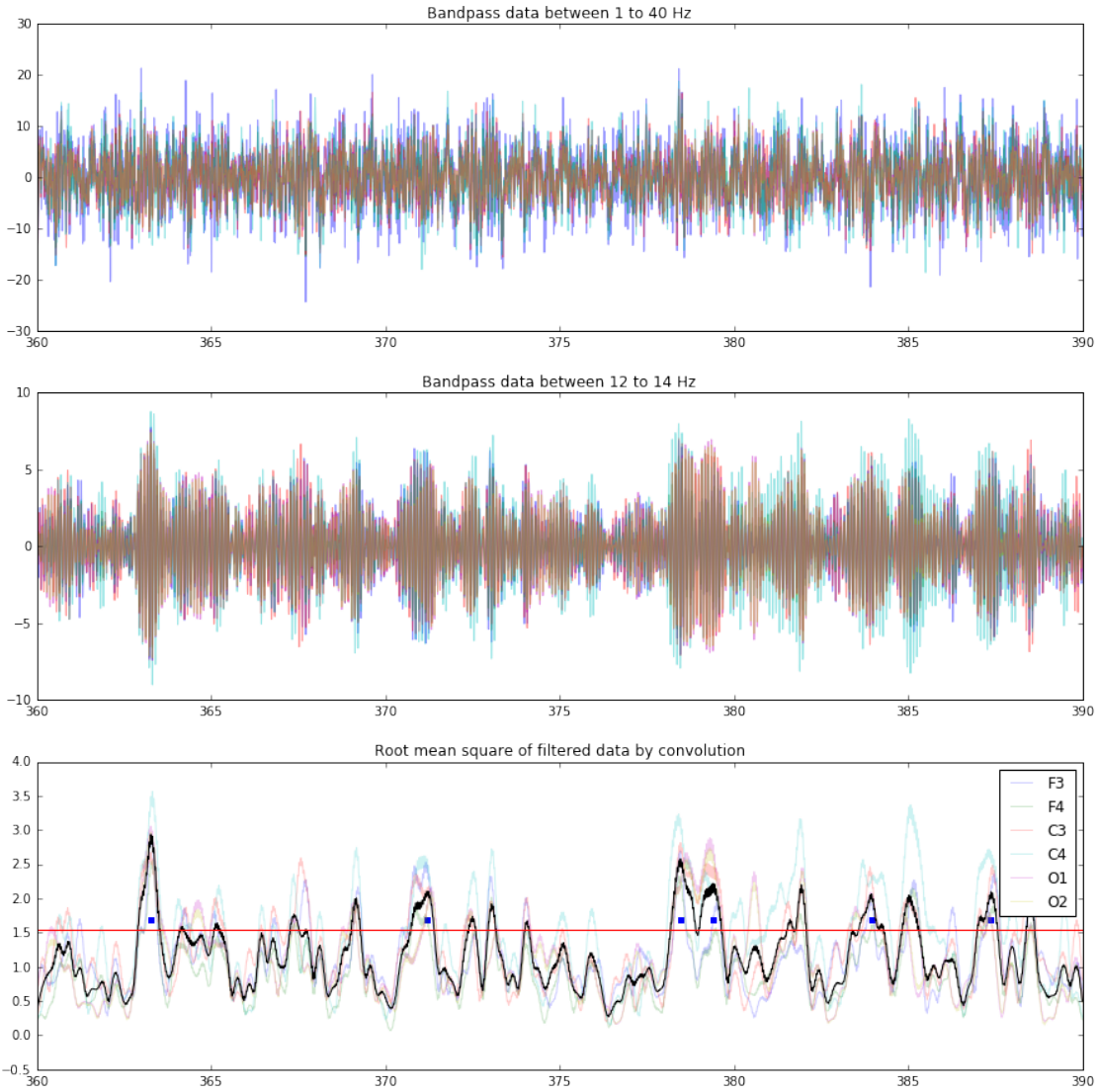

270.0s to 300.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

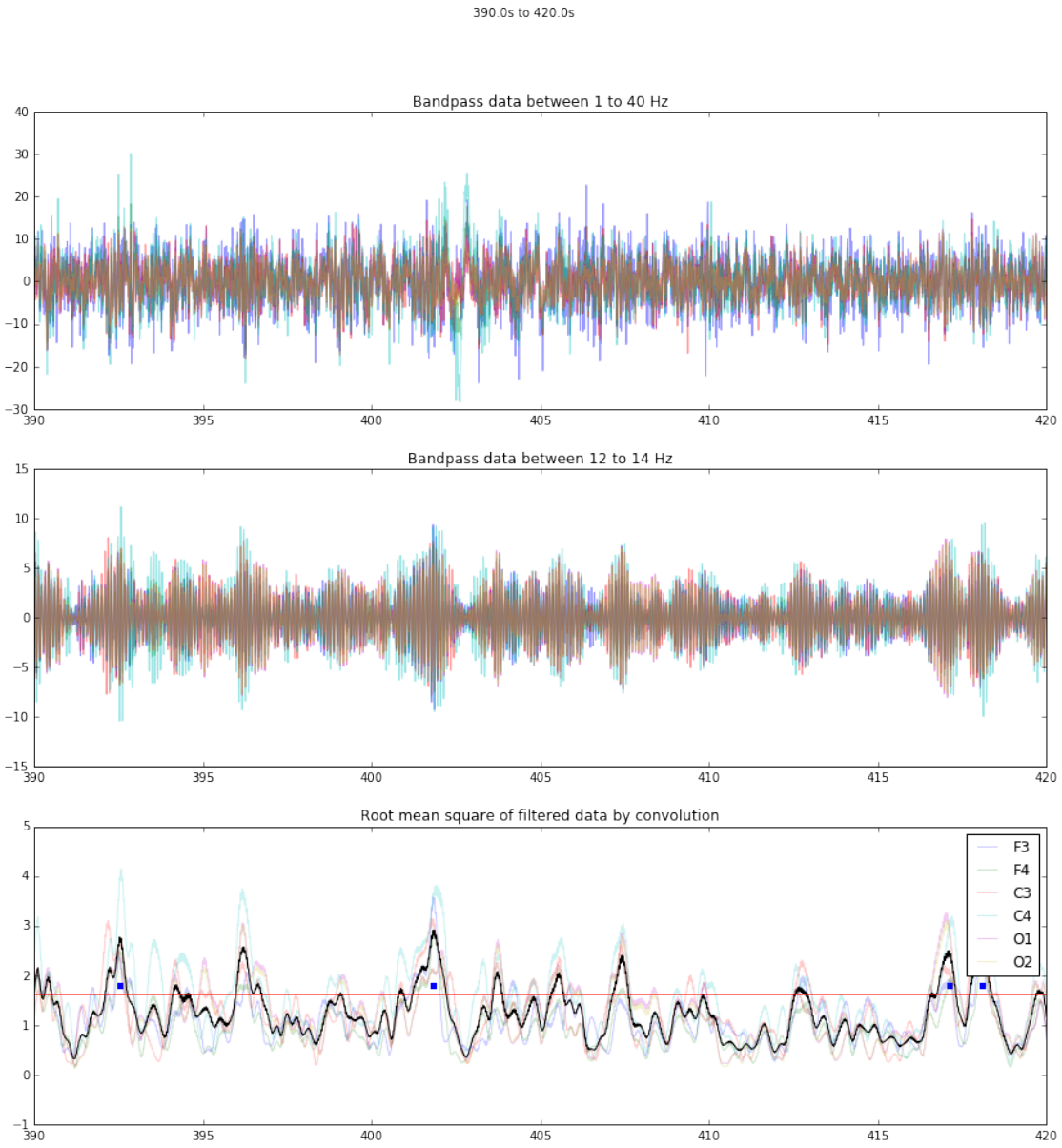
300.0s to 330.0s

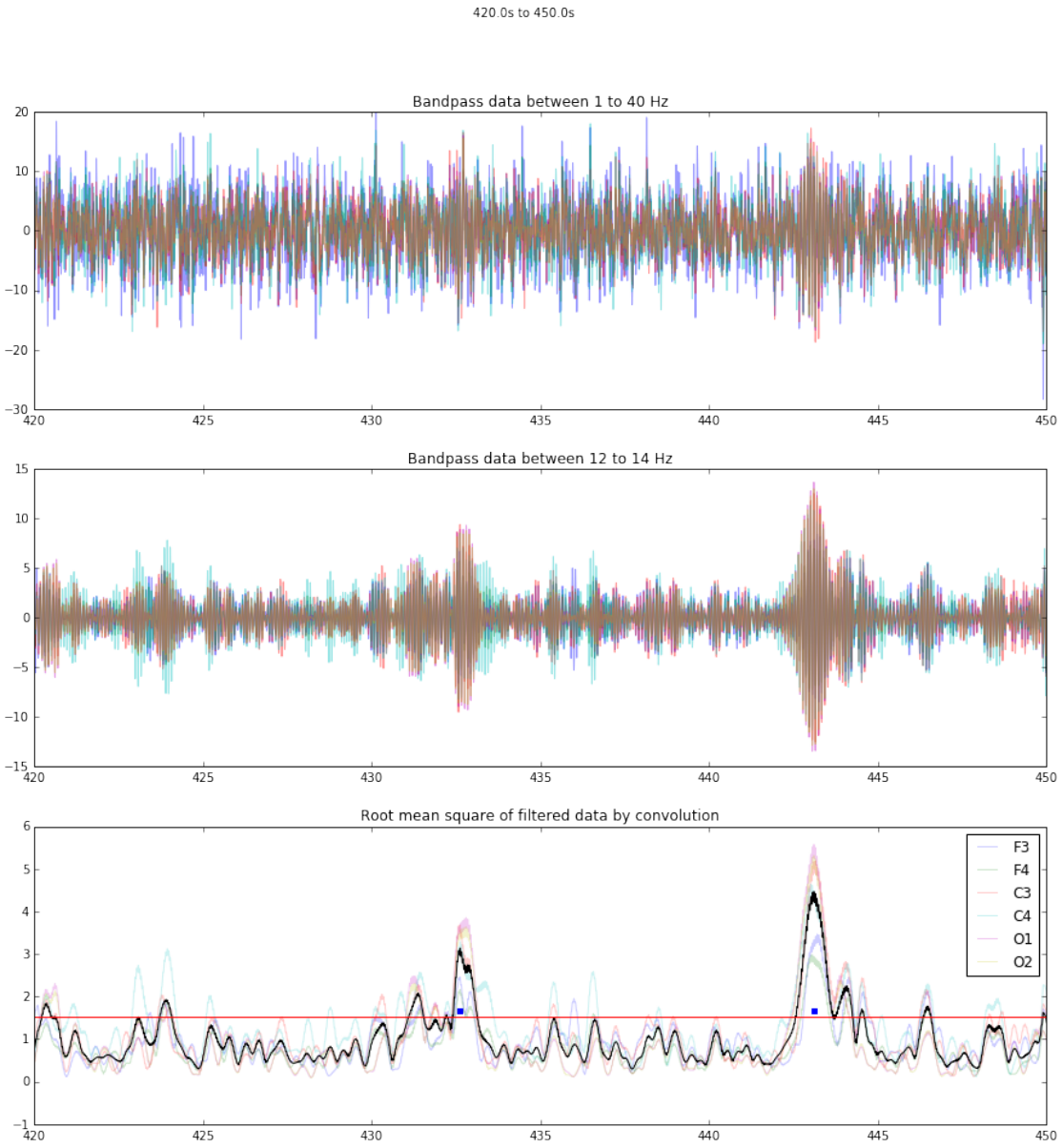


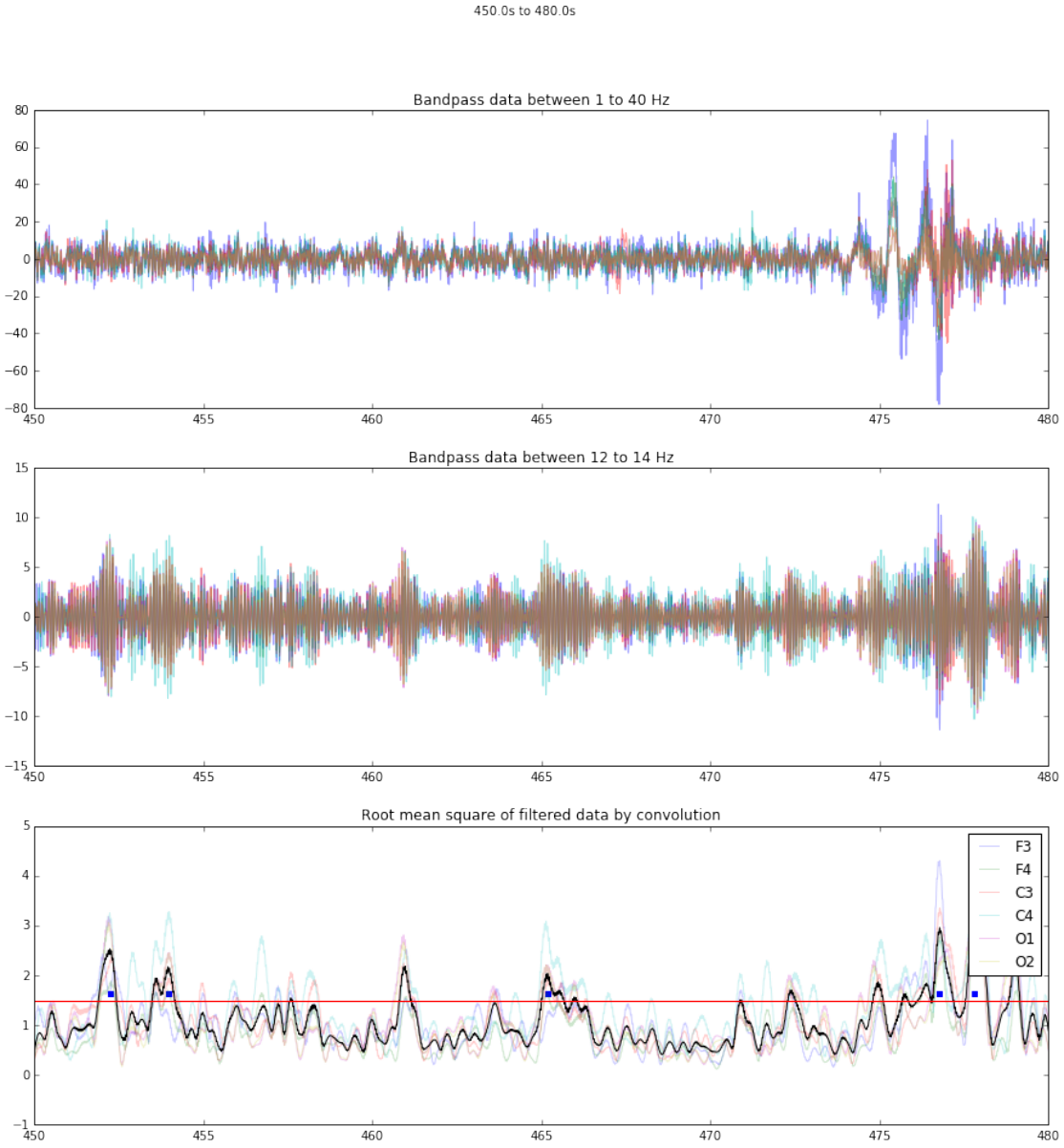

330.0s to 360.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

360.0s to 390.0s

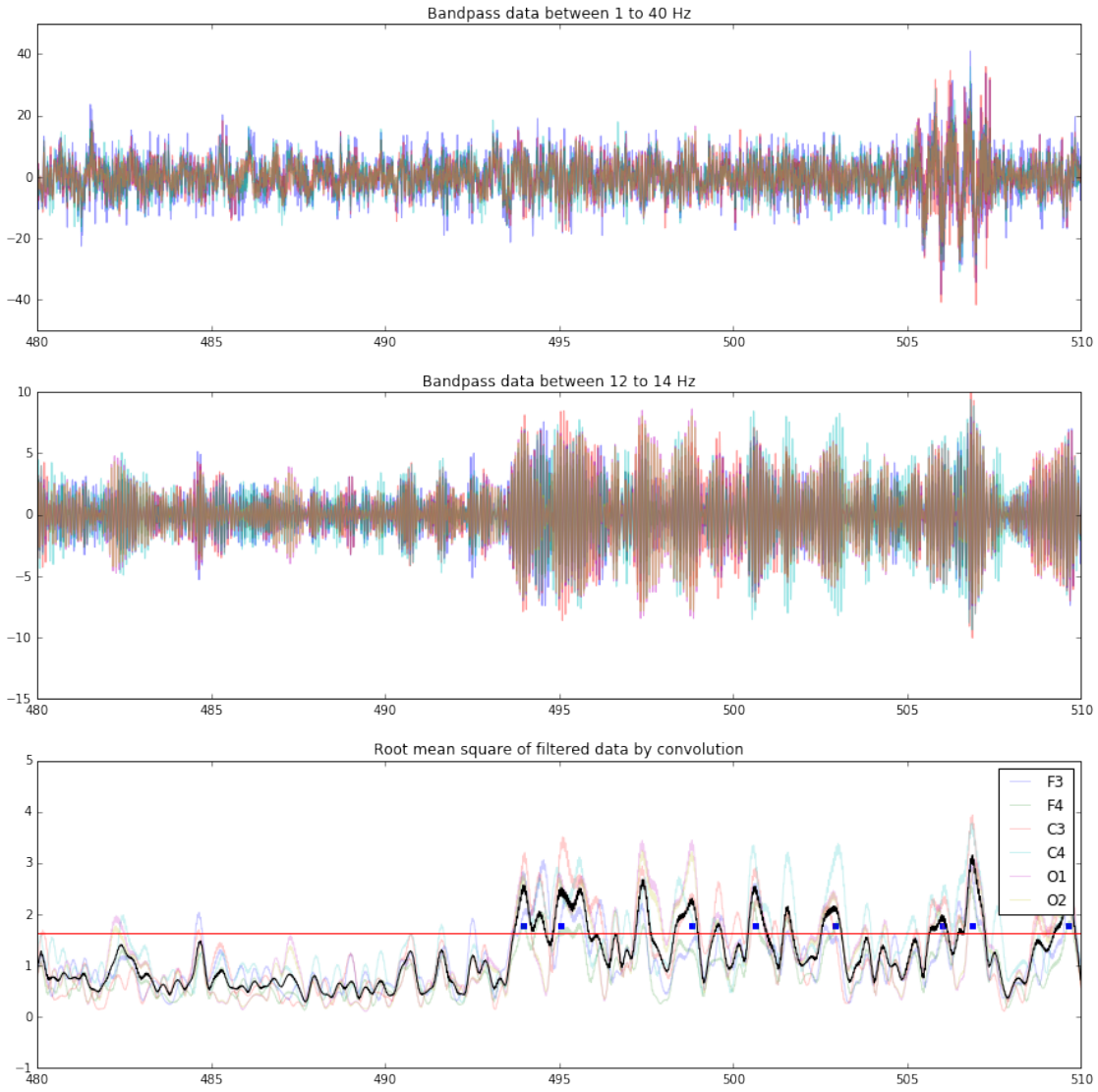


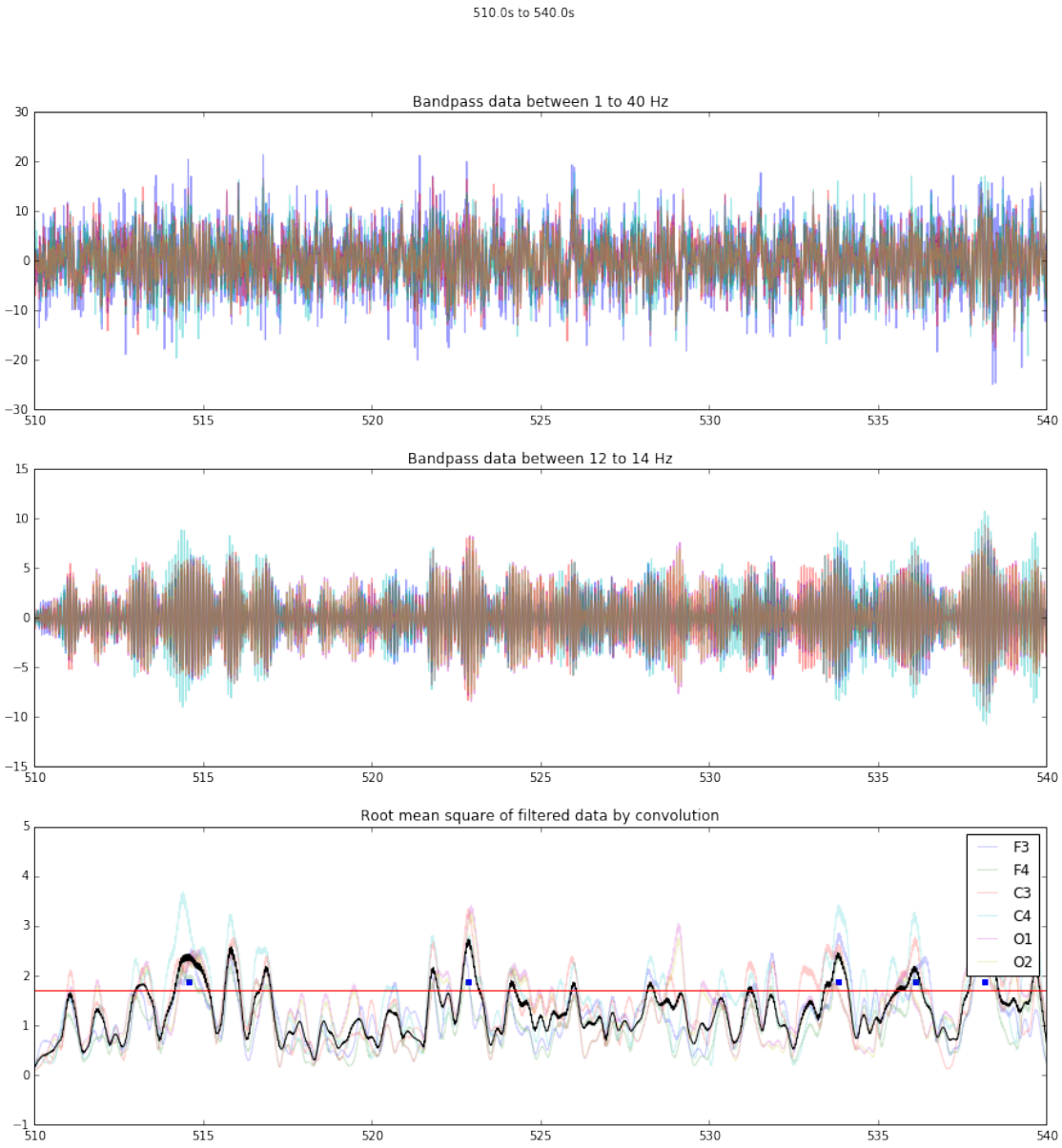

390.0s to 420.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

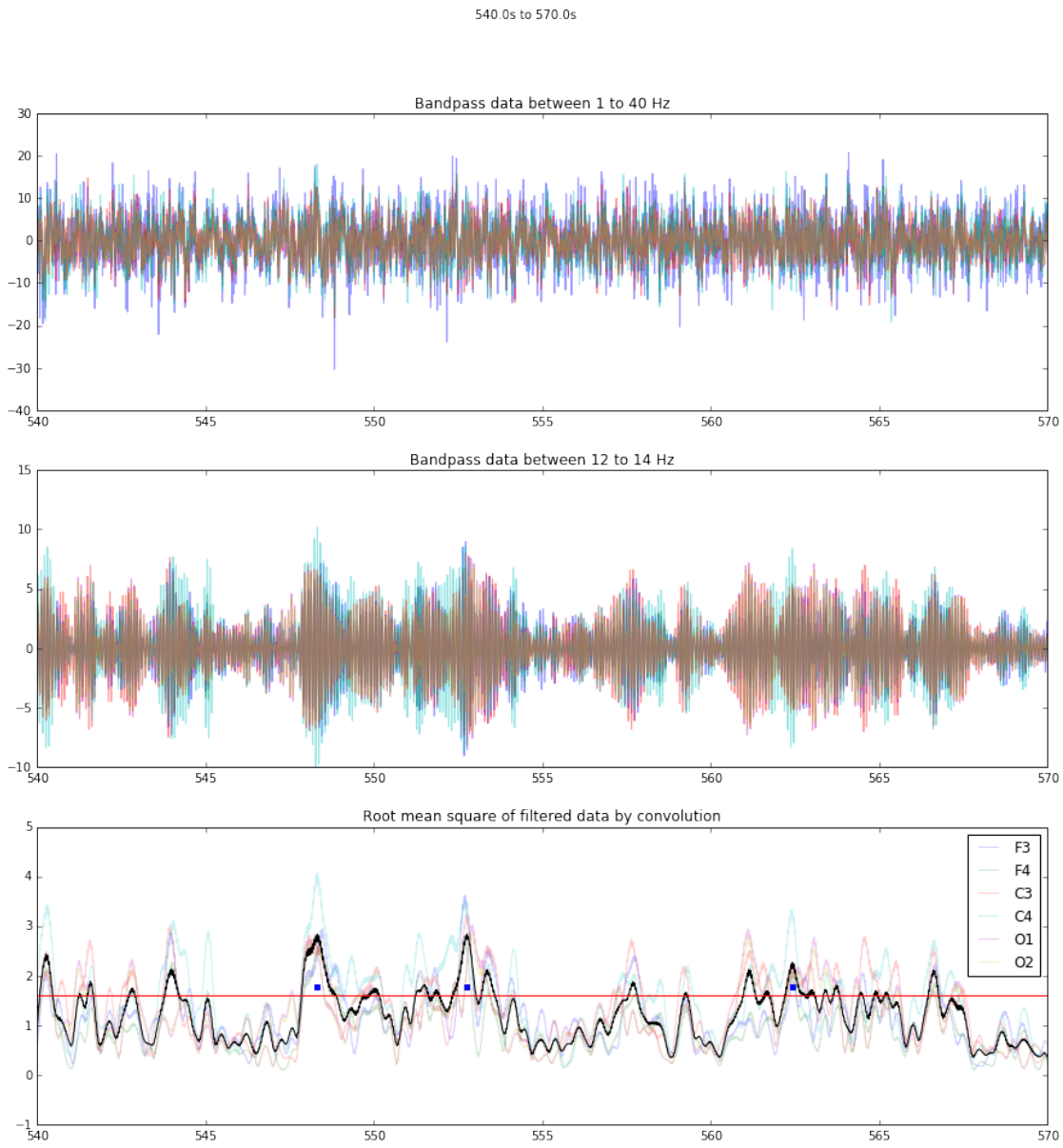

420.0s to 450.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2


450.0s to 480.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

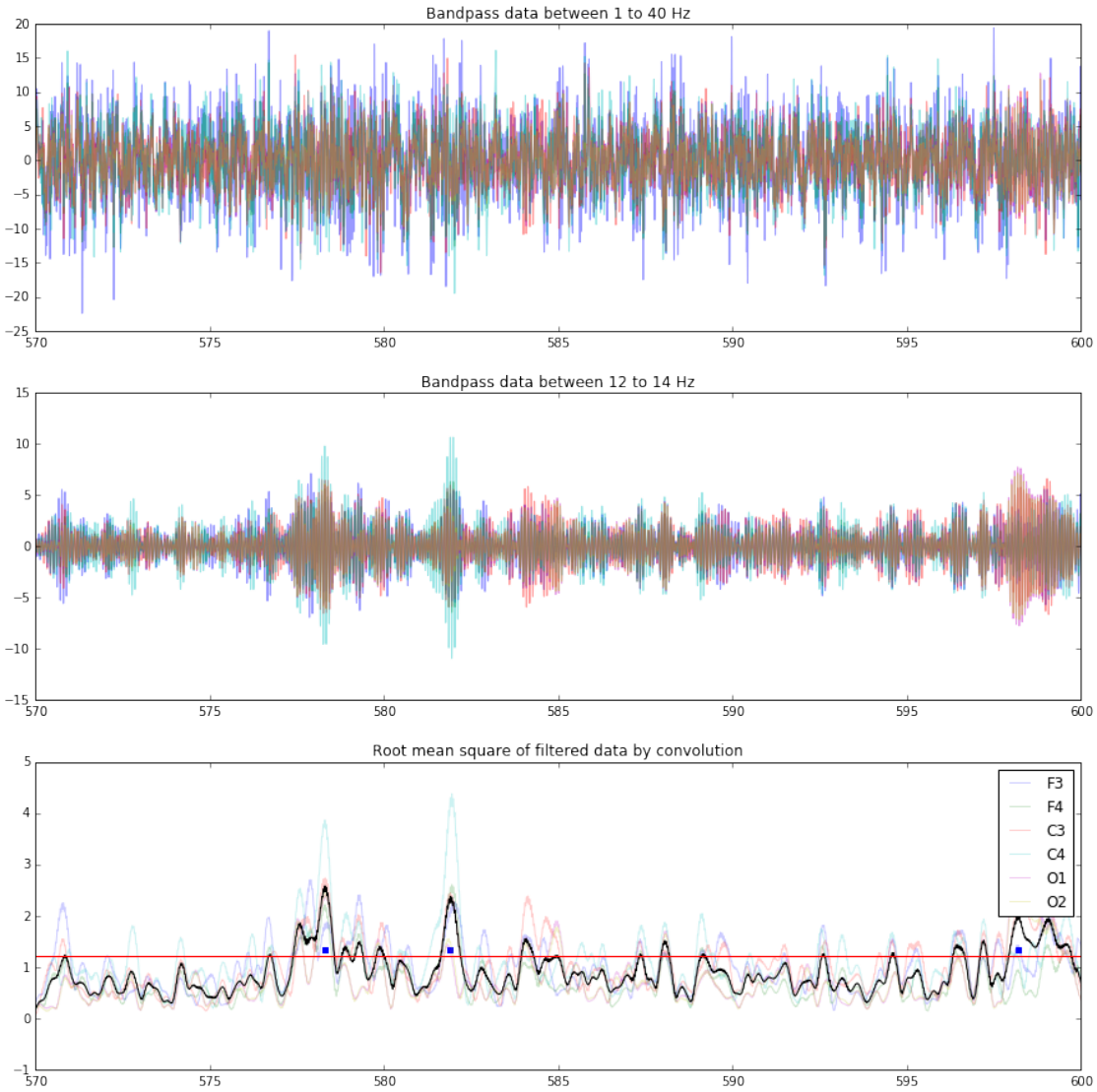
480.0s to 510.0s

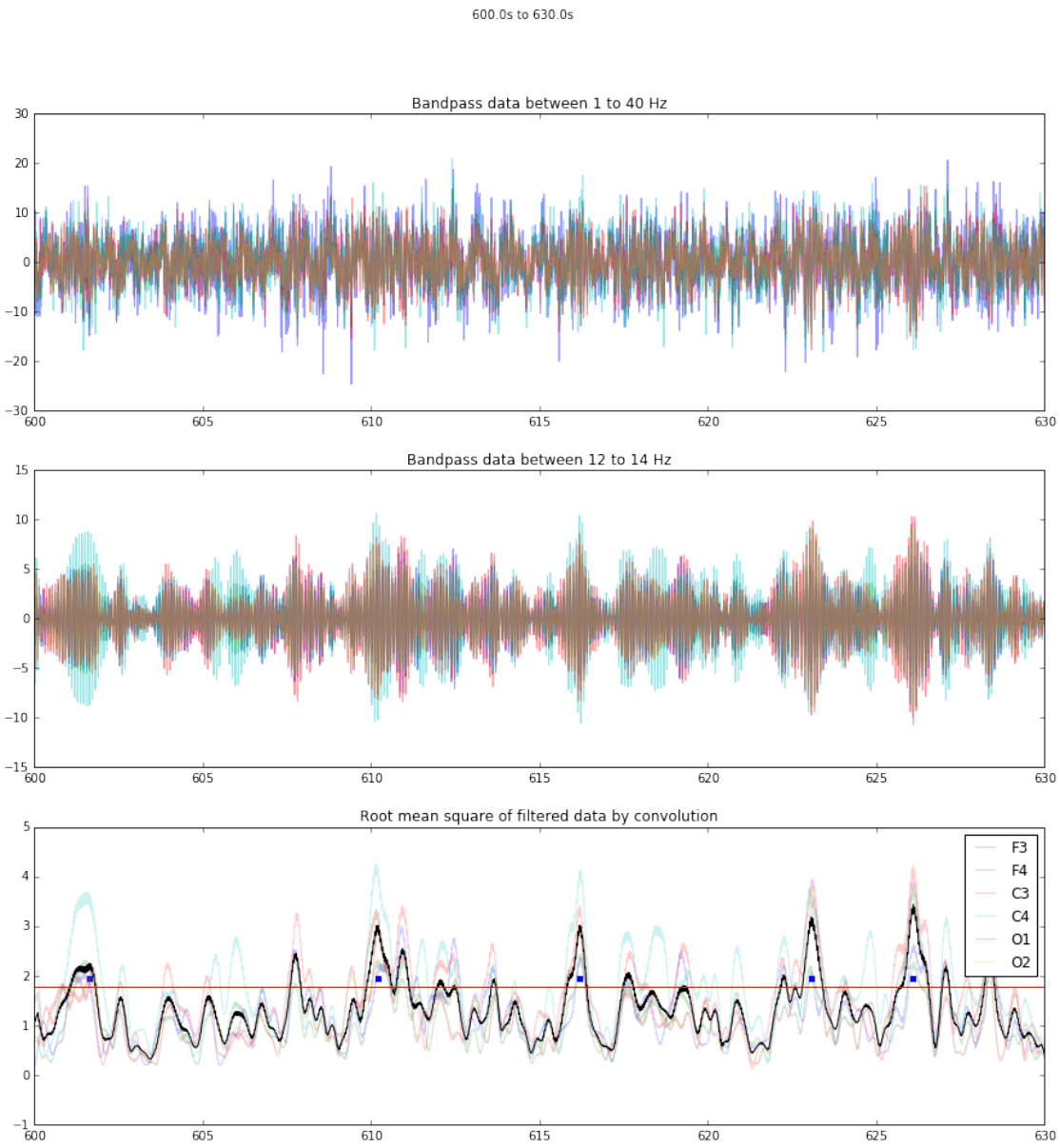



510.0s to 540.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

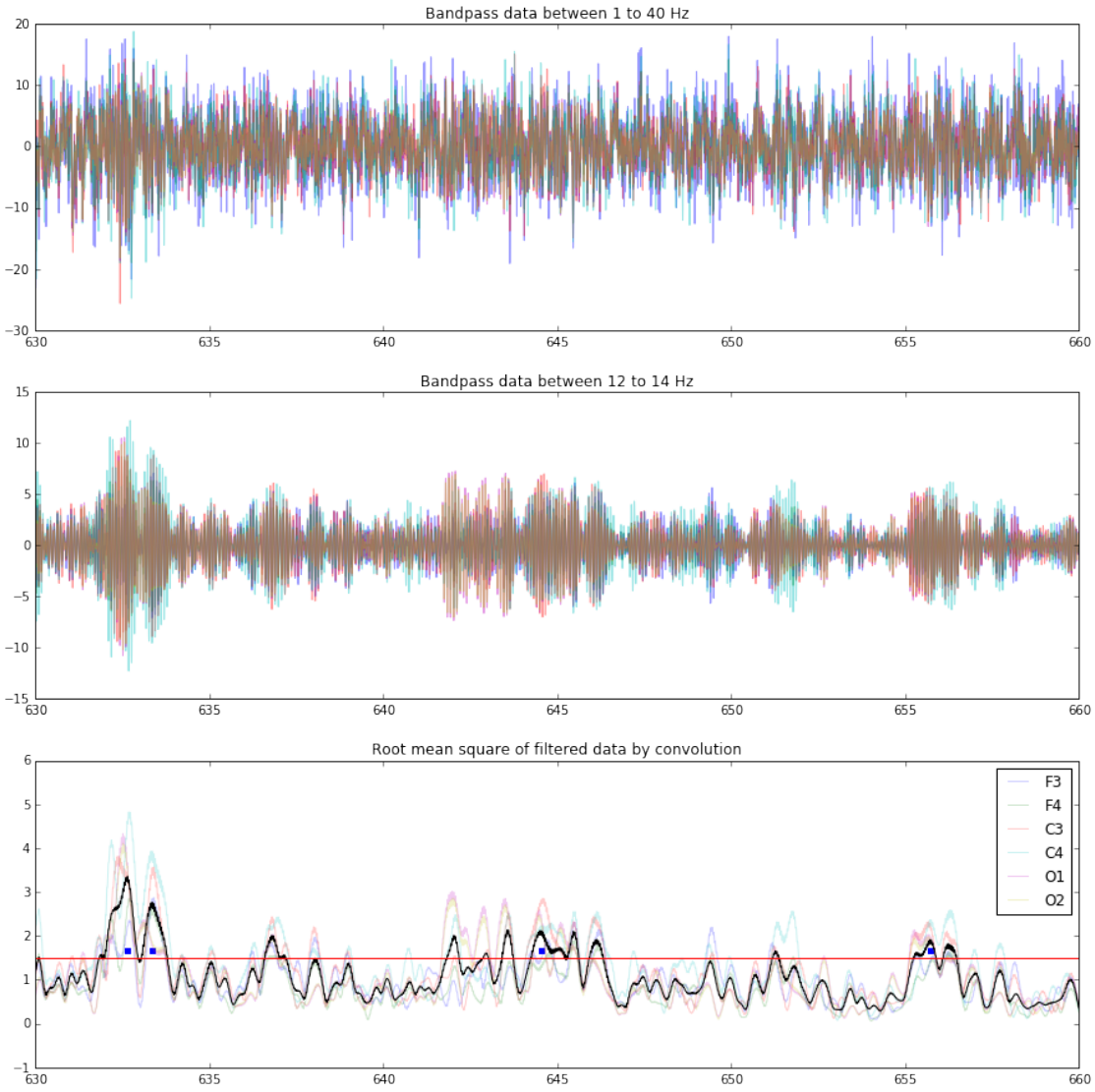

540.0s to 570.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

570.0s to 600.0s

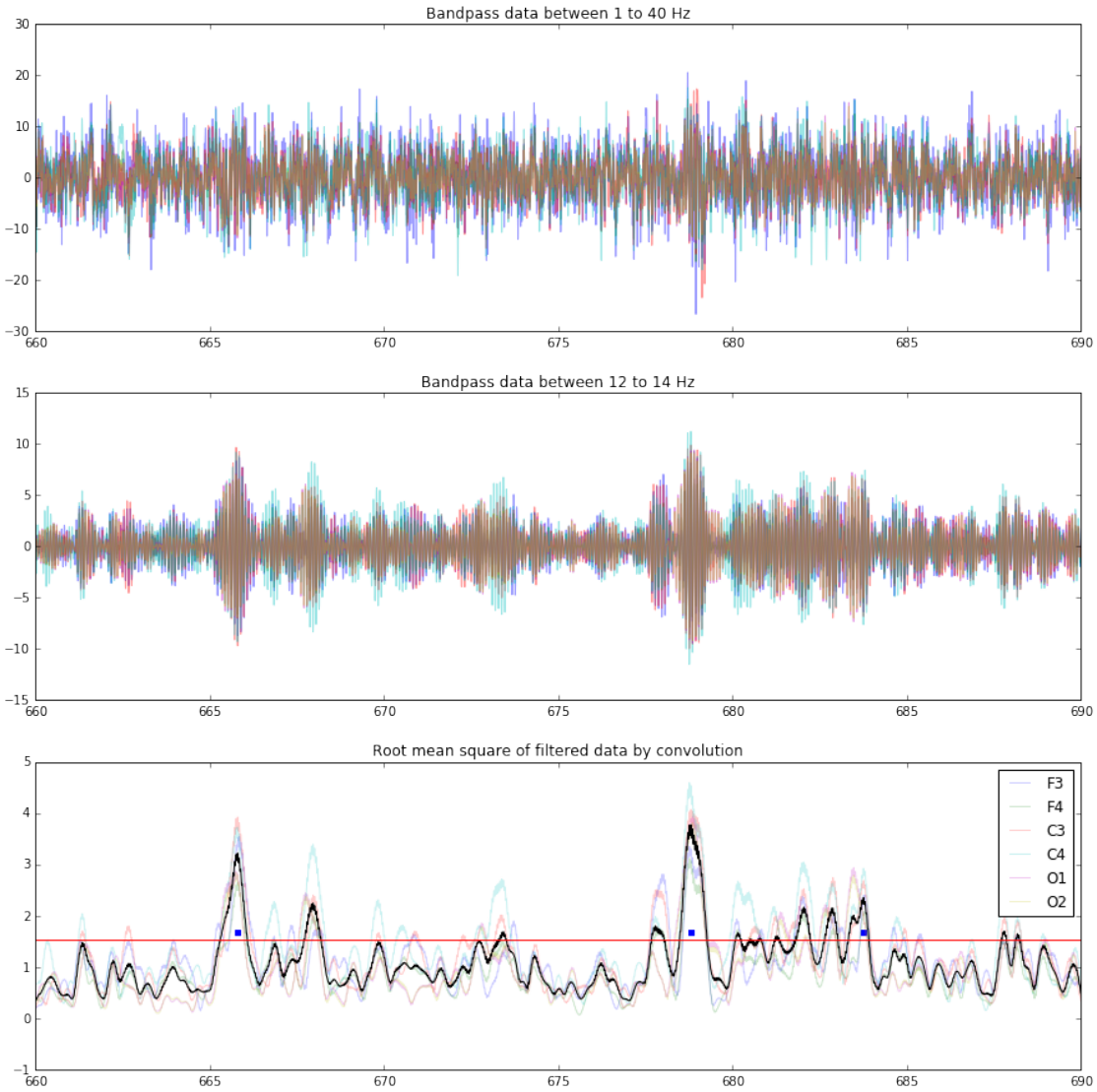



600.0s to 630.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

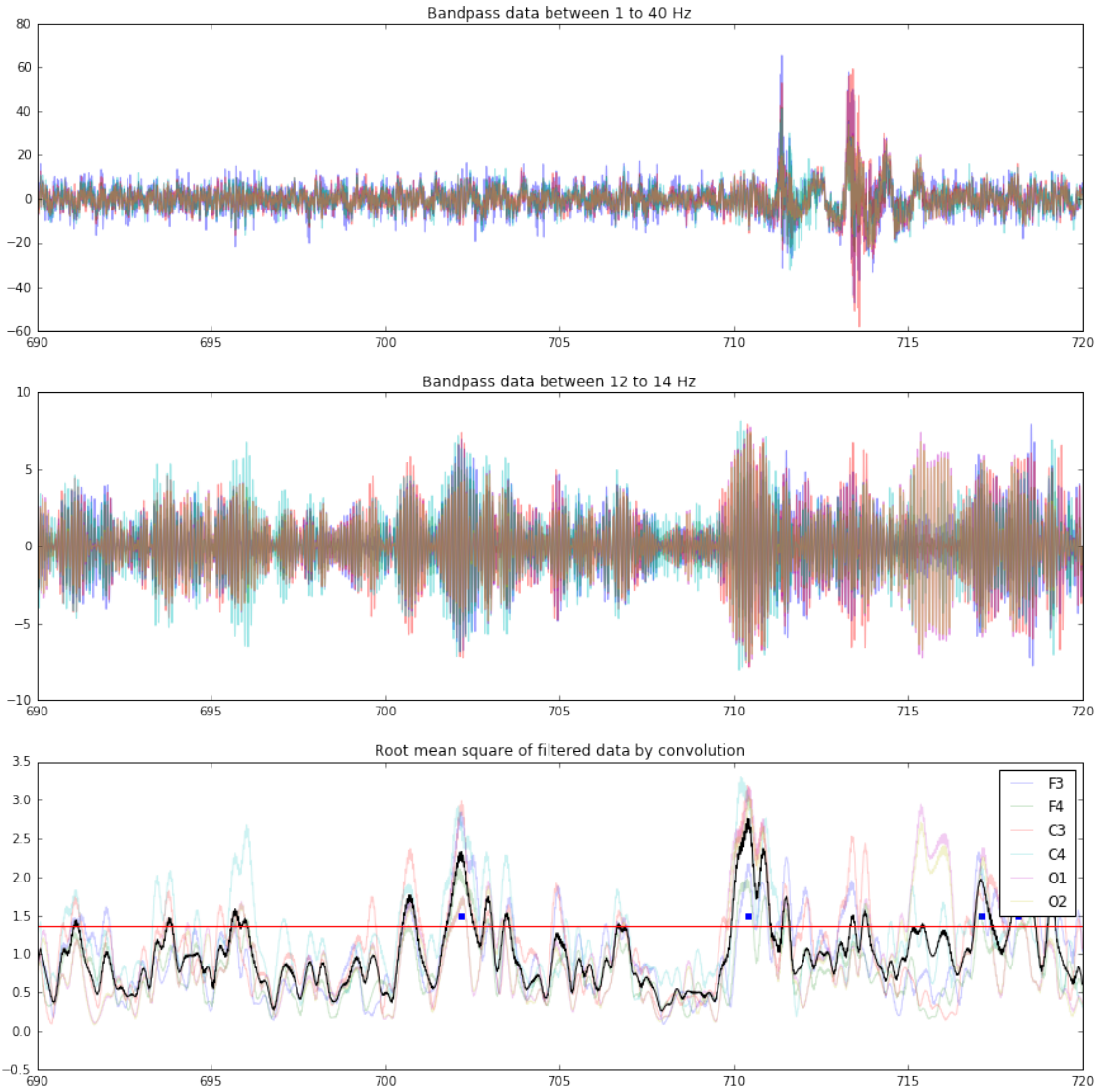
630.0s to 660.0s



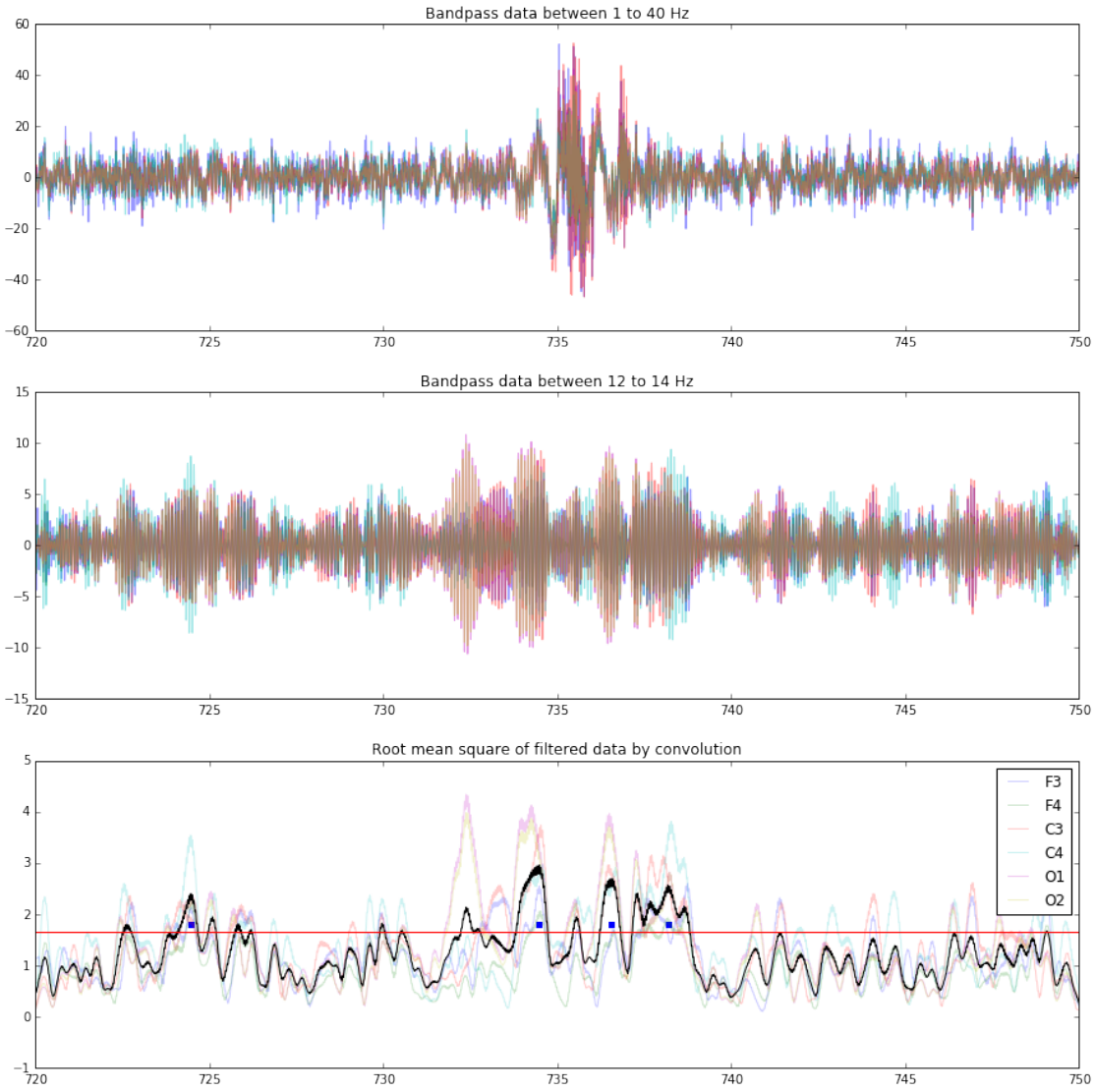
660.0s to 690.0s

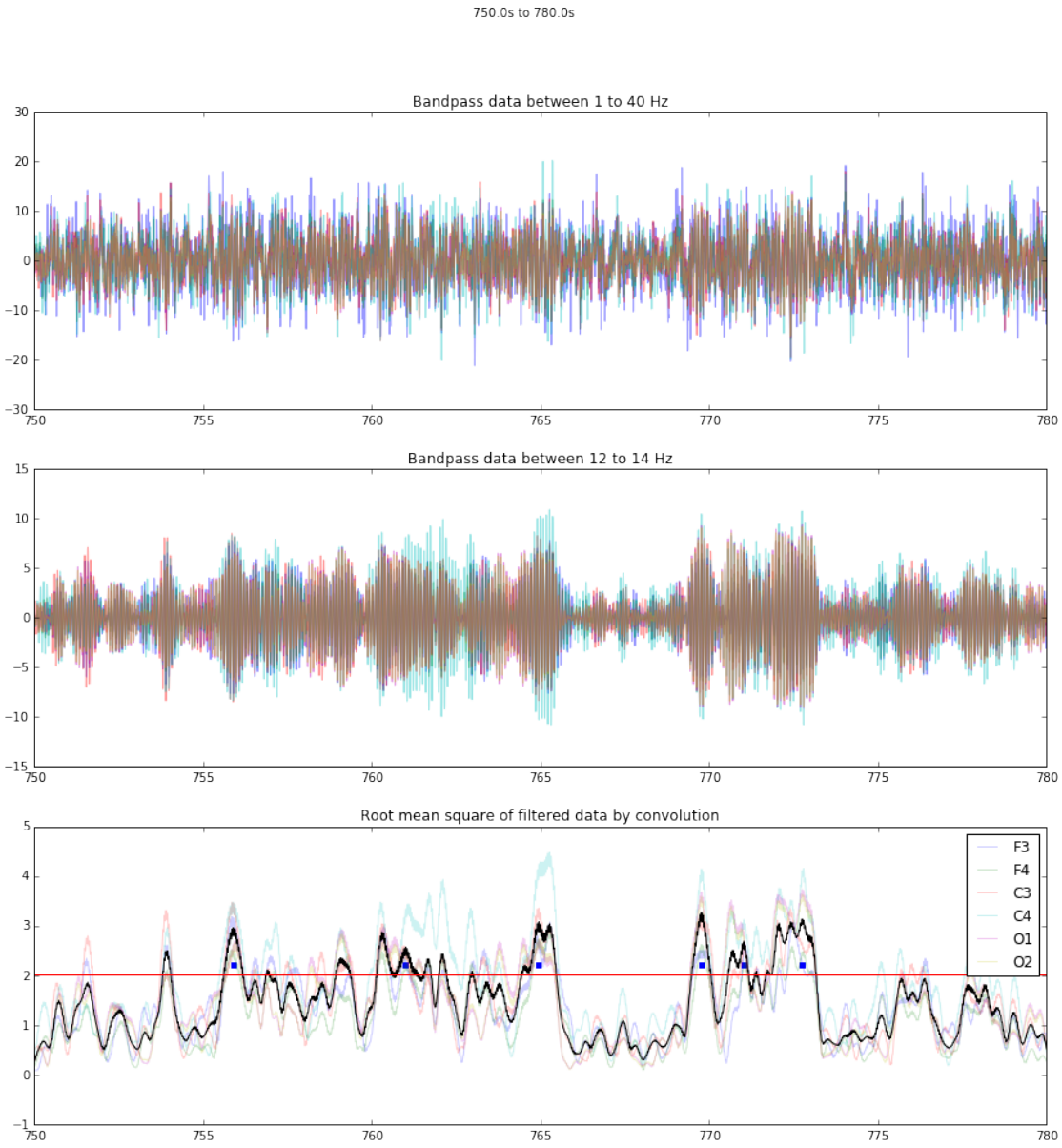


690.0s to 720.0s

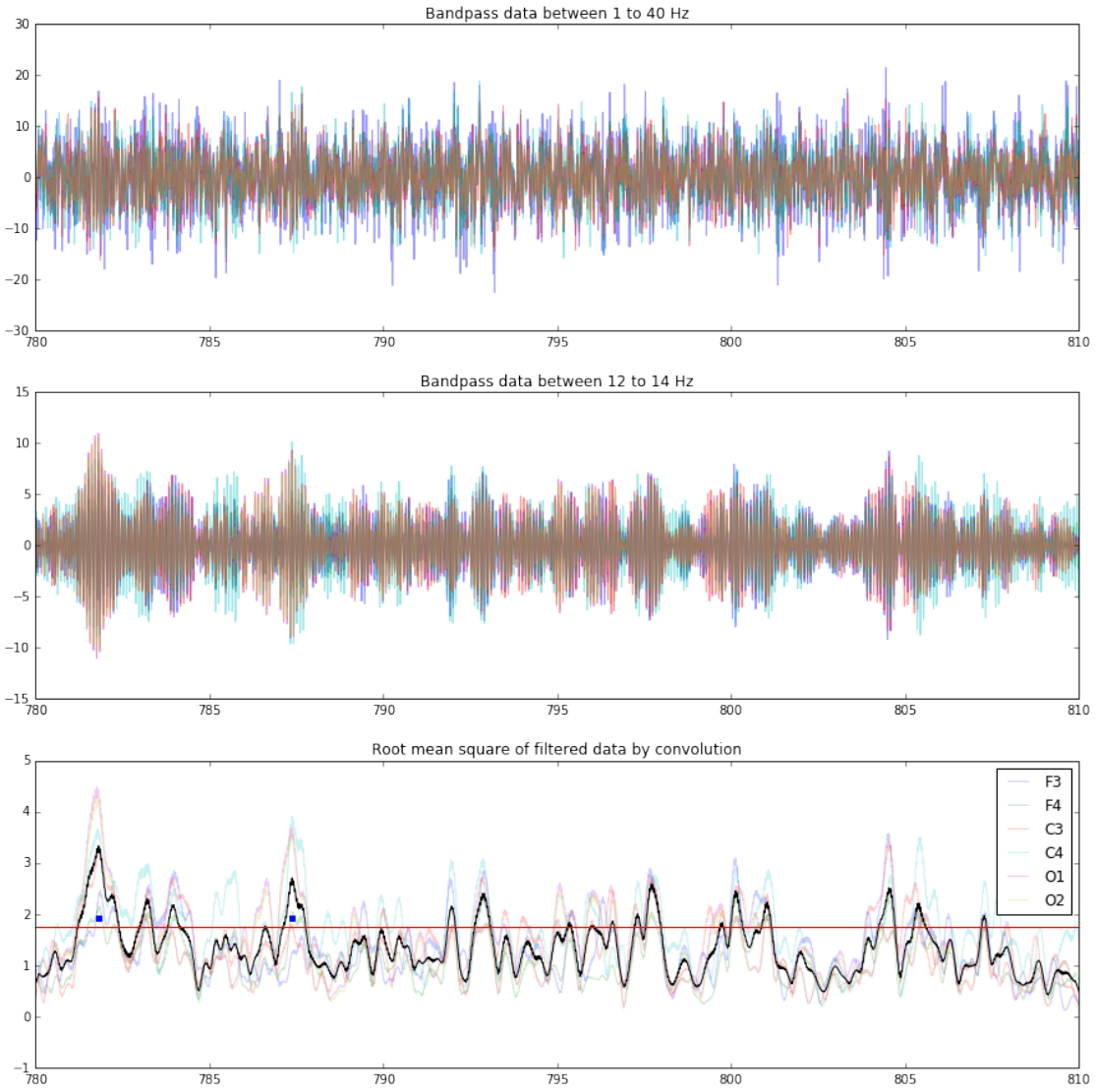


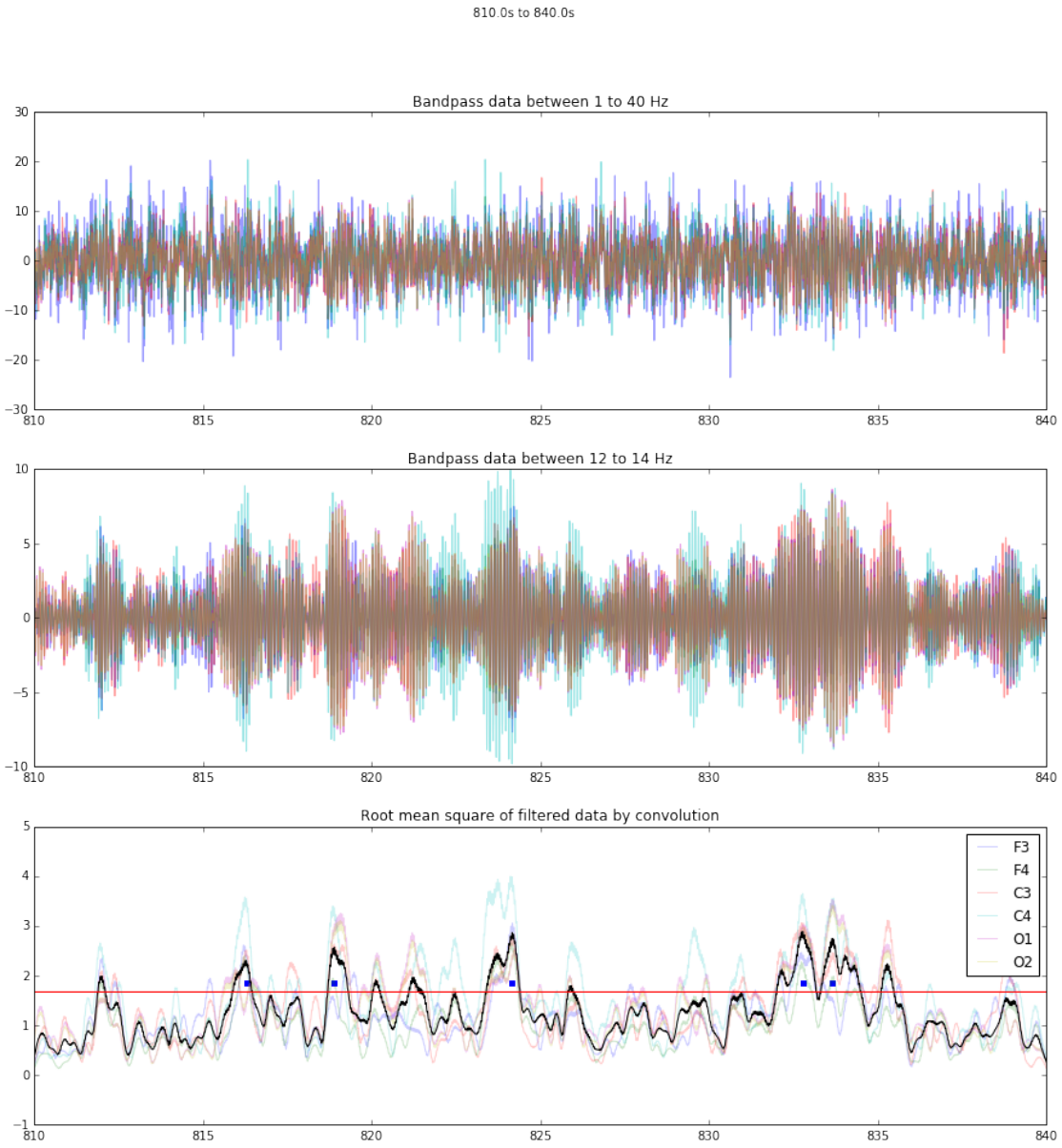
720.0s to 750.0s



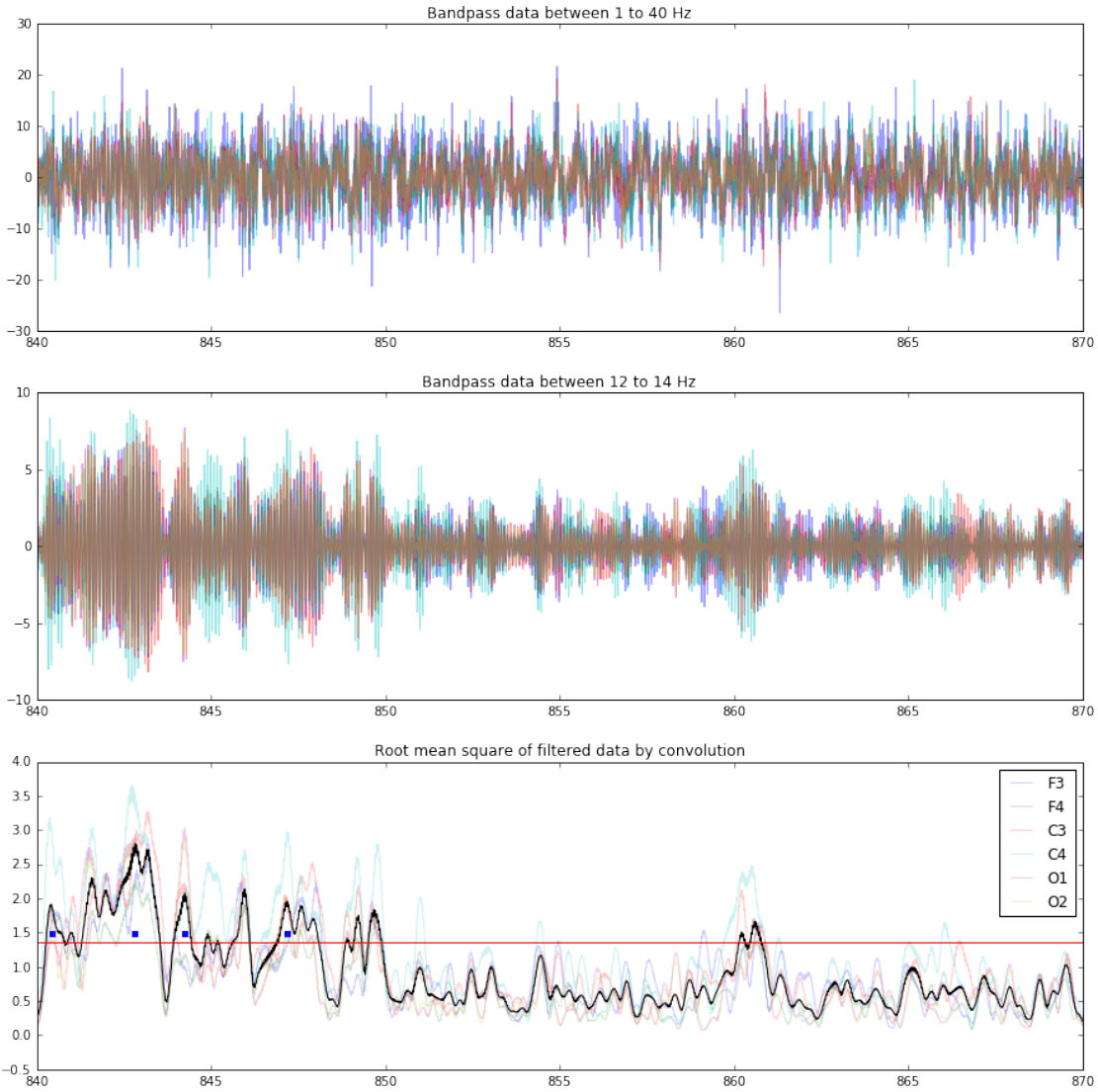

750.0s to 780.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

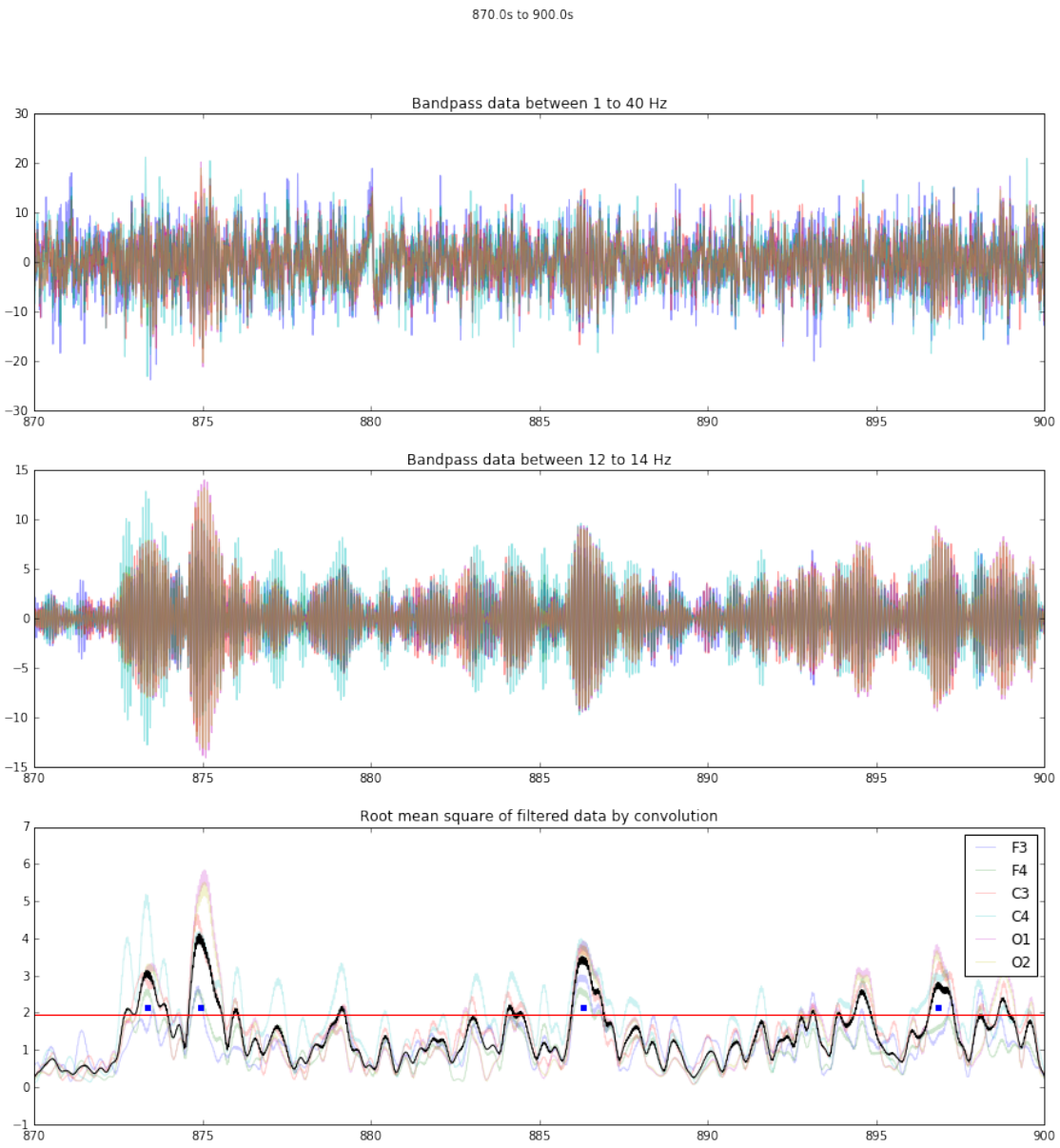
780.0s to 810.0s



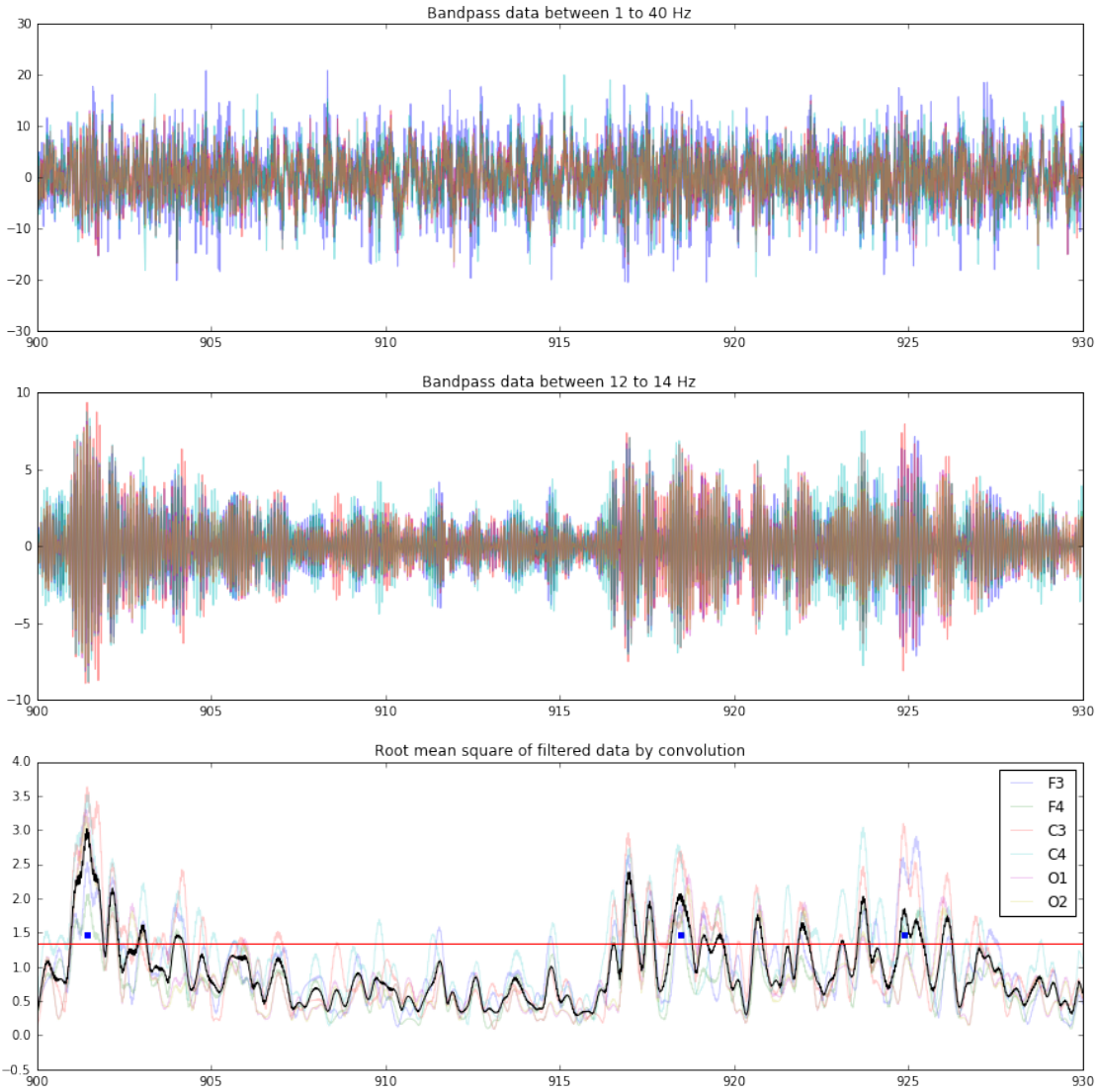

810.0s to 840.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

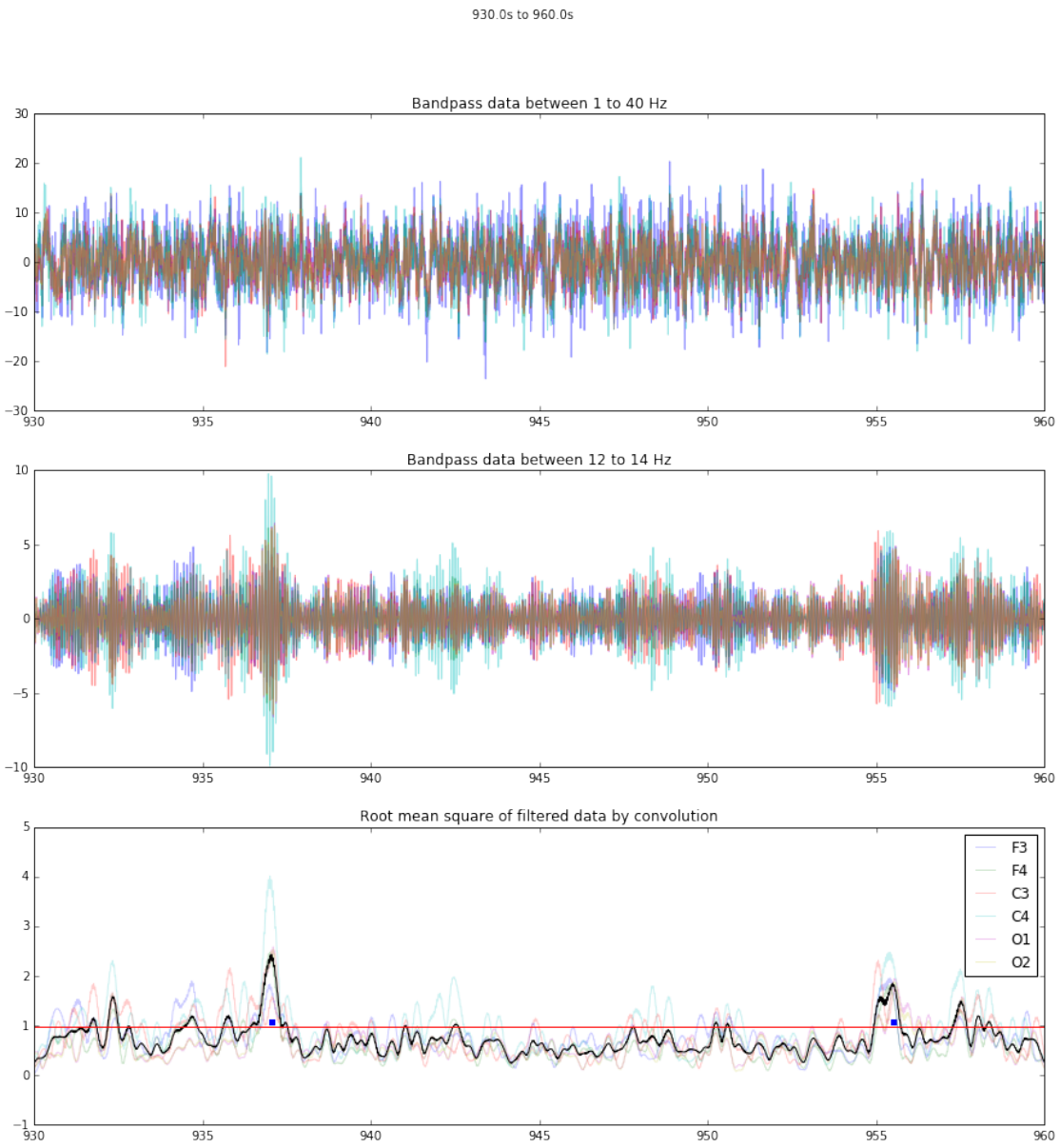
840.0s to 870.0s



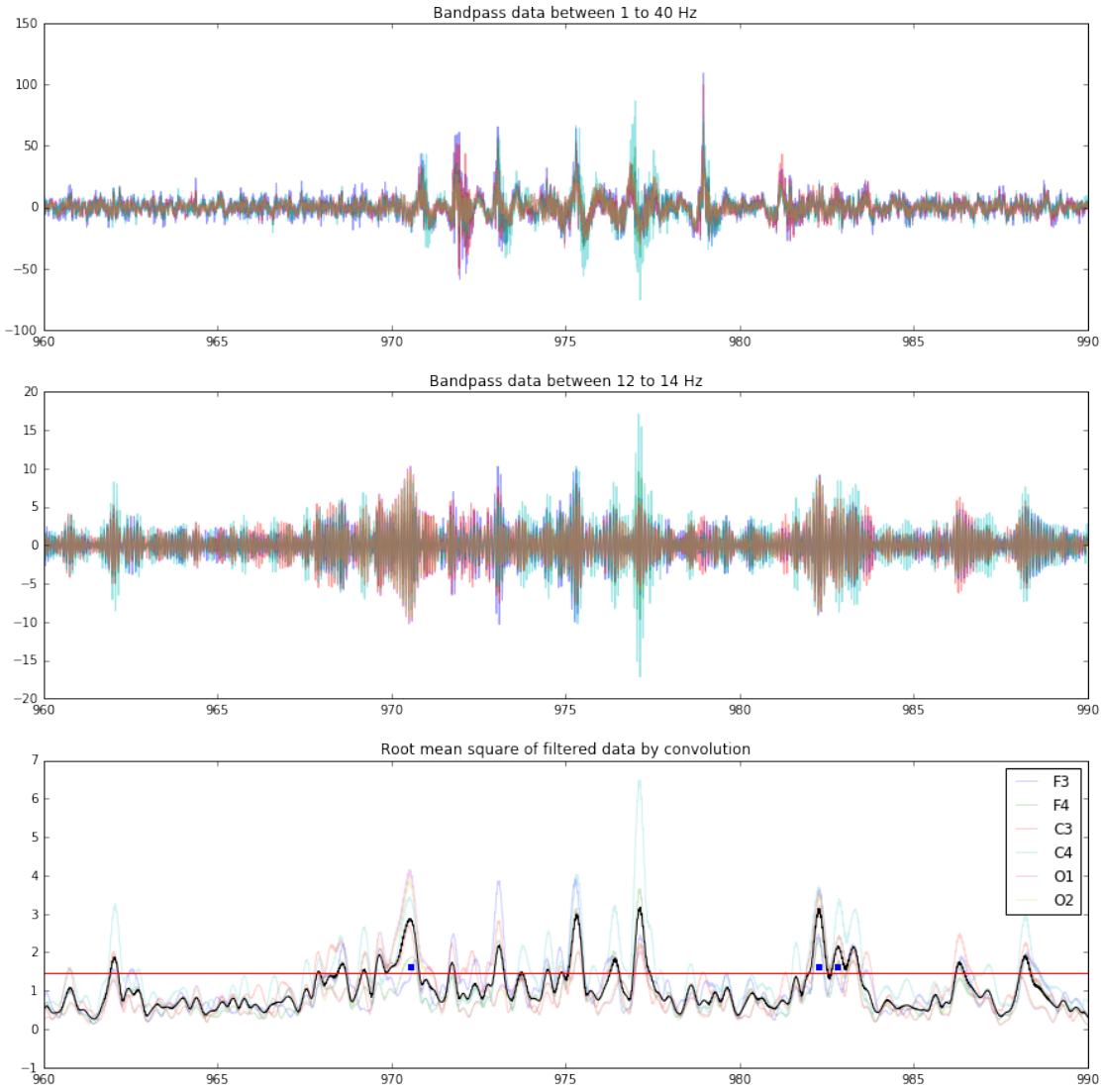

870.0s to 900.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

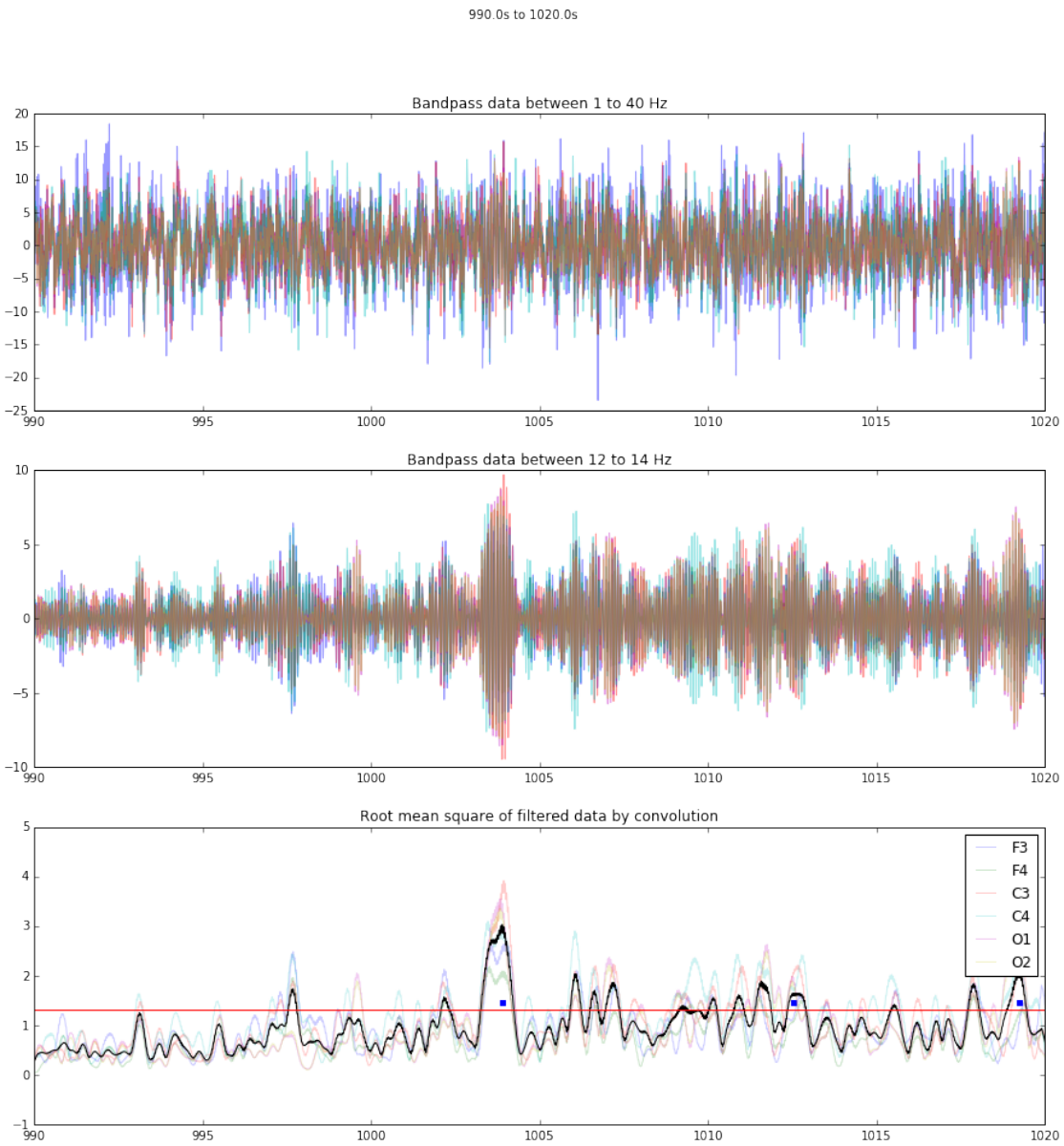
900.0s to 930.0s

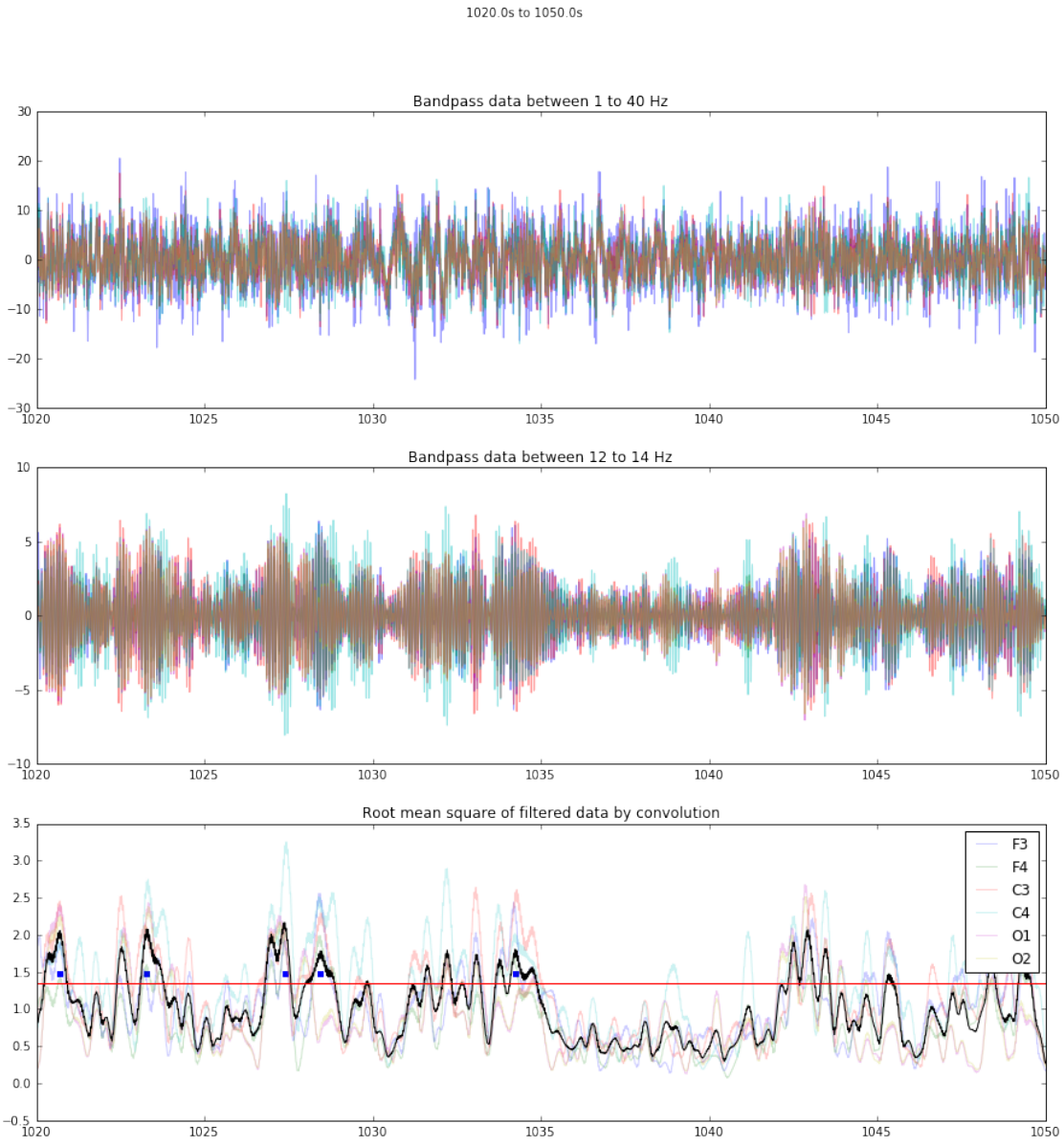


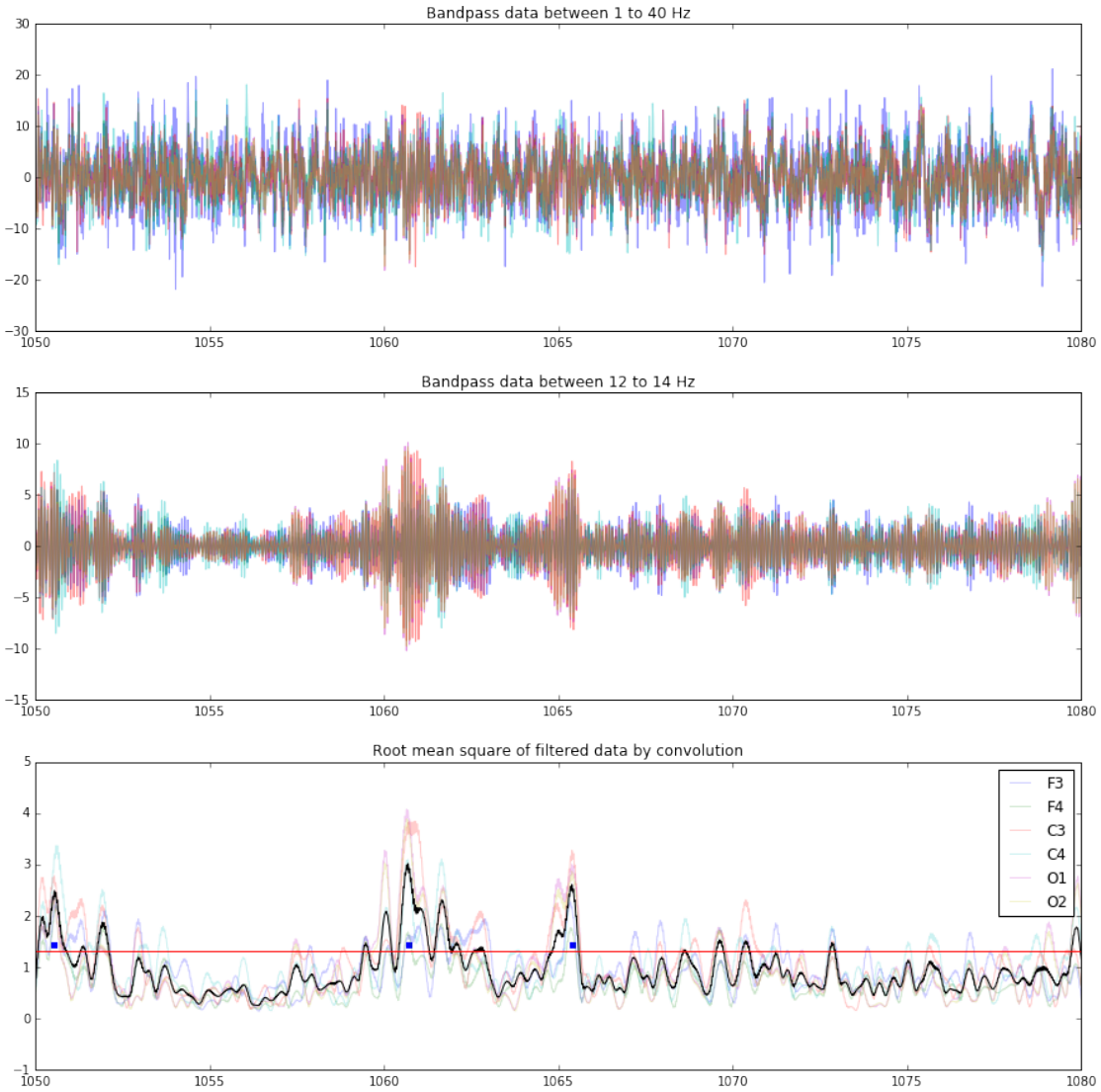

930.0s to 960.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

960.0s to 990.0s

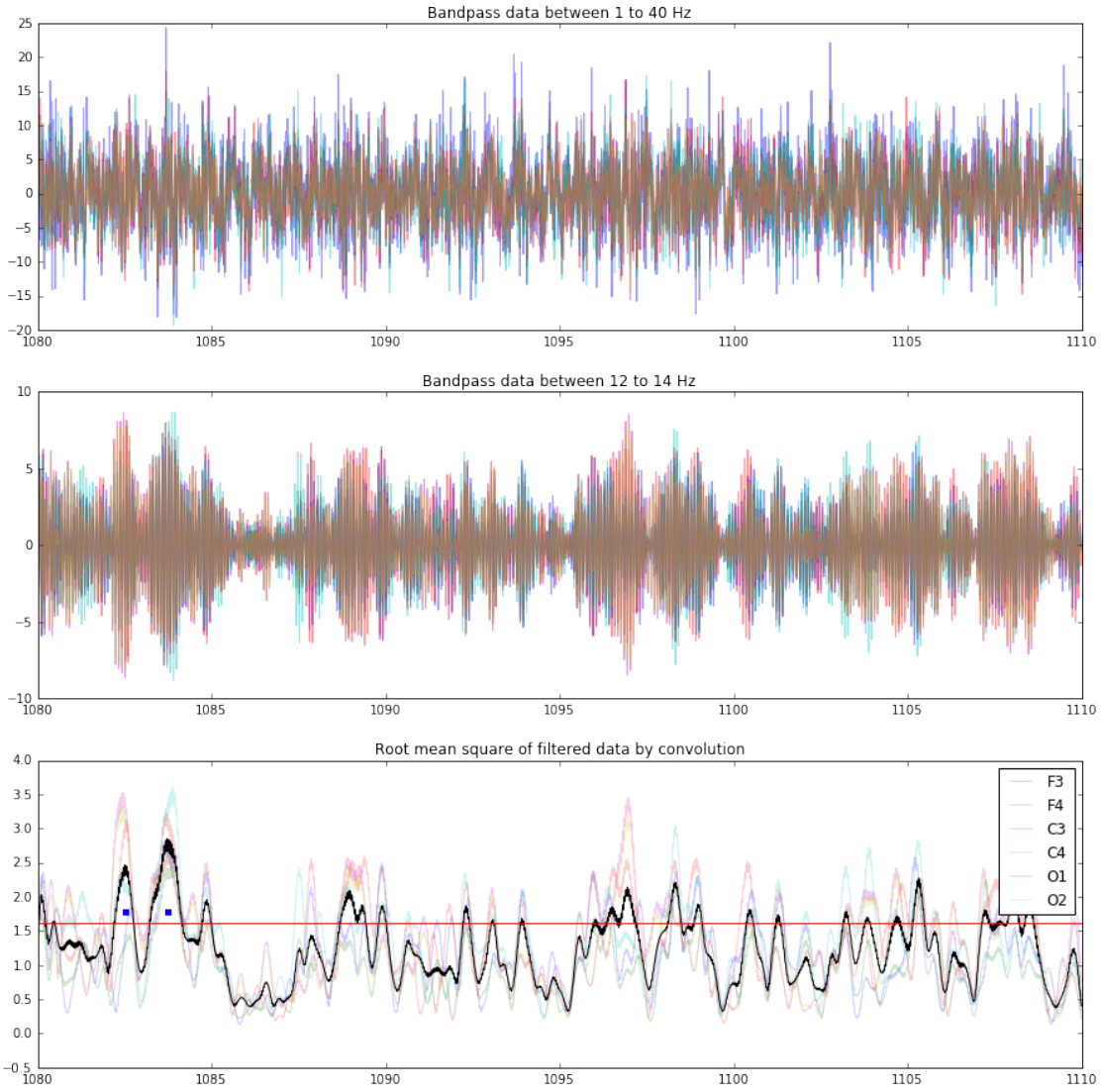



990.0s to 1020.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

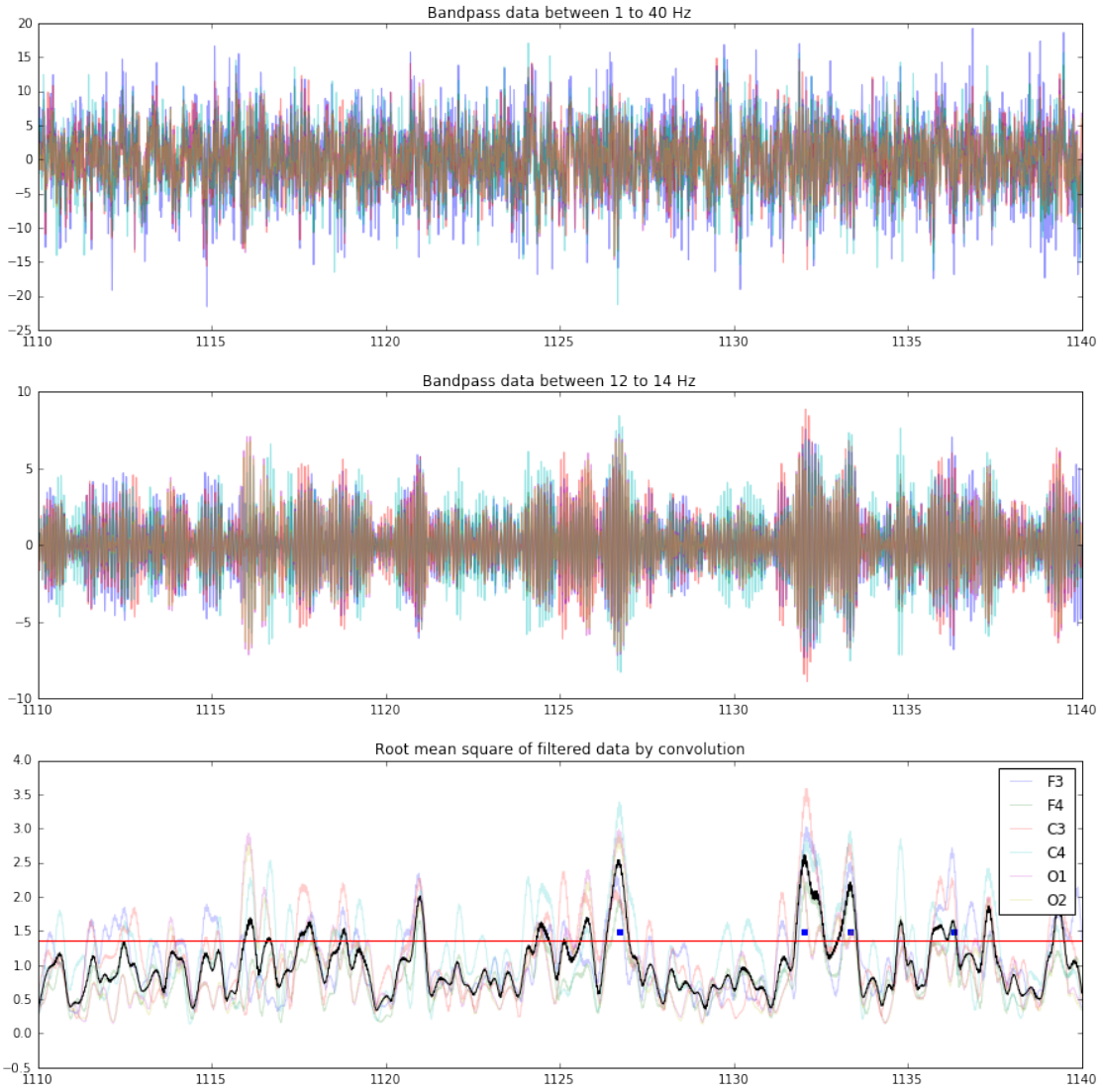

1020.0s to 1050.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

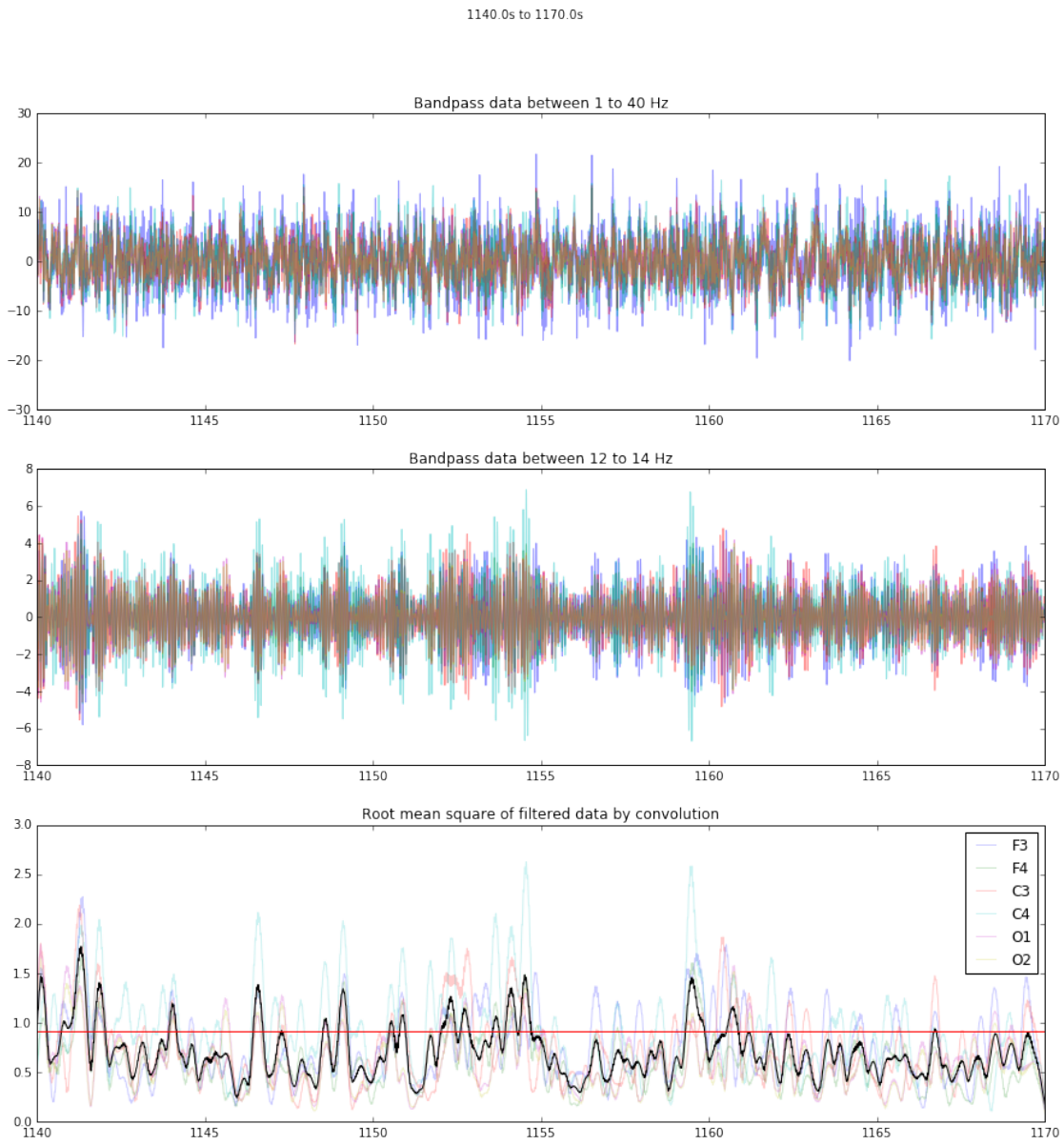

1050.0s to 1080.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

1080.0s to 1110.0s

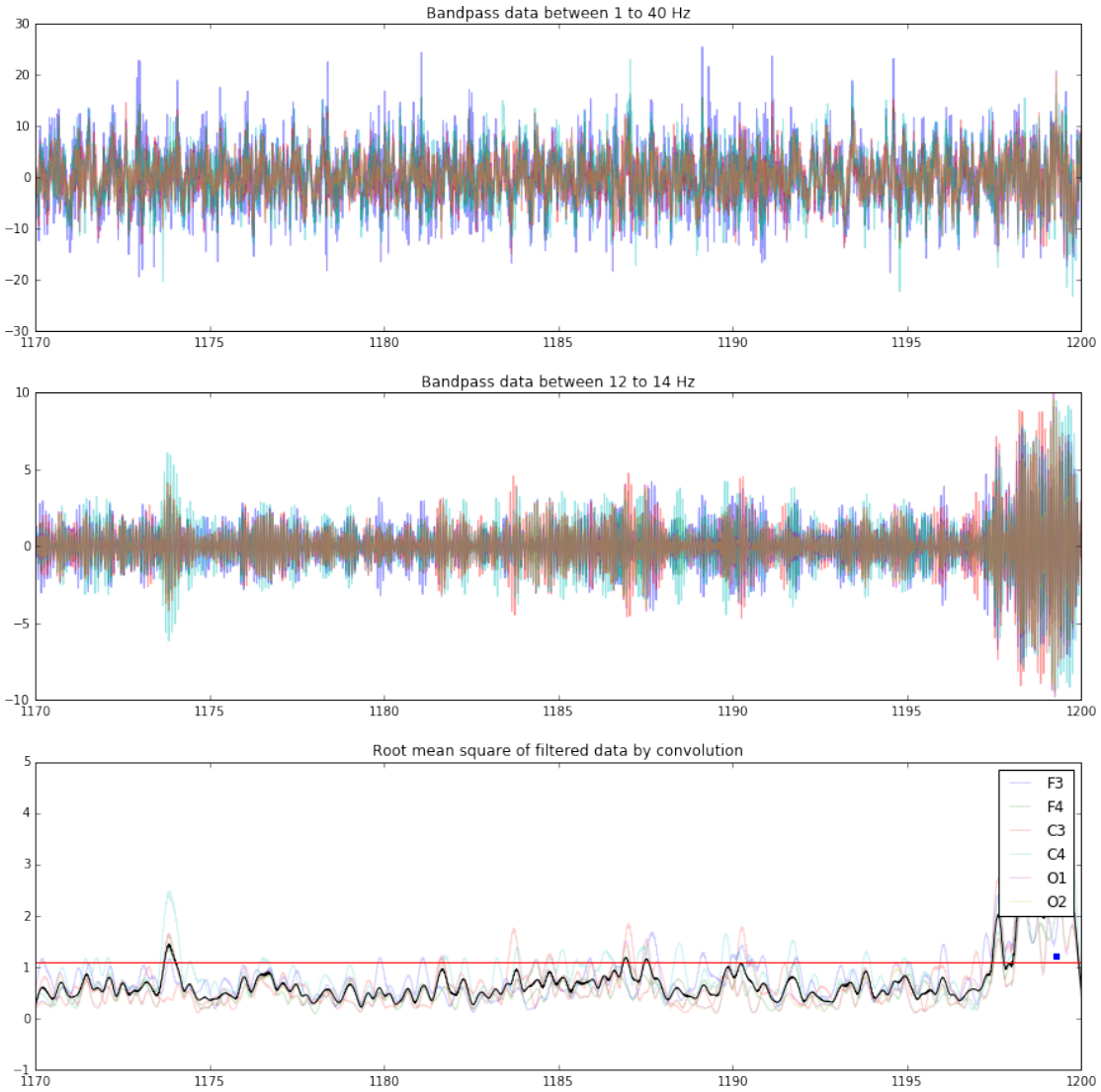


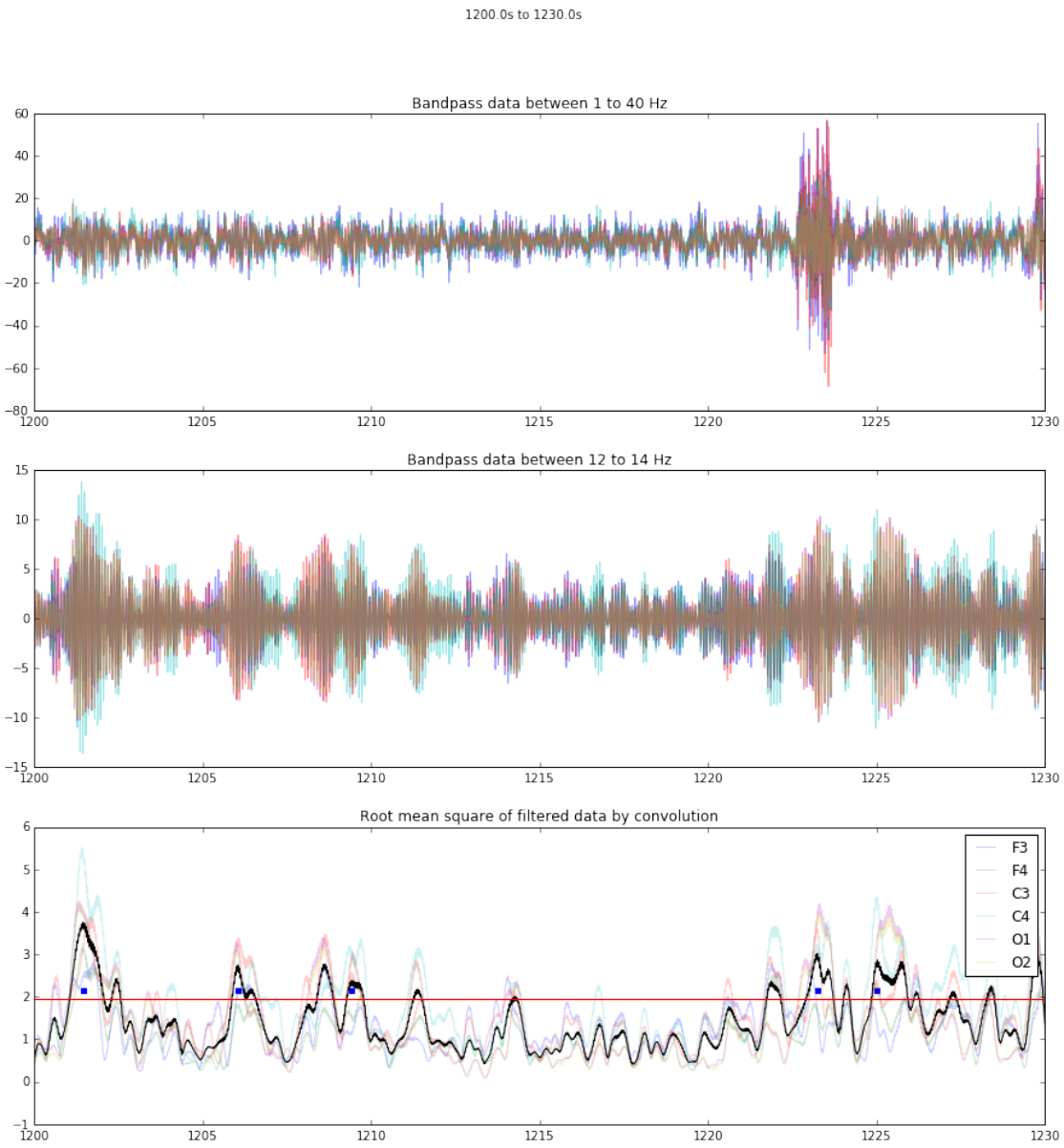
1110.0s to 1140.0s

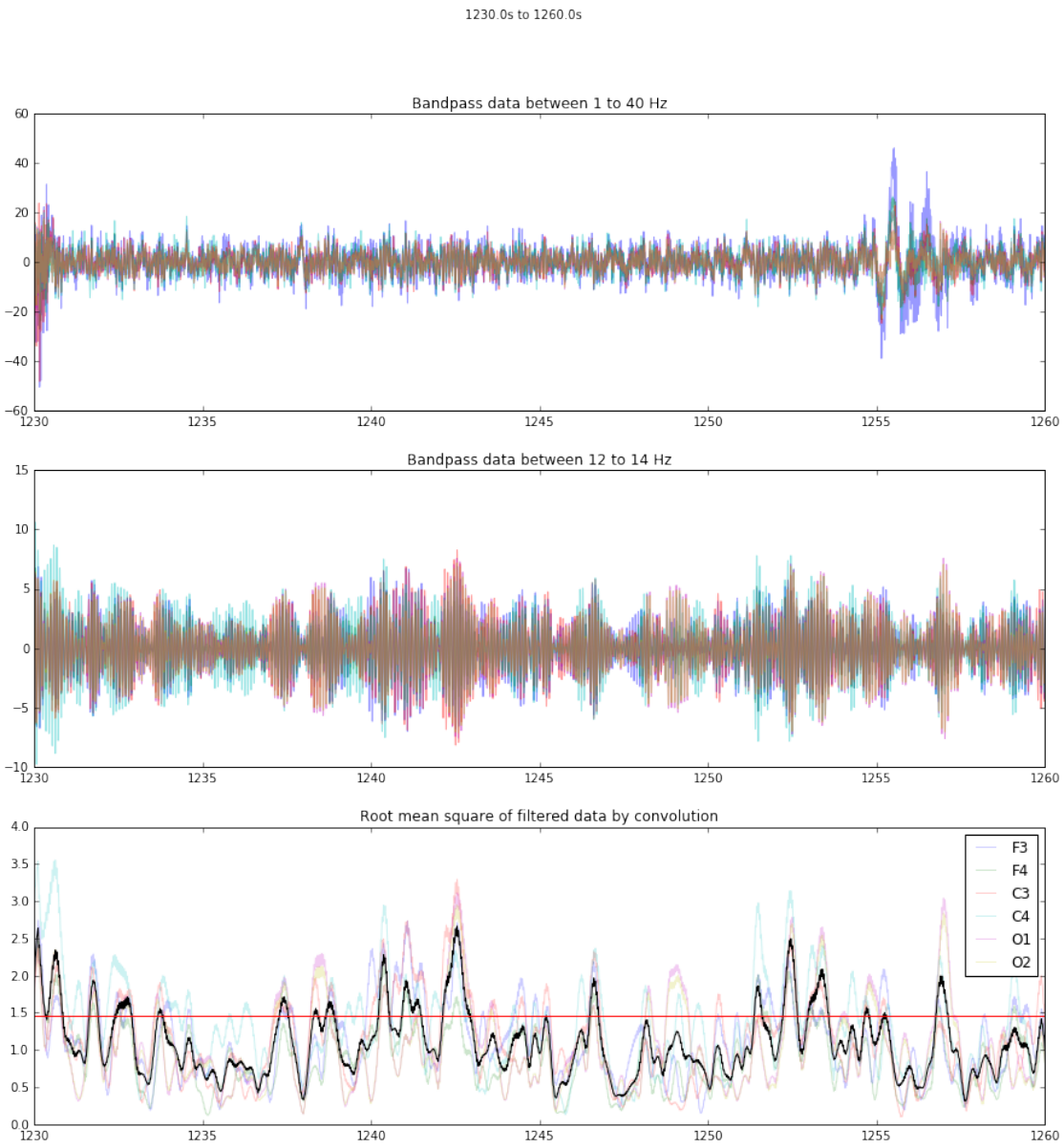



1140.0s to 1170.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

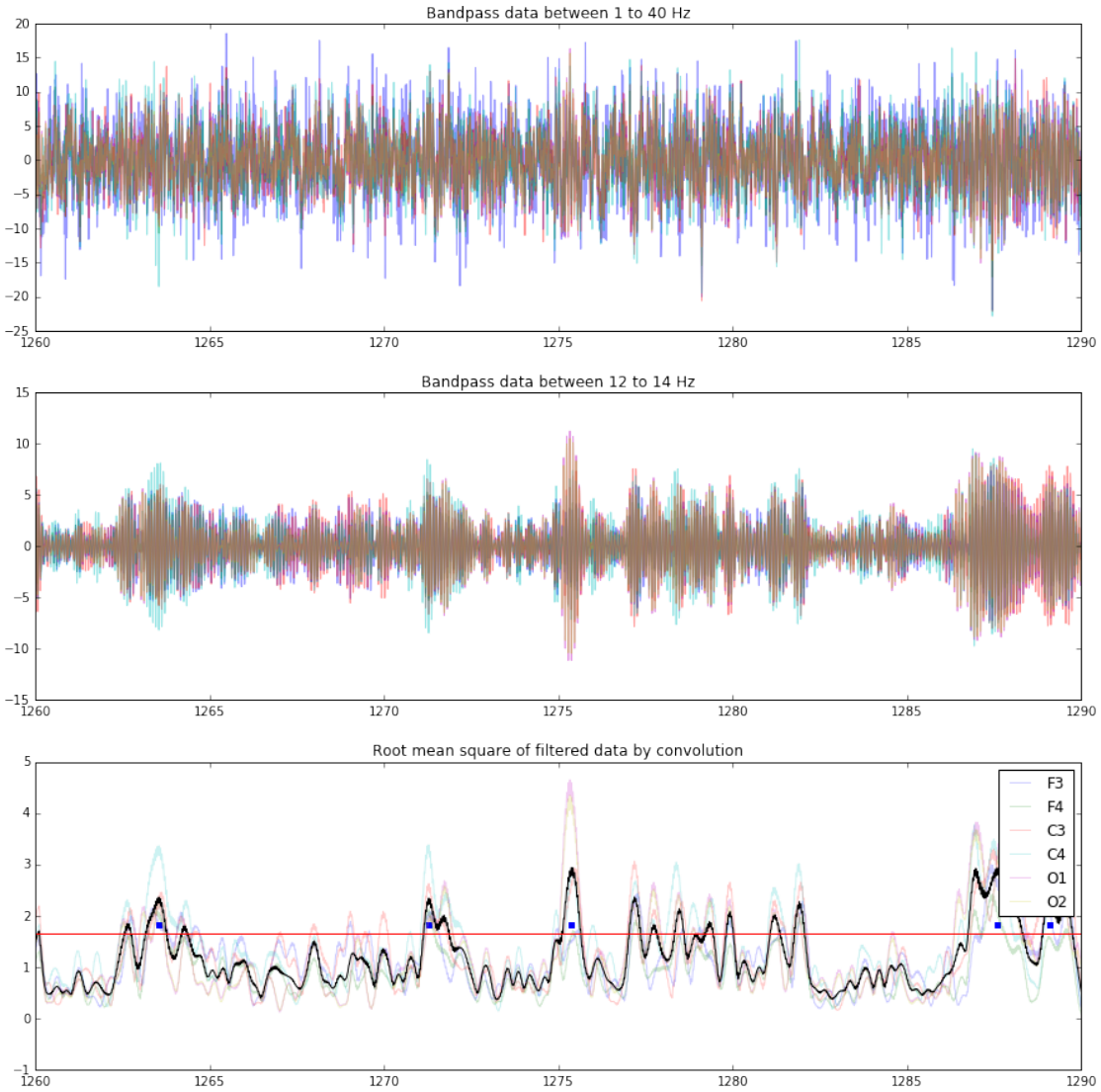
1170.0s to 1200.0s



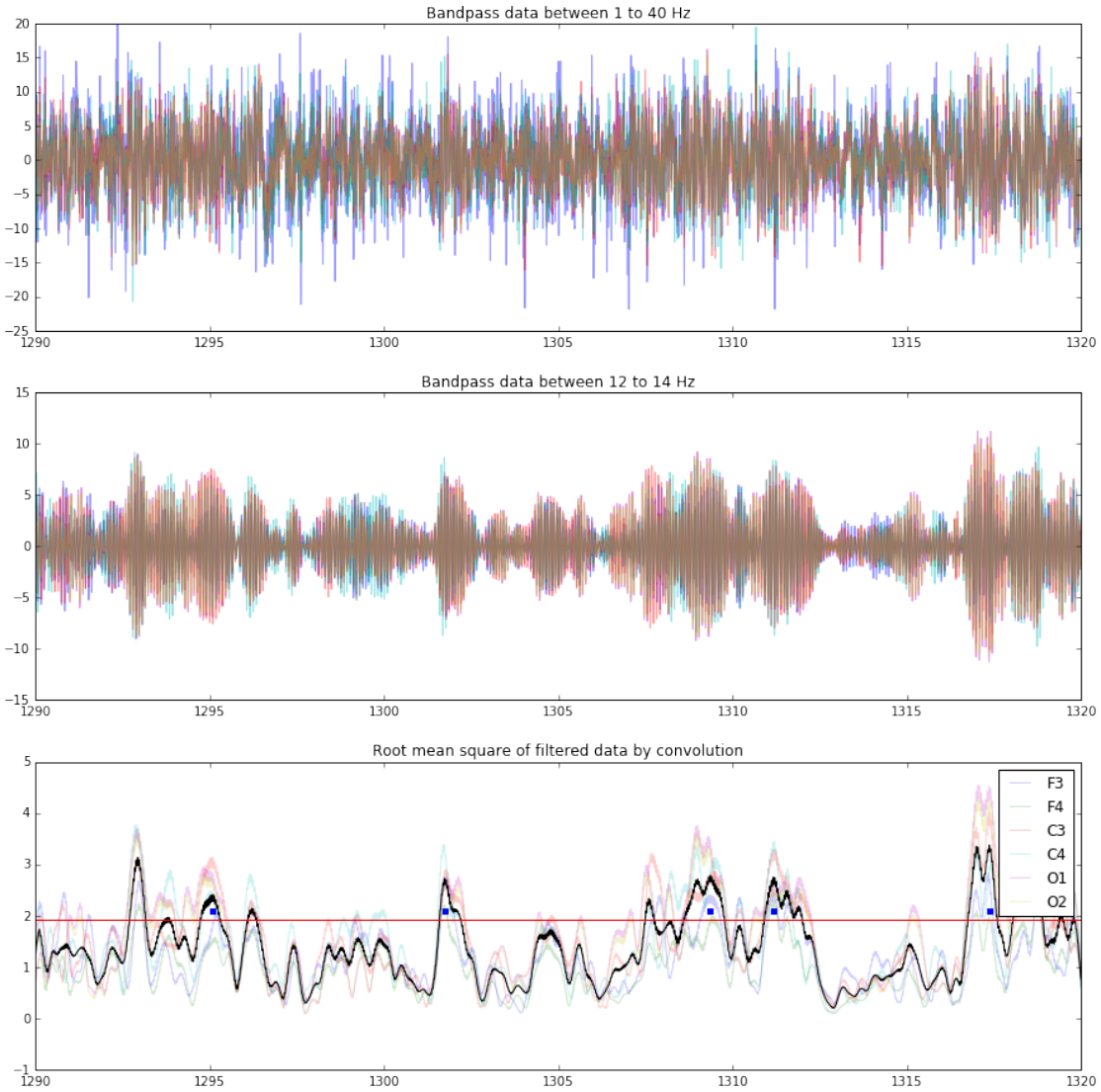

1200.0s to 1230.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

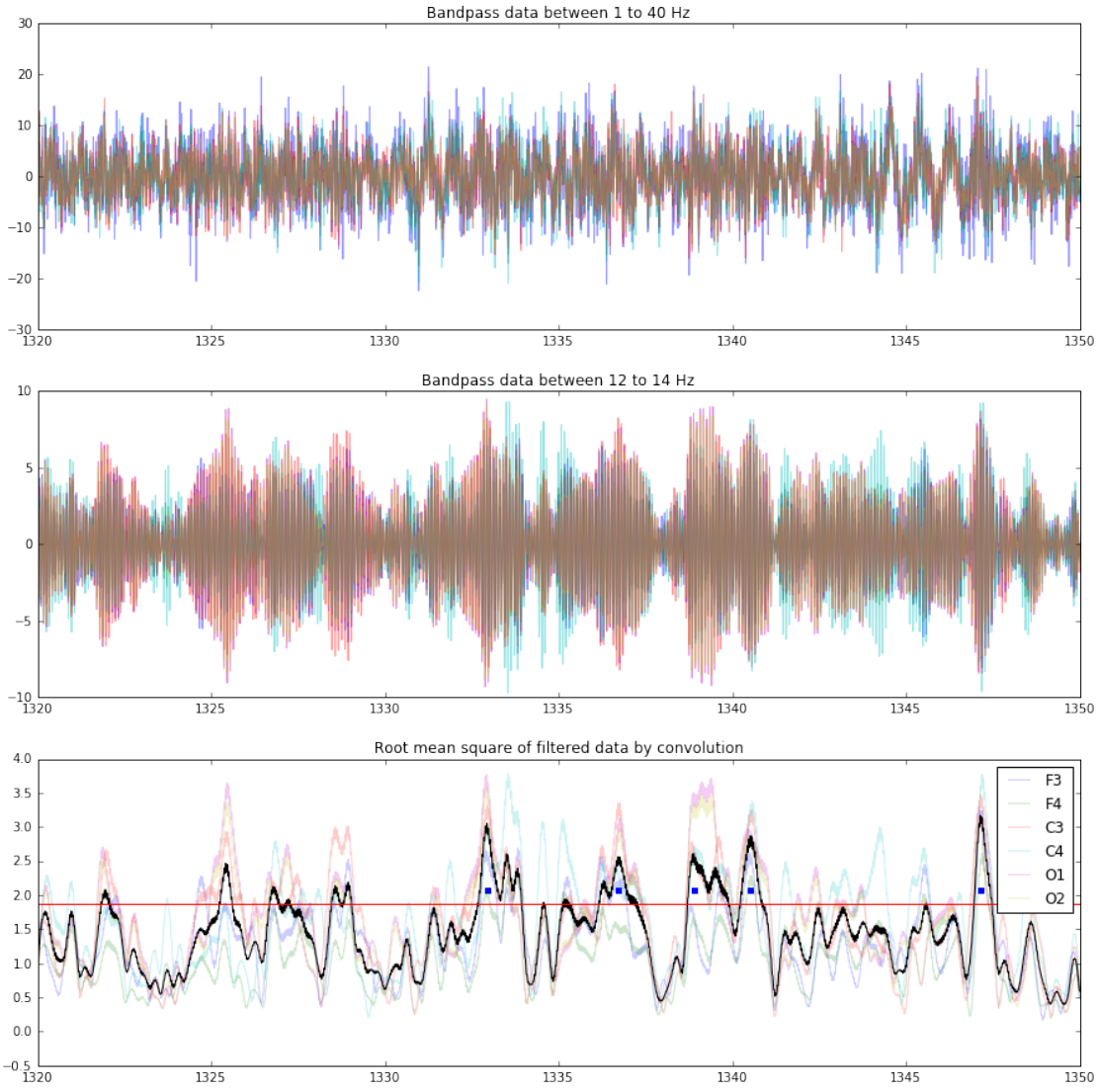

1230.0s to 1260.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

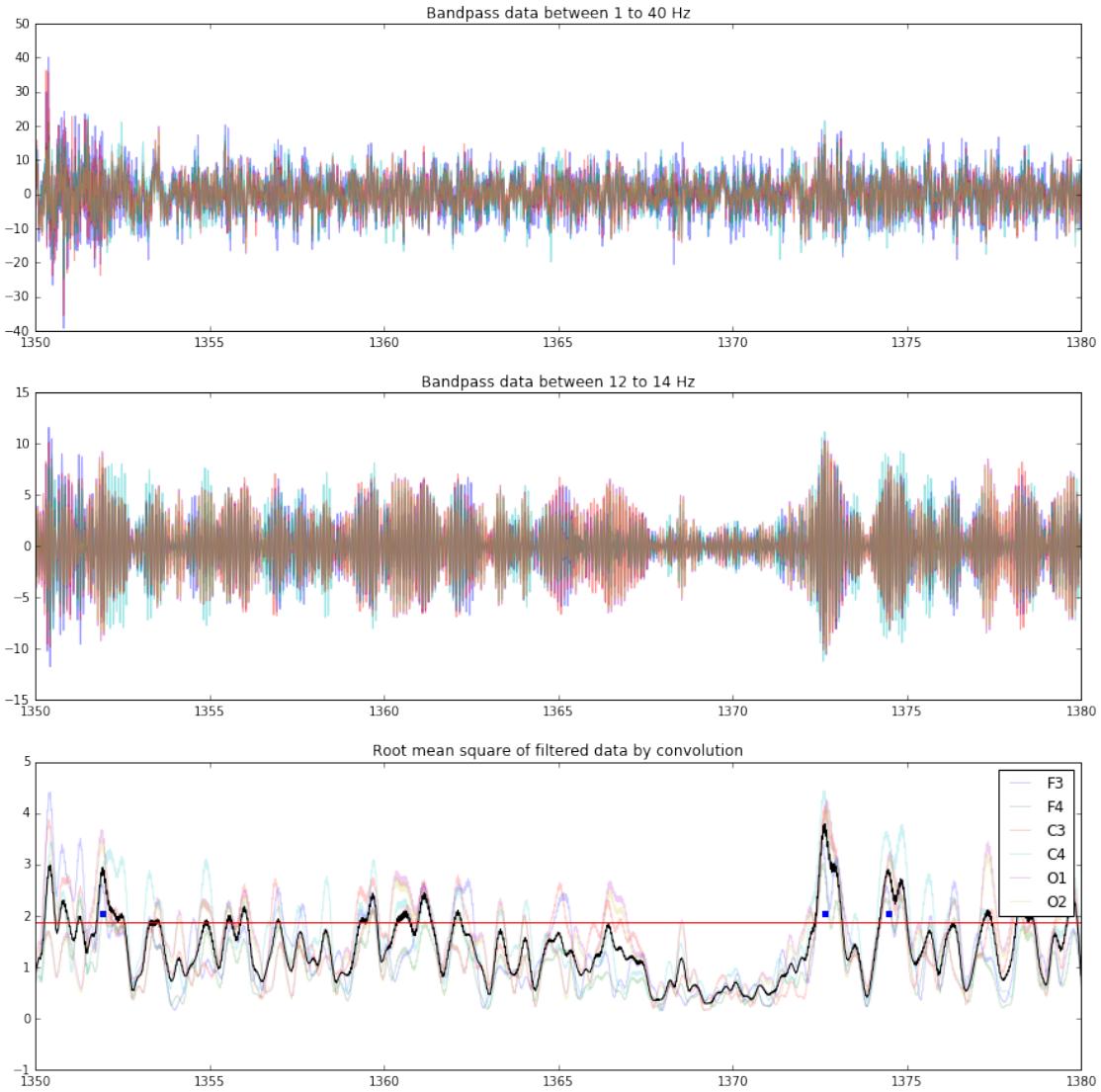
1260.0s to 1290.0s

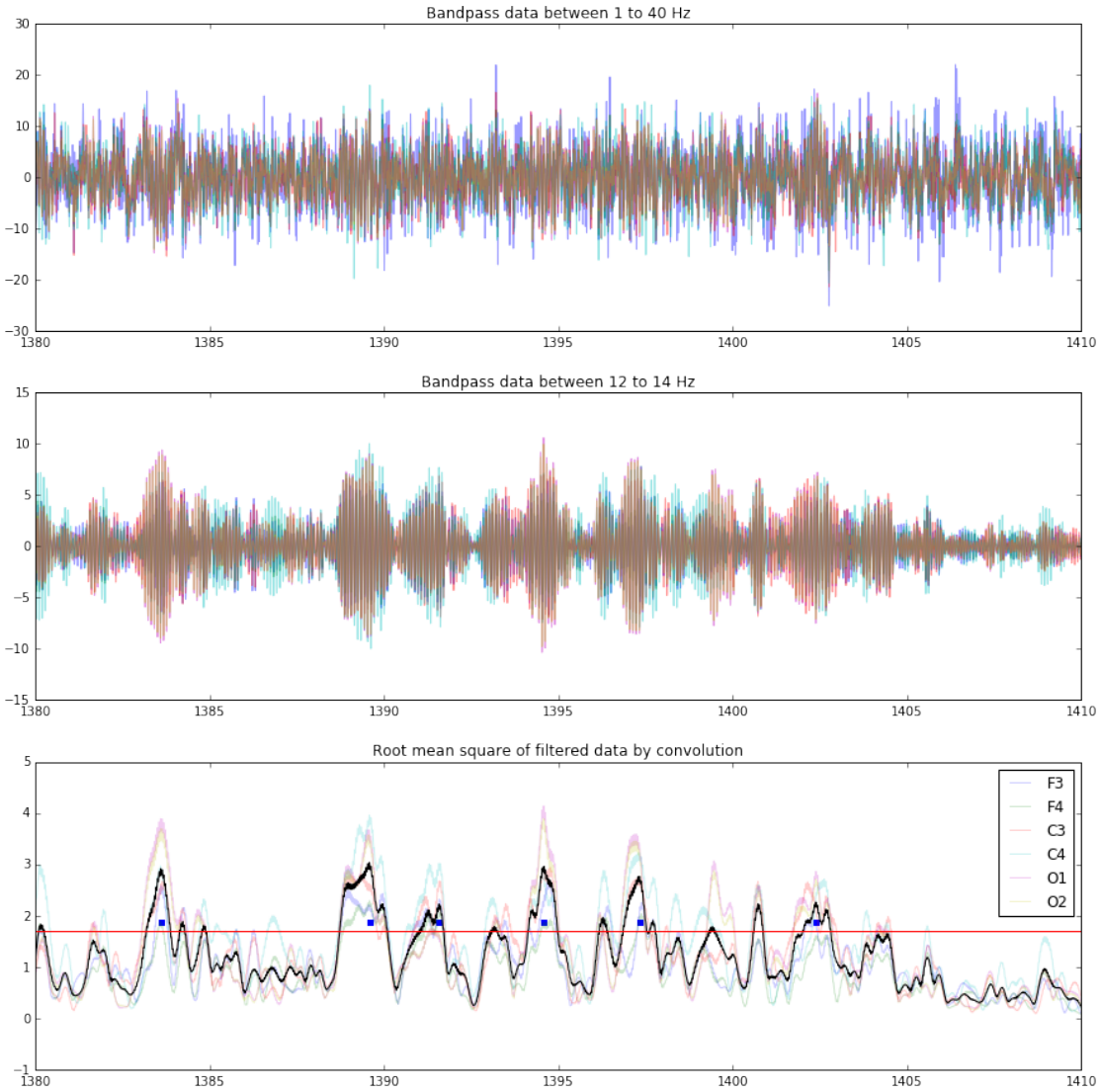


1290.0s to 1320.0s

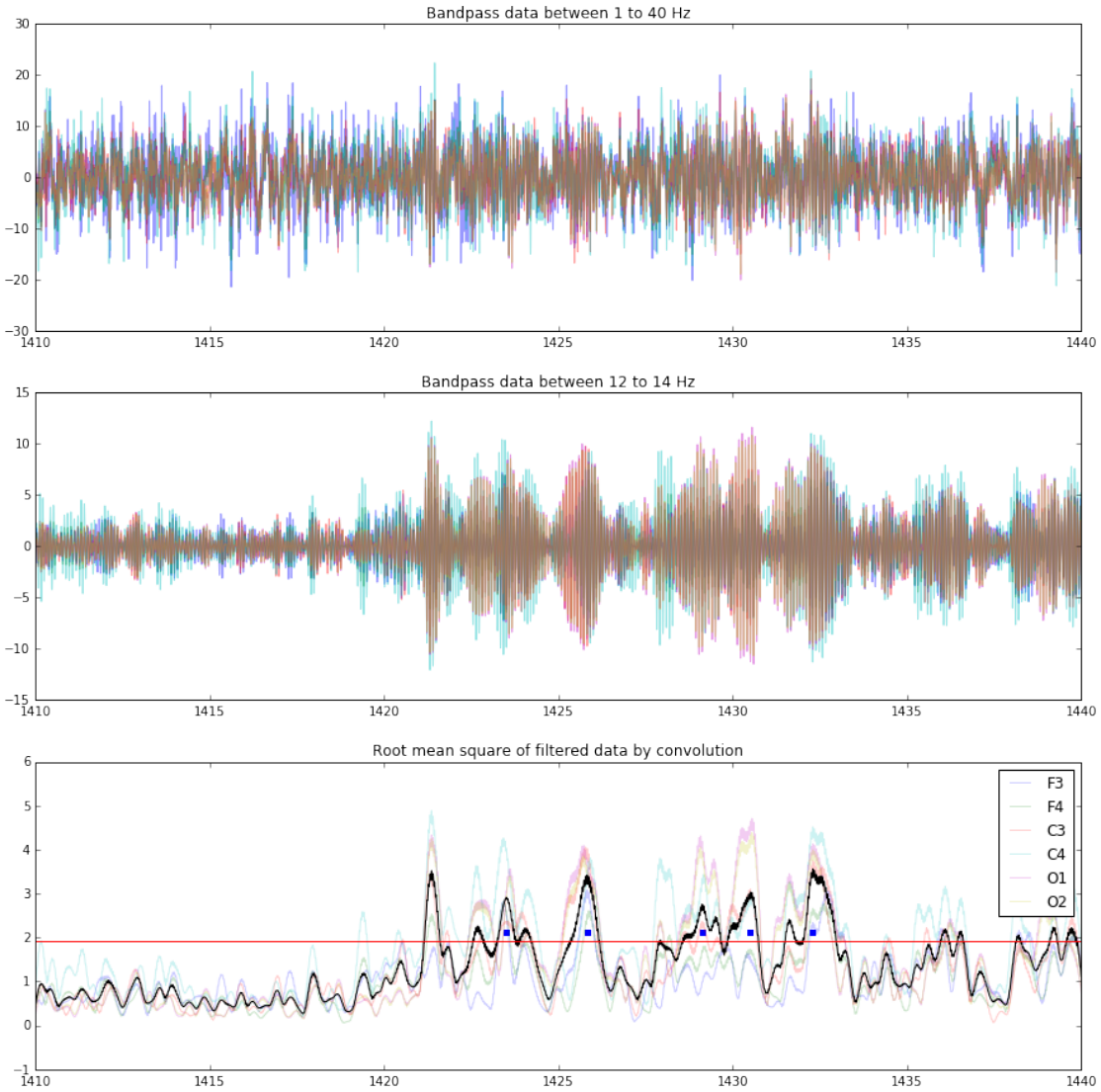



1320.0s to 1350.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

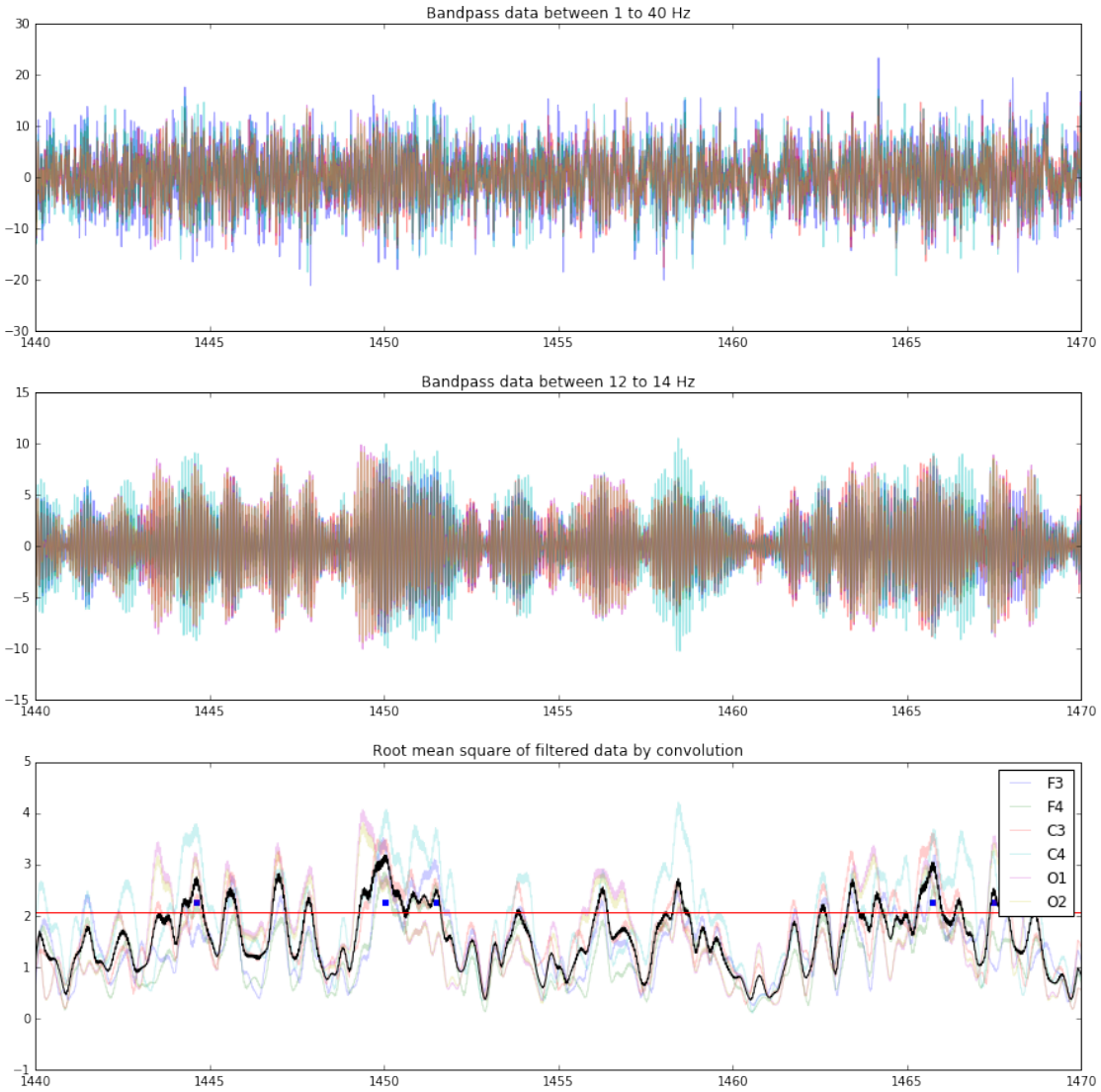

1350.0s to 1380.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2


1380.0s to 1410.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

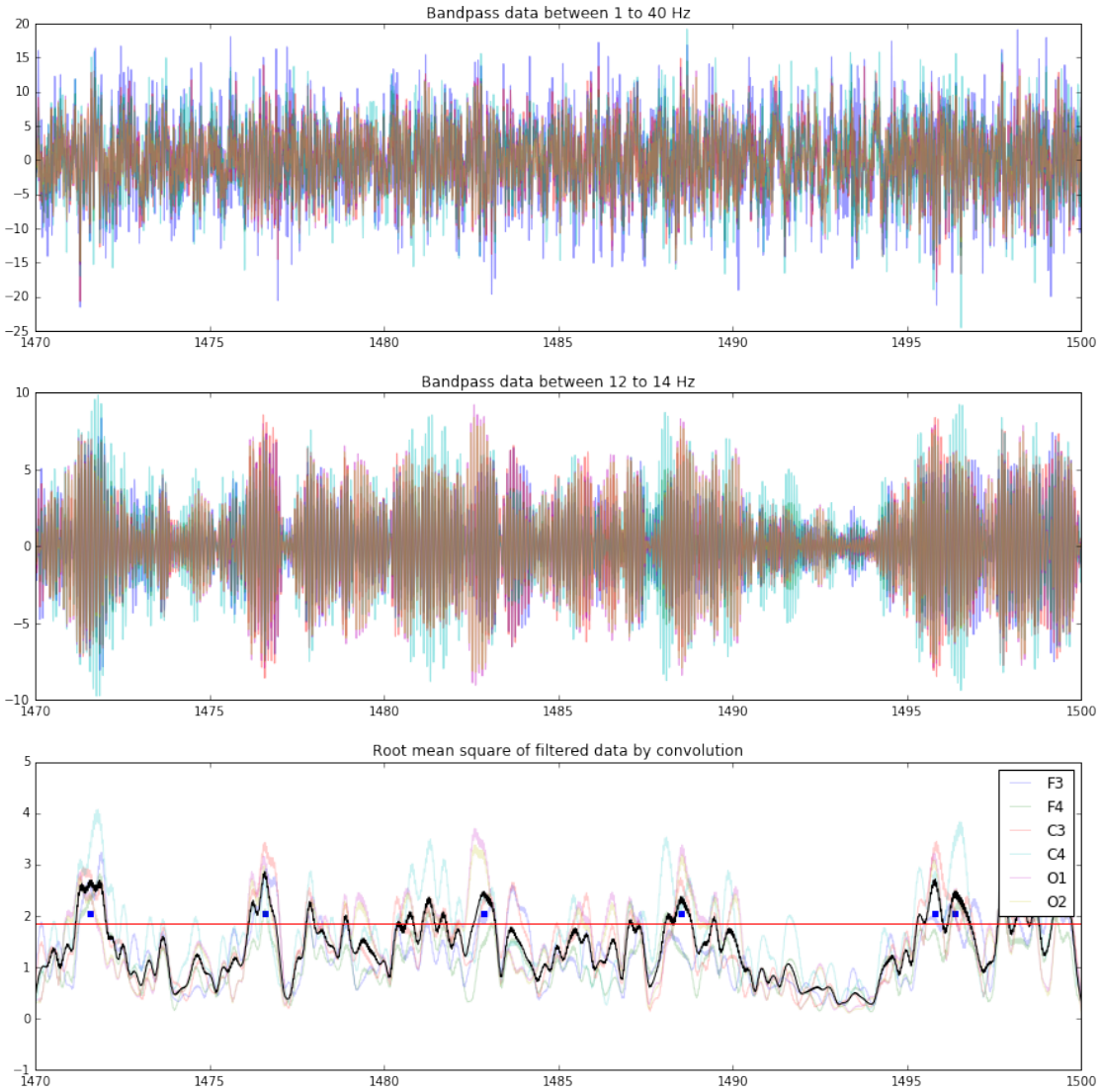
1410.0s to 1440.0s

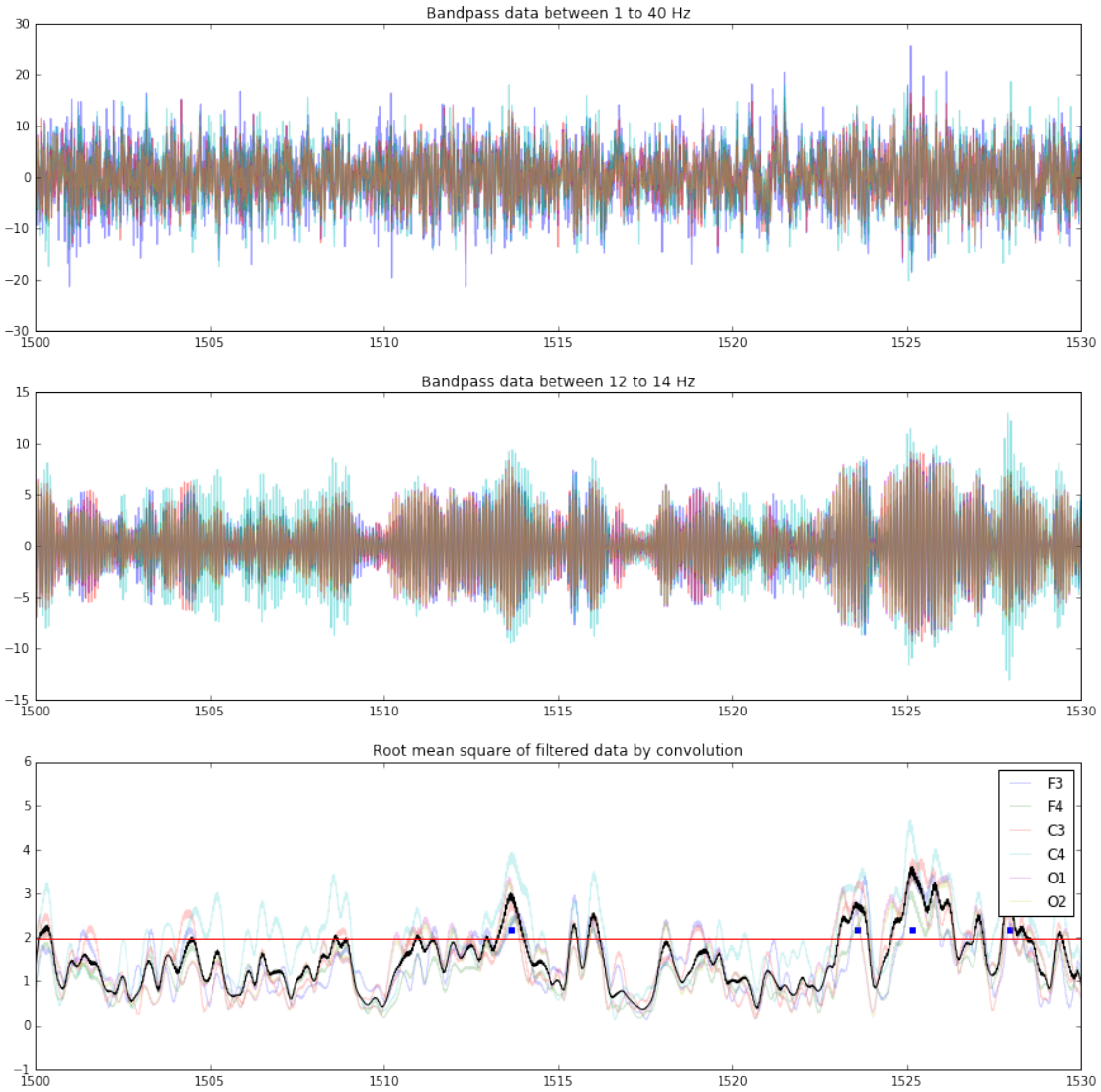


1440.0s to 1470.0s

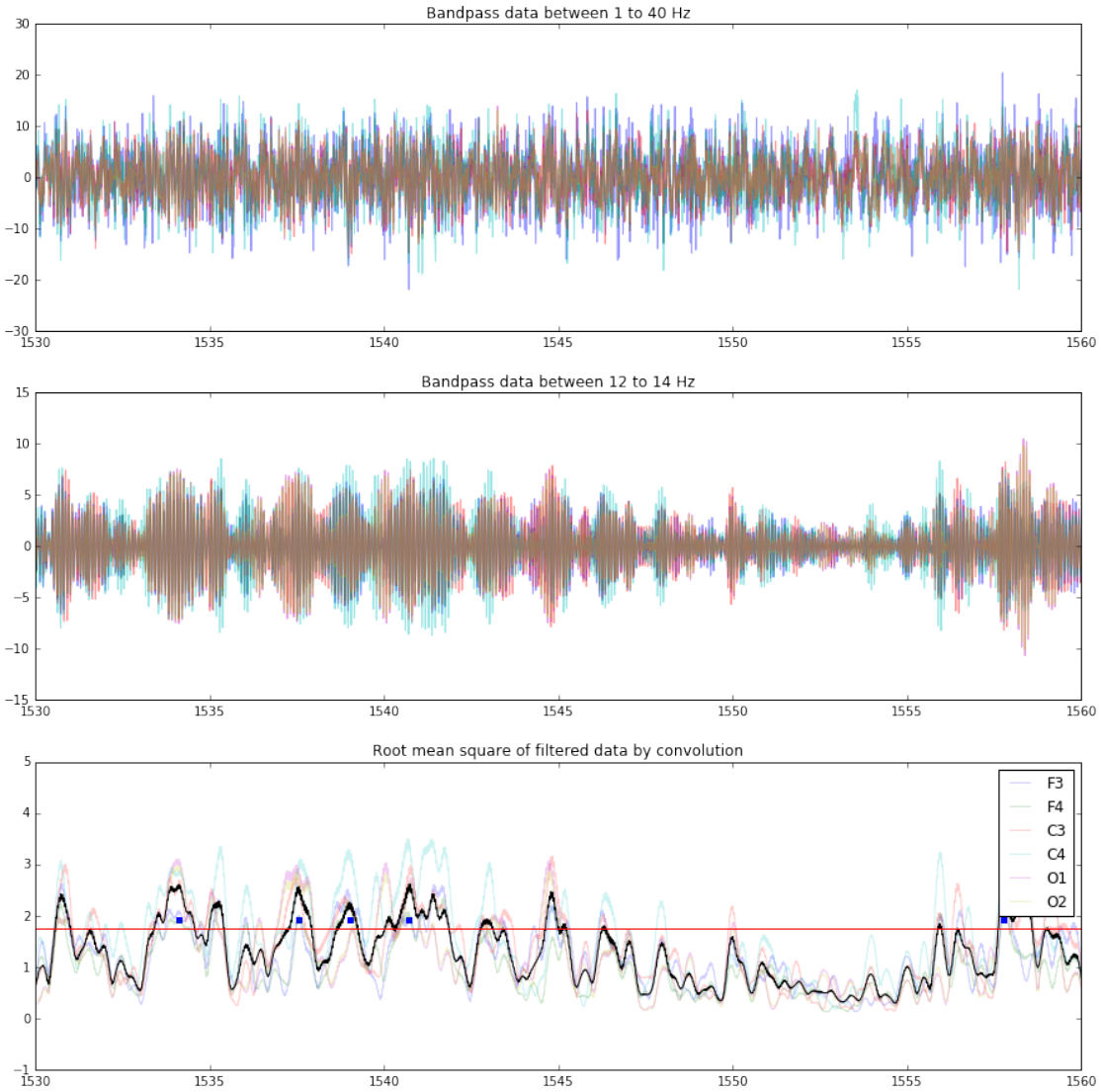


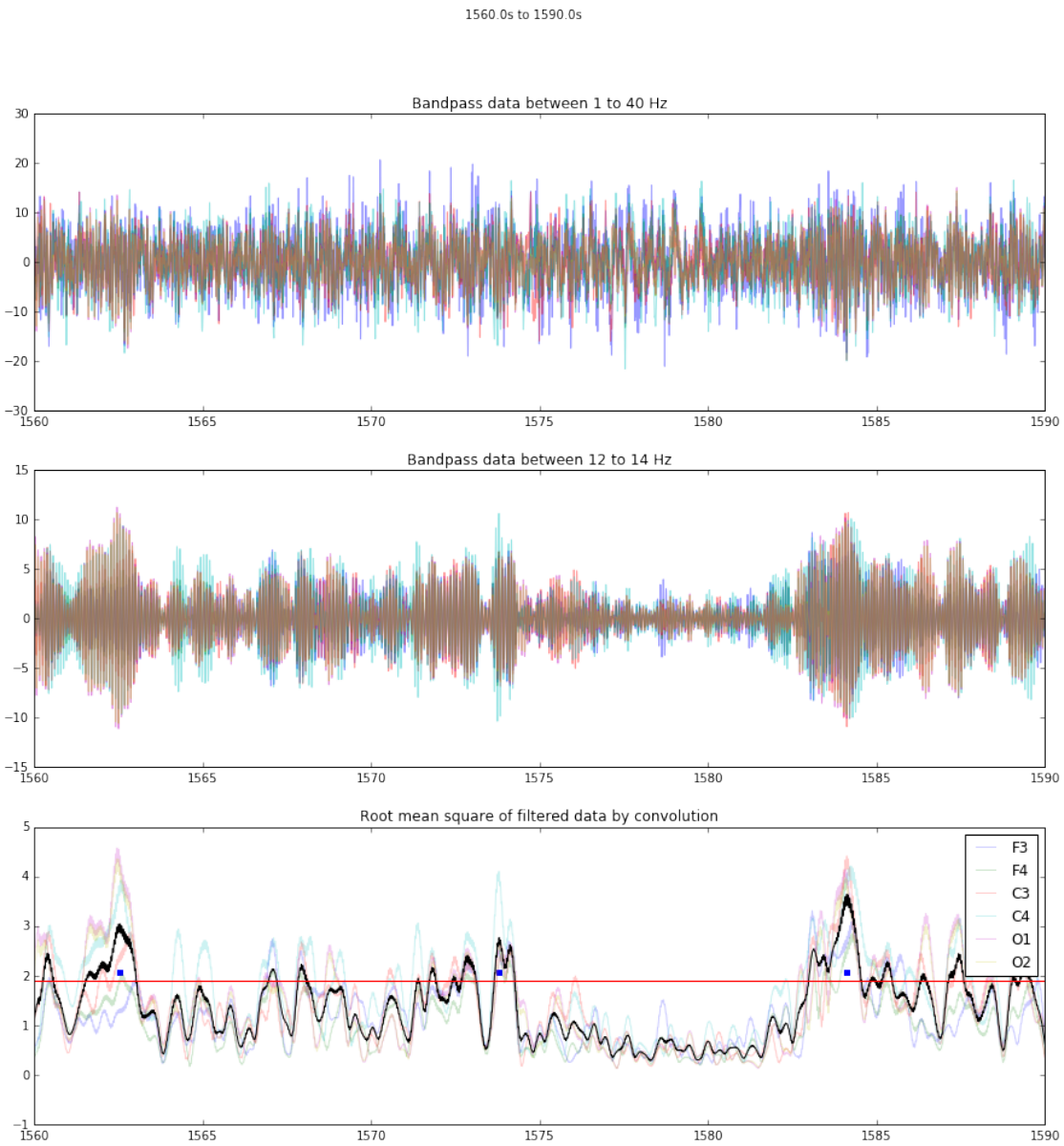
1470.0s to 1500.0s

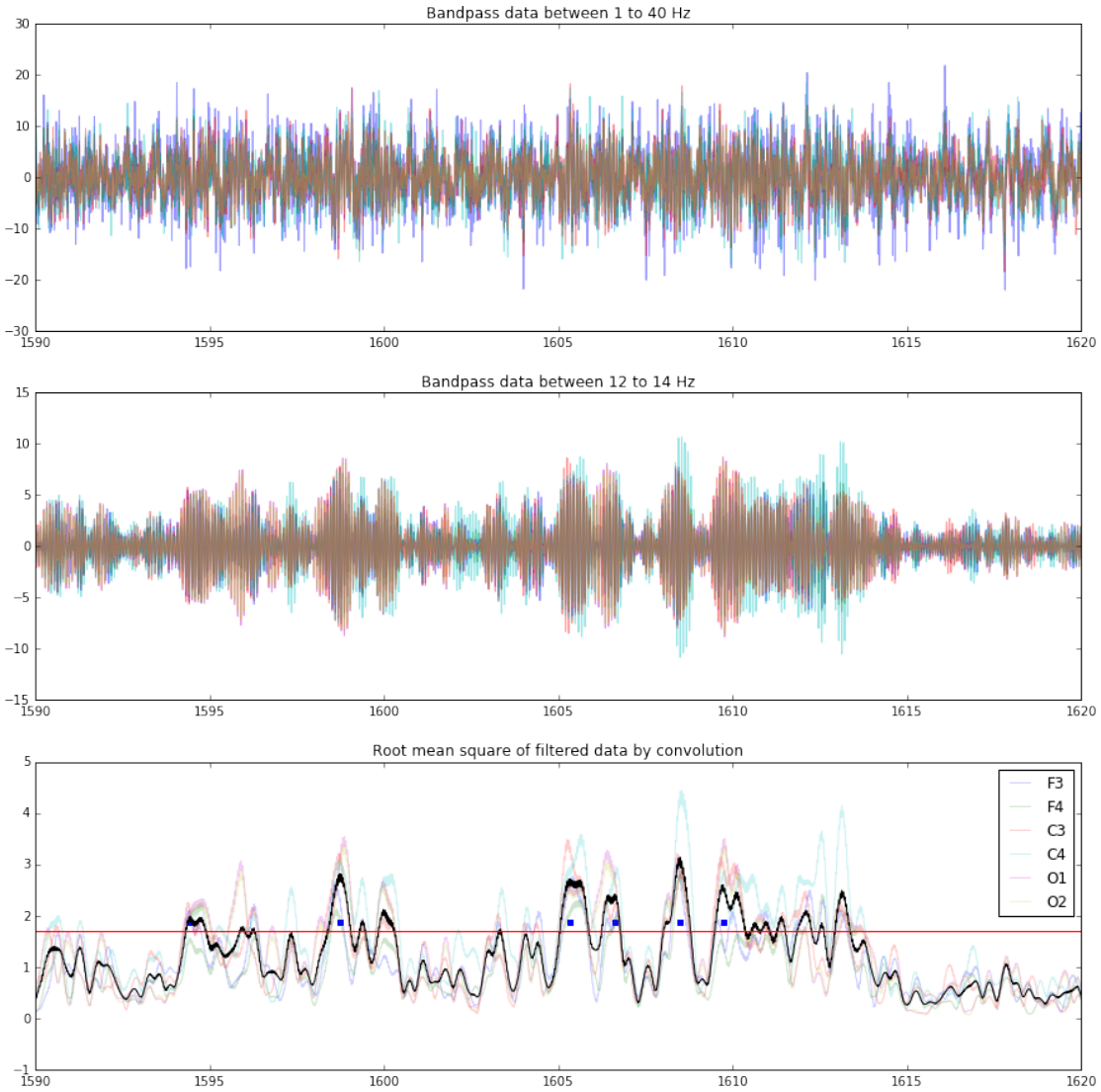


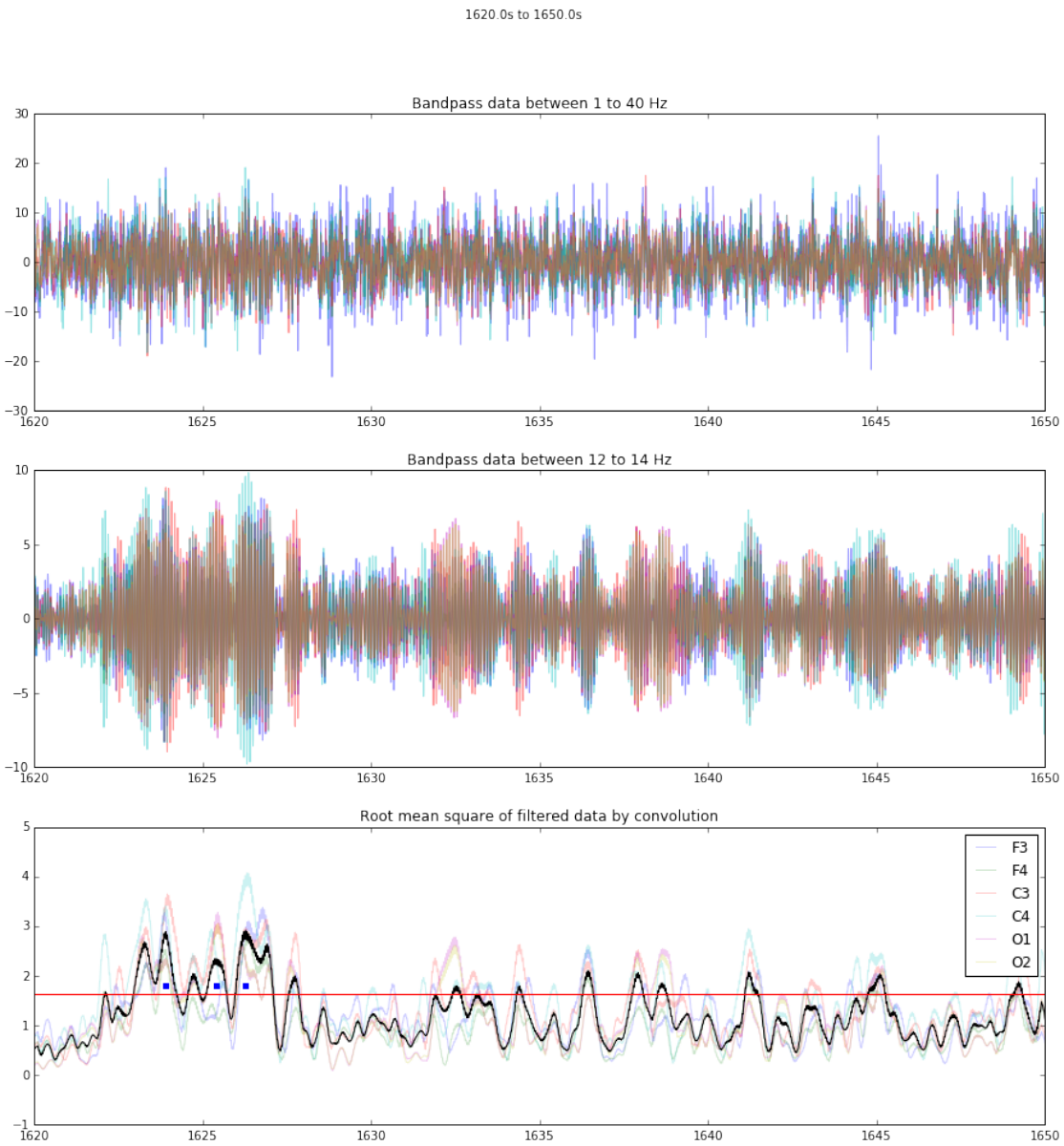

1500.0s to 1530.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

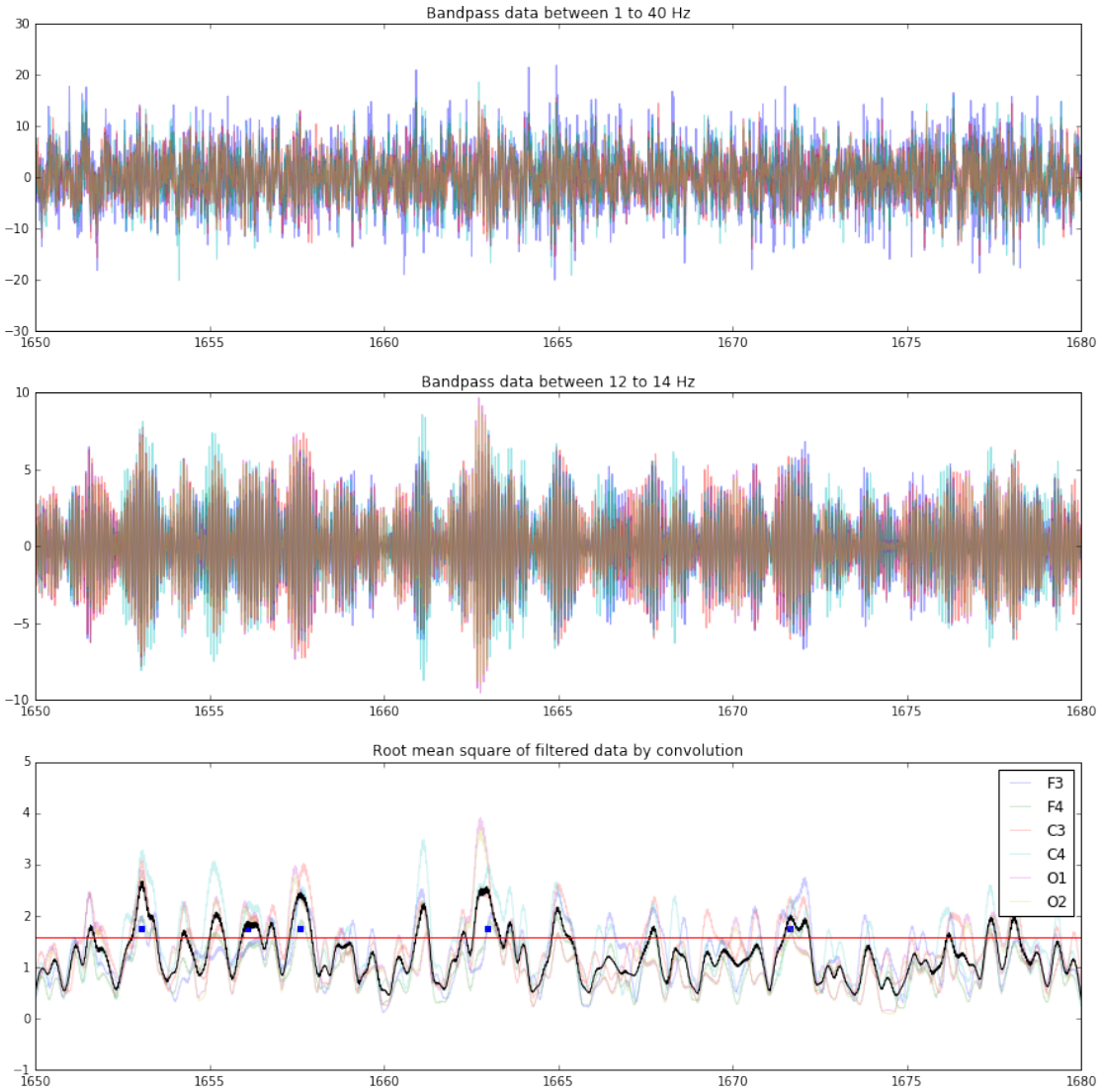
1530.0s to 1560.0s



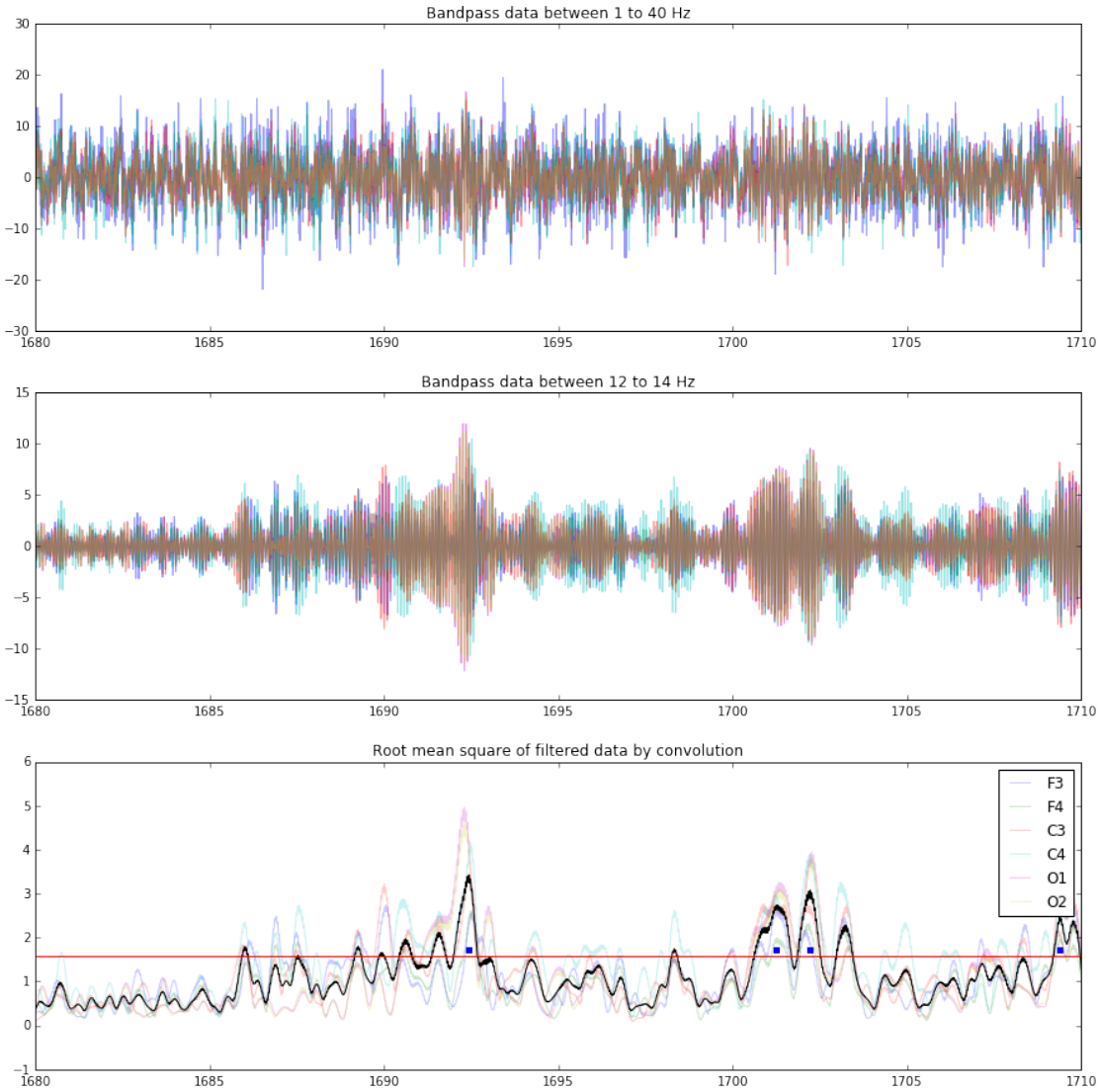

1560.0s to 1590.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2


1590.0s to 1620.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

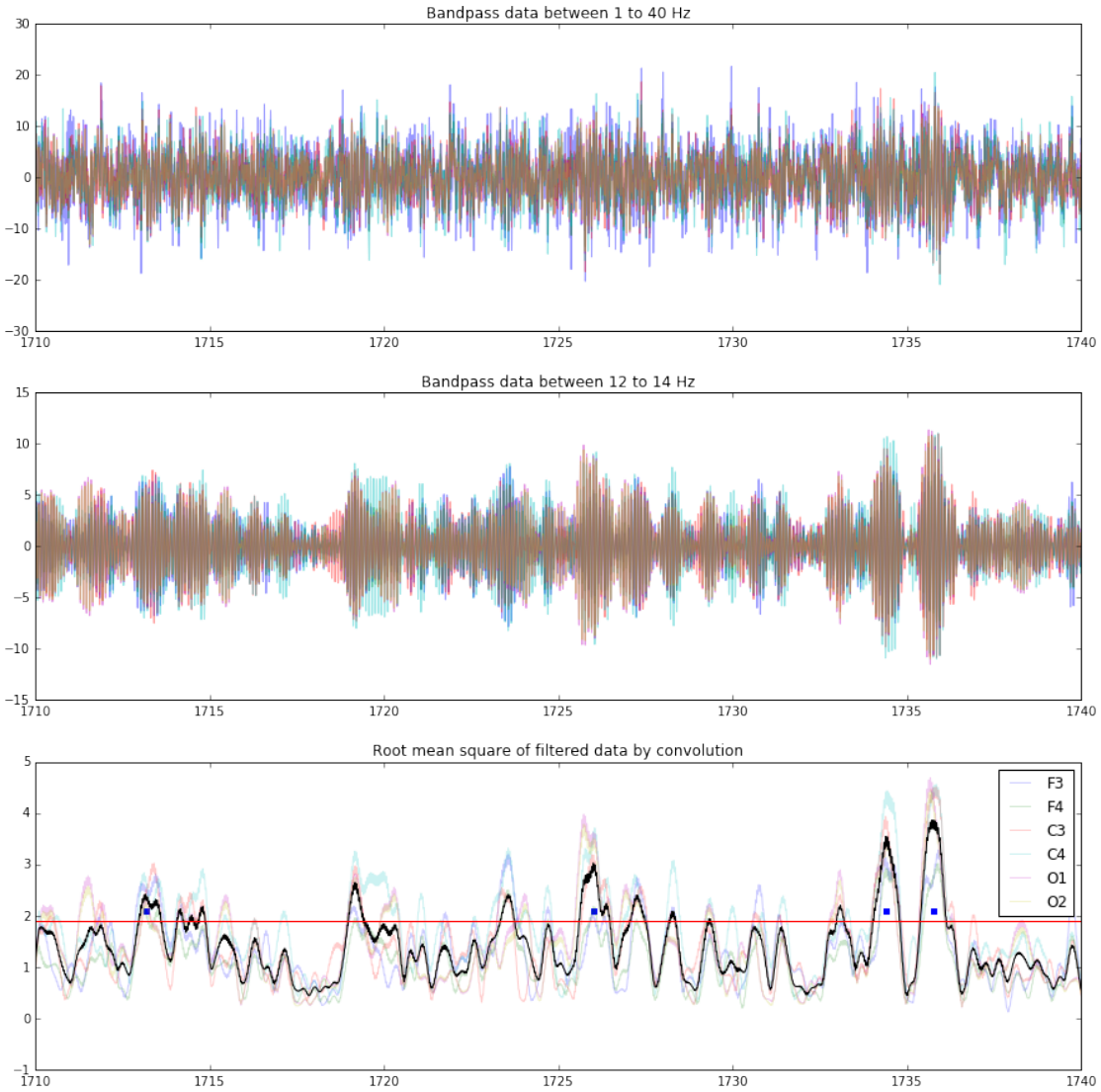

1620.0s to 1650.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2


1650.0s to 1680.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

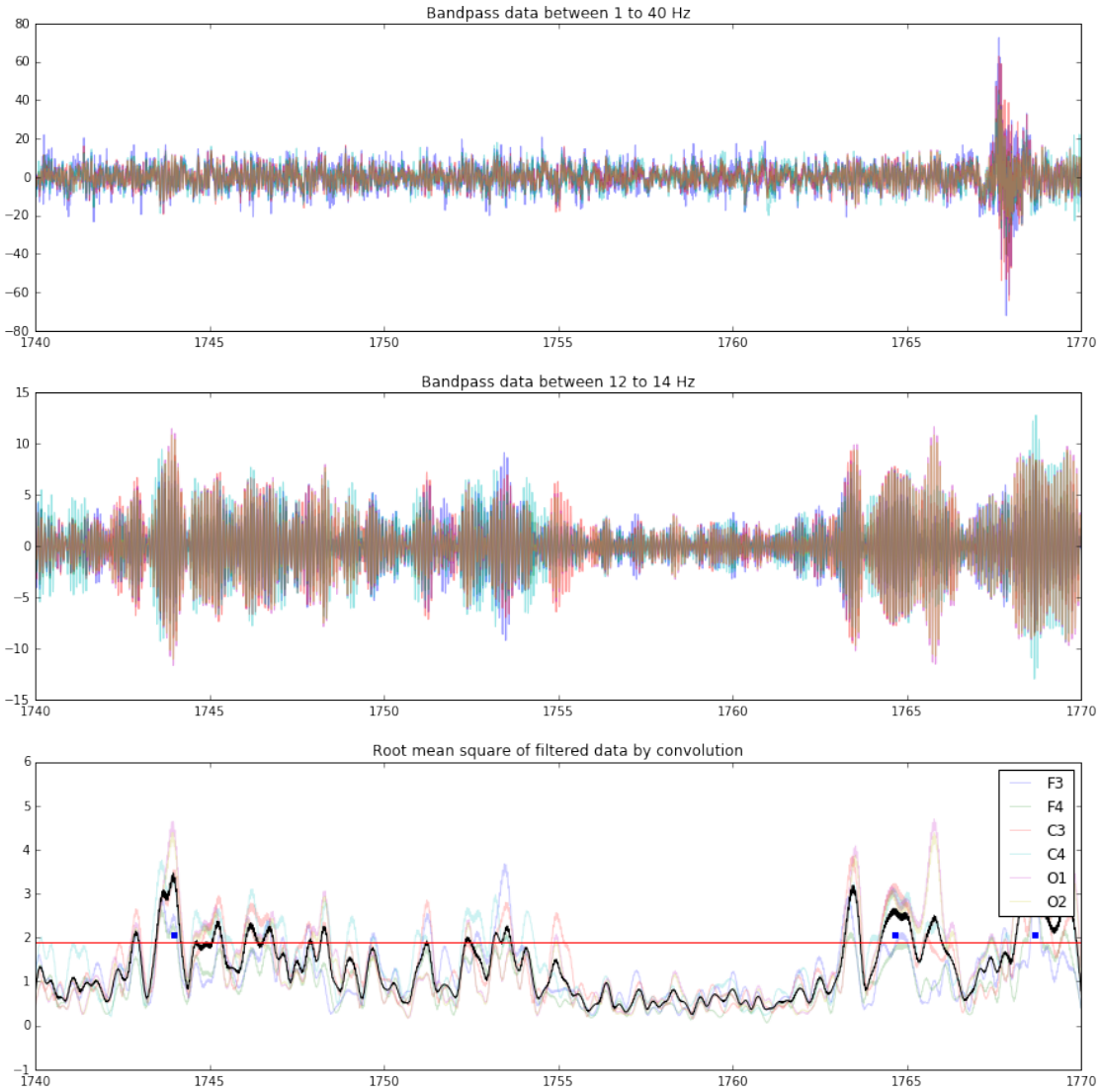
1680.0s to 1710.0s

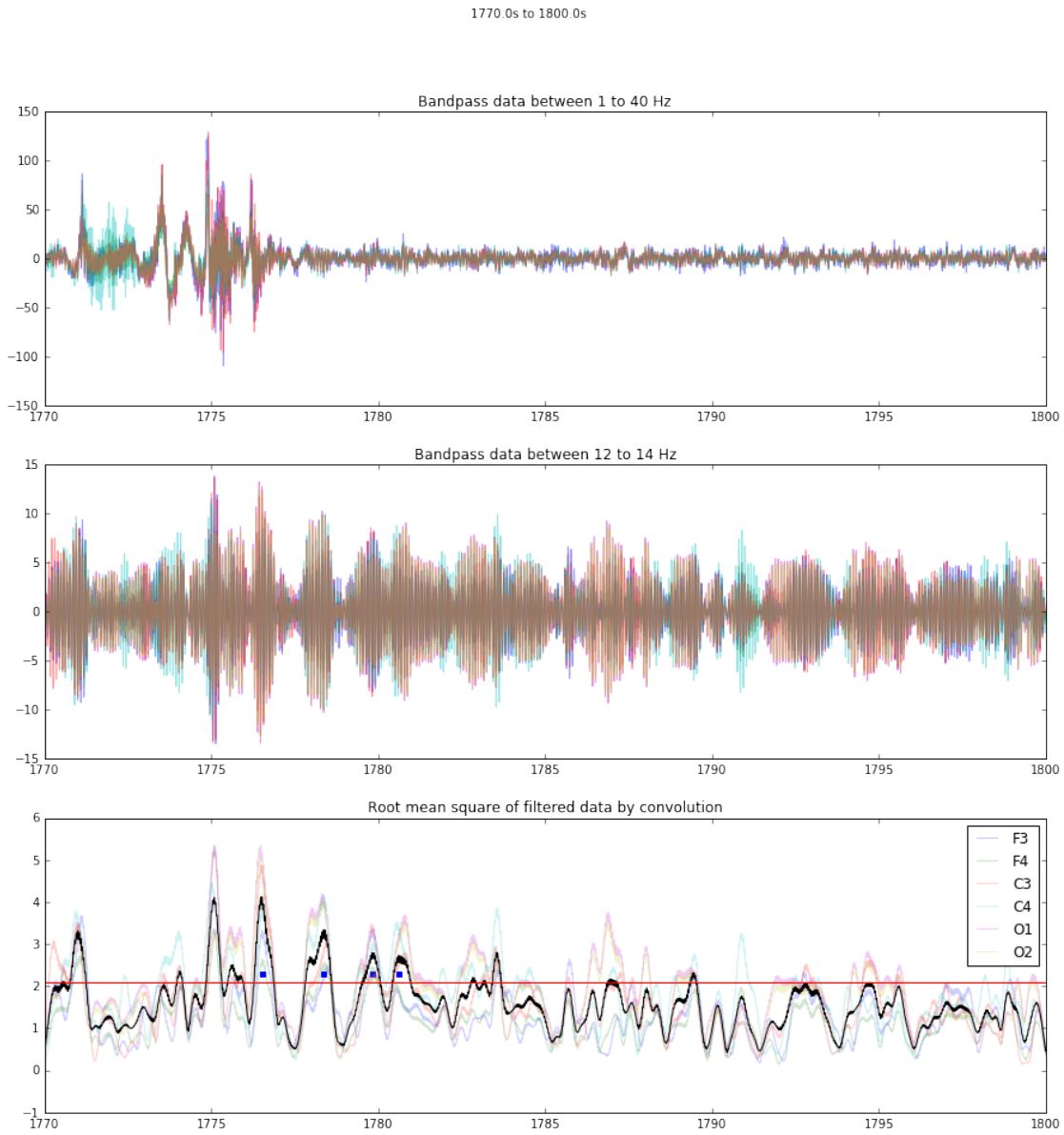


1710.0s to 1740.0s



1740.0s to 1770.0s




1770.0s to 1800.0s
Bandpass data between 1 to 40 Hz
Bandpass data between 12 to 14 Hz
Root mean square of filtered data by convolution
F3
F4
C3
C4
O1
O2

In [ ]: