New Task

What to do :

1. Segregate trial for the 18 different conditions:

a. 9 conditions corresponding to correct trials for Same (SOA1, SOA2 and SOA3), Opposite (SOA1, SOA2 and SOA3) and perpendicular (SOA1, SOA2 and SOA3) conditions.

b. Similarly 9 conditions corresponding to In-correct trials for Same (SOA1, SOA2 and SOA3), Opposite (SOA1, SOA2 and SOA3) and perpendicular (SOA1, SOA2 and SOA3) conditions.

2. When the answer to second question is No : Then this will be a condition corresponding to NoT2 (acronym for No second target). The Correct T2 field in this case would be coded as 5 so:

next you have to find what was the answer given by the participant:

a If givenResponse T2 field is == 5 then the answer is correct and store these trials as NOT2 correct
b. If givenResponse T2 field is == 6 then, the trial is incorrect then store it in another array NoT2 Incorrect.

The task started by importing the data- ParticipantData. Made 2 matrices to store the correct 3r response and correct 2nd response(total occurrences not indexes this time). The mastix name are :

1. correctSOADir
2. incorrectSOADir

This represent SOA being on x-axis and dir being on y-axis.
What we did was storing @)A 1 on (1,1) SOA2 on (1,2) and SOA3 on (1,3) respectively and write direction herer.

Also there are two variables for storing correct and incorrect 2nd rsponse.as ‘correctResponseNo’ and ‘incorrectResposeNo’

First checking if first response is correct or not. If not we have not considered that case.

There are two students whose data were not recorded so we need to exclude by checking if size of their data events is greater than 11.

Also there are some cases where GivenRes1 is not given and GivenRes3 is not given so excluding those events for those these are not mentioned

Now checking the conditions given to us and increasing the count for that event or case.
Also we are not given correctRes so we need to give them values according to directions.

Numbering system provided to us :
Codes for first response: Up/Down/Right/Left

0: 1
90: 2
180: 3
270: 4

Codes for second response: Yes/No

Yes :6
NO : 5

Codes for Third response: Same/Opposite/Perpendicular

Same : 7
Opposite: 8
Perpendicular: 9

New Task

What to do :

Make 3 plots one for SOA1, second for SOA2 and third for SOA3. In this plot 3 lines one would be representing same direction for second target and similarly other for opposite and perpendicular.

Then go to EP struct and in this we have T1_start and T2_start and their ending time as well.

There is a struct PD(Pupil Diameter). We need to plot this. But take average in two stages:

1. For a single participant. For example if 2,4,5 DataEvent correspond to SOA1 same then take PD for all of them and store the average.
2. For all participant. This means at end take average for all participants.

For now we took PD from T1_start to T1_start+1000 index only.

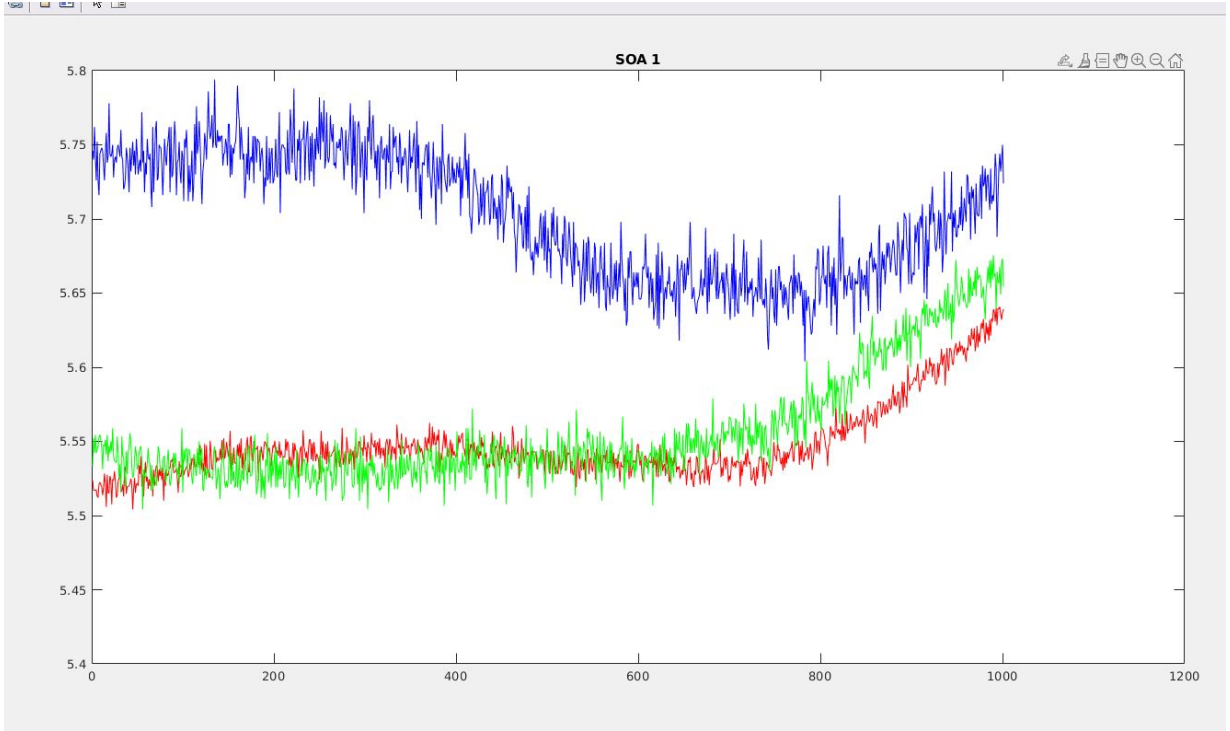
Made two functions:

1. doit - This function will take 5 arguments givenRes3, j (index showing which DataEvent is it) and three matric sameSOAindex, oppSOAindex and perpSOAindex and according to givenRes3 we update one of the matrix respectively

2. temp - This function takes 4 argument X(which is in fact storing the ParticipantData of the ith student), then the matrix we made from above function.

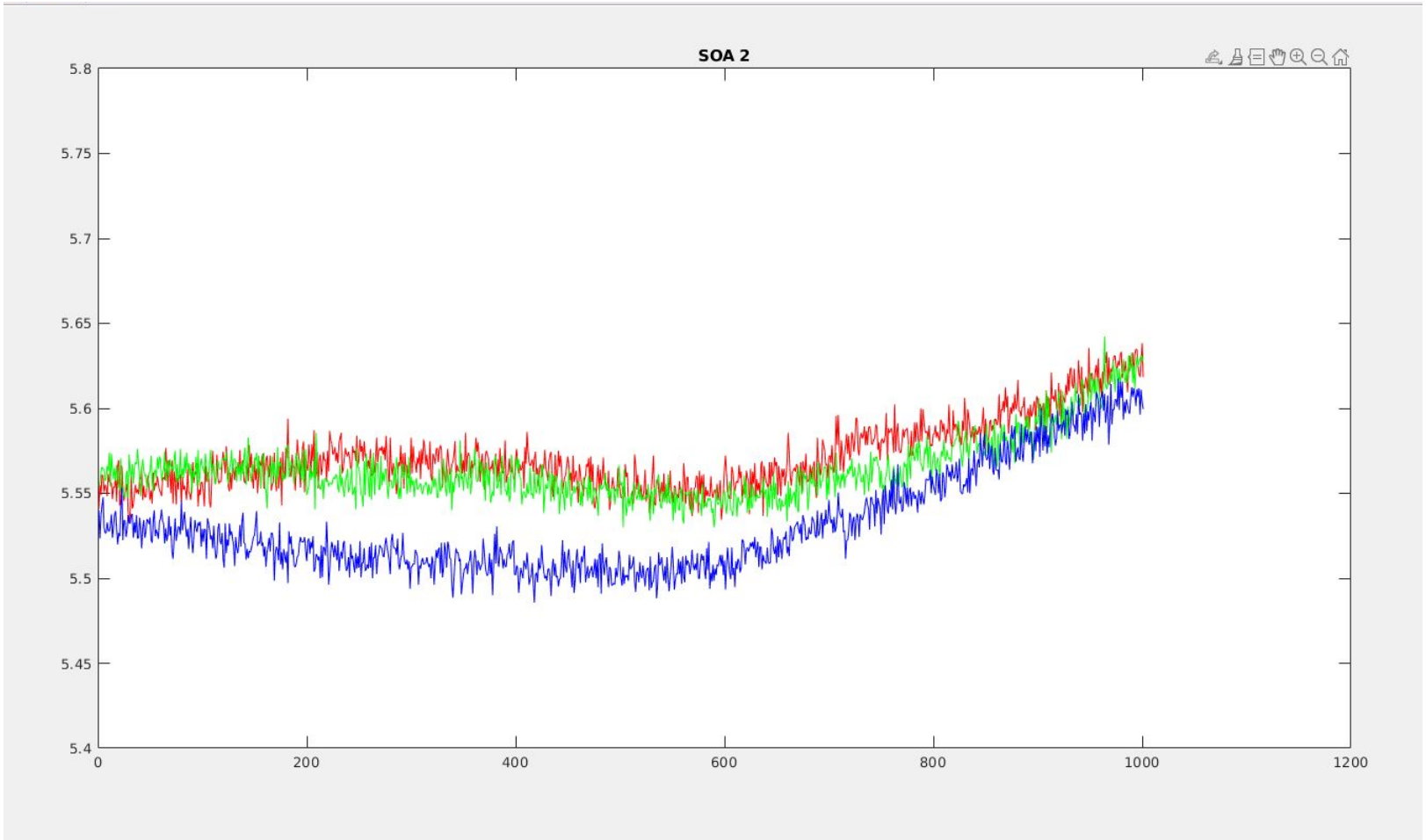
   In this we are doing is checking the indexes and then going to that EP struct for that index, storing its T1_Start and updating the valueSame, valueOpp,valuePerp which is a matrix of size 1001 initially storing zeros and then adding the PD values to them.
   At end taking average by dividing by size of that index matrix.

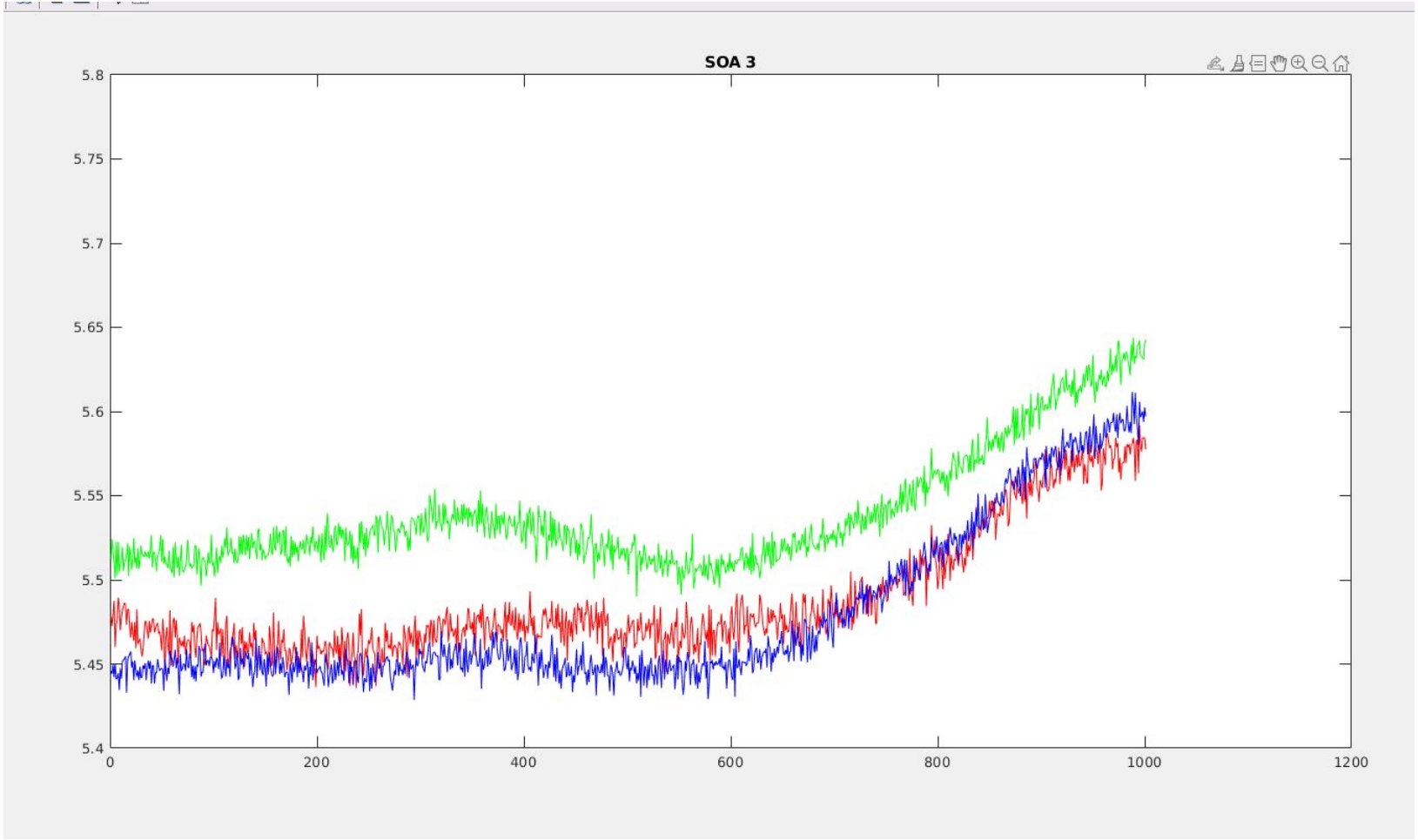
   At last dividing by the number of participants



SOA 1

For all graphs
Red - same dir
Green - opp dir
Blue - perp dir


SOA 2
5.8
5.75
5.7
5.65
5.6
5.55
5.5
5.45
5.4
0
200
400
600
800
1000
1200


SOA 3
5.8
5.75
5.7
5.65
5.6
5.55
5.5
5.45
5.4
0
200
400
600
800
1000
1200

New Task

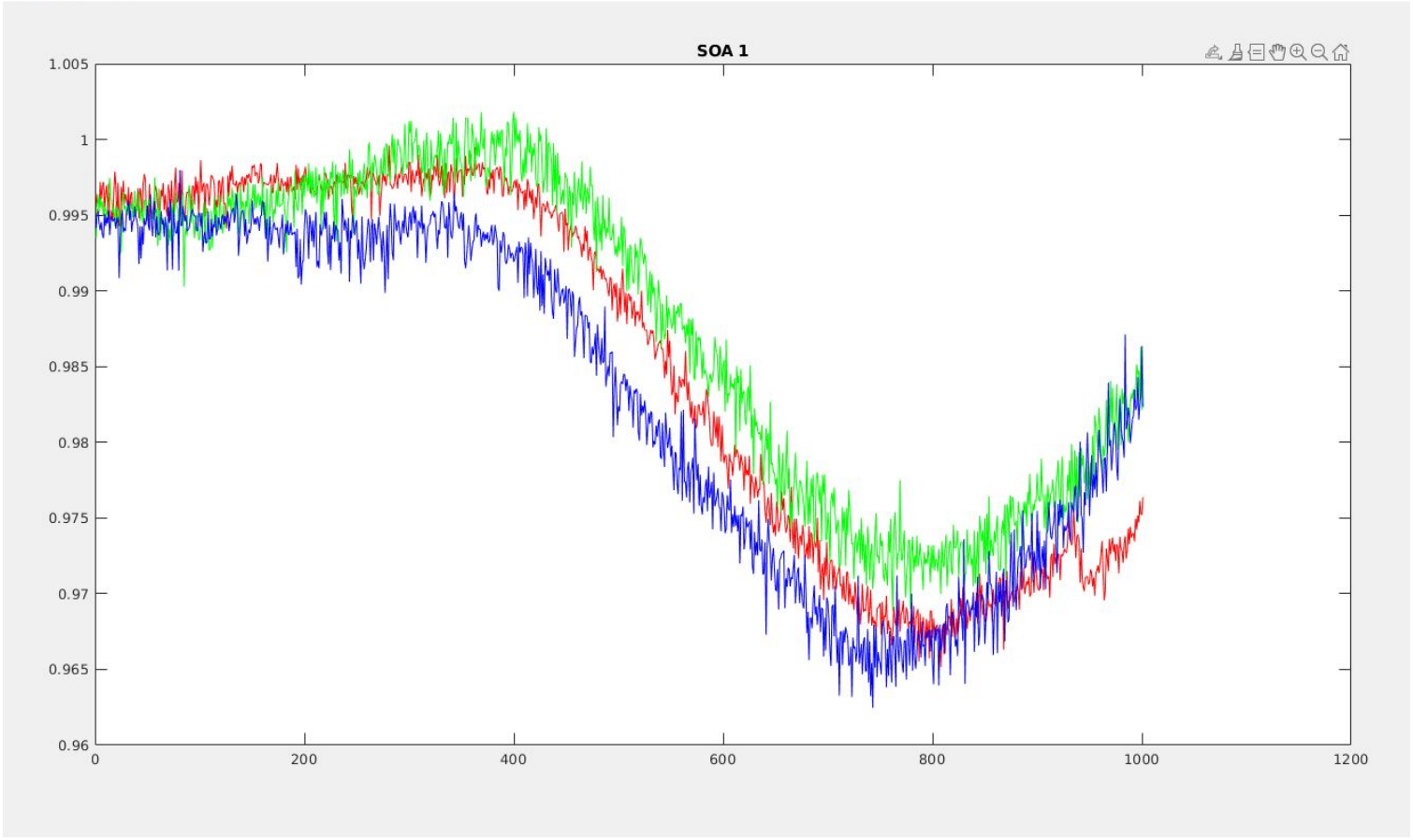
What to do :

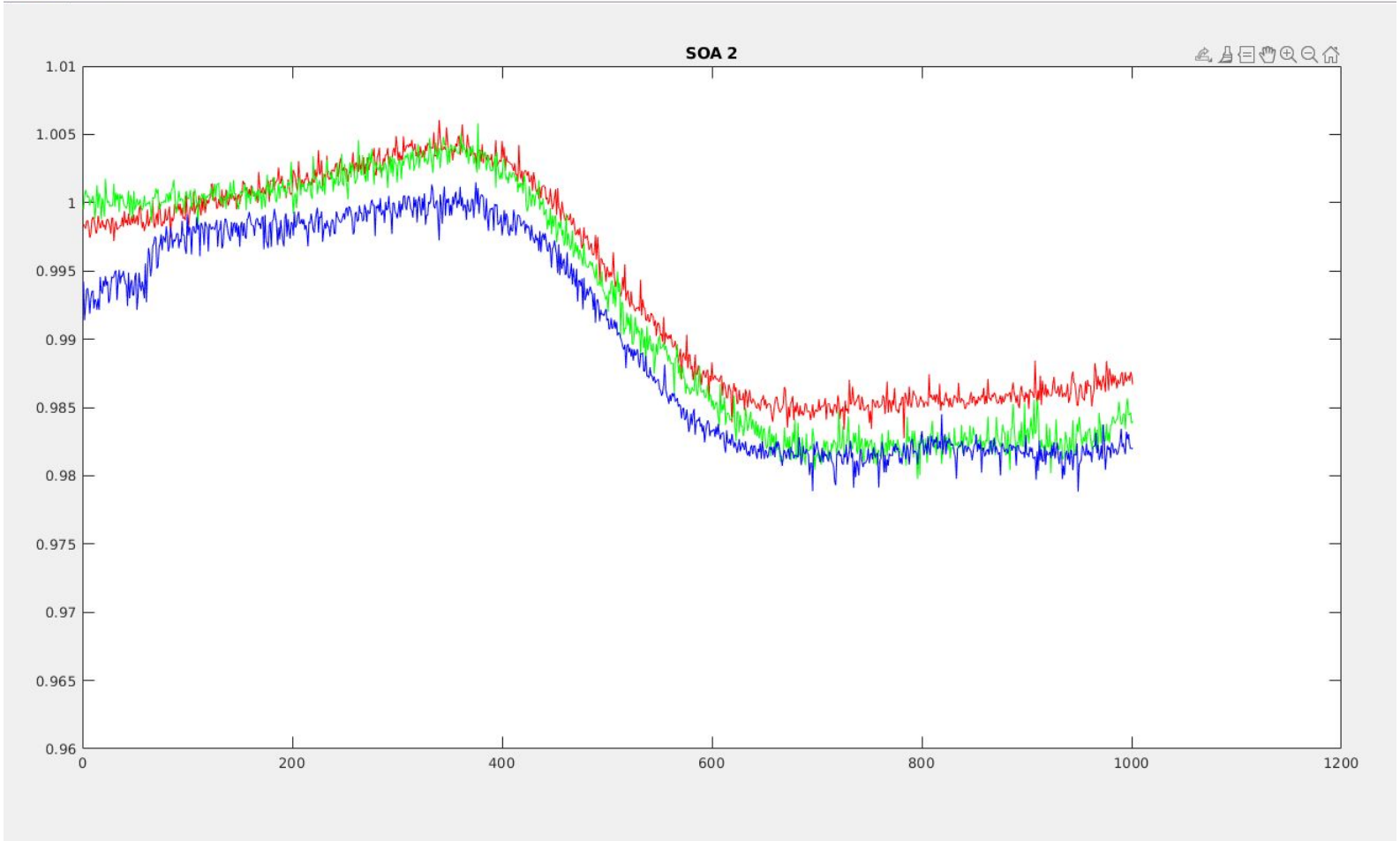
In this task we need to normalize the values:

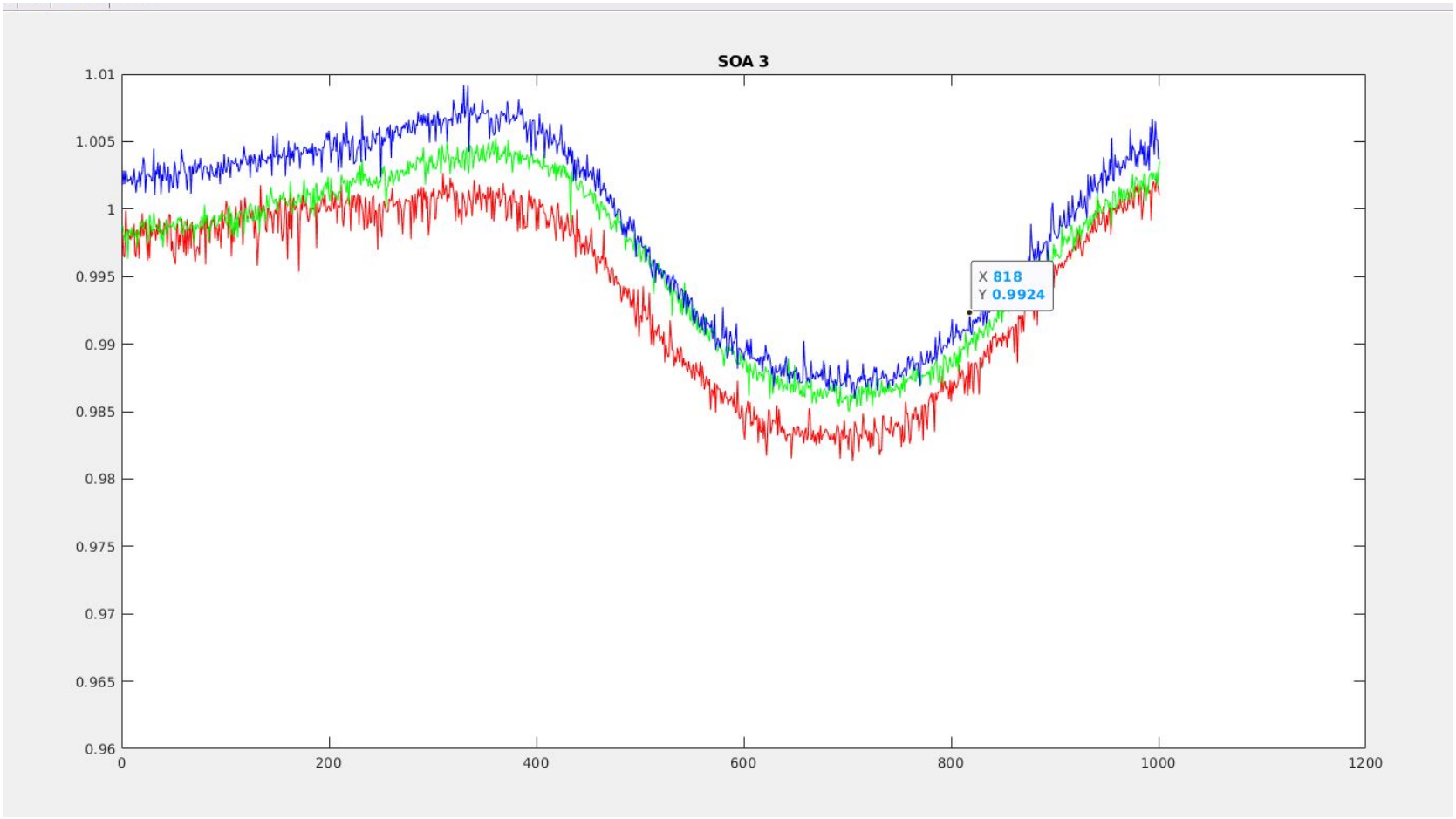
I did this by taking the first 250 values of PD taking mean of it then dividing each index of that 1001 matrix with the value that will for each DataEvent and for each Participant as well

Then made changes to temp:

1. There were data events for which no PD stored only zero was stored so ignore that not including that for plotting.
2. Also for particular dataevent there was no DataEvent for whihc second target is in perpendicualr direction


SOA 1
1.005
1
0.995
0.99
0.985
0.98
0.975
0.97
0.965
0.96
0
200
400
600
800
1000
1200


SOA 2
1.01
1.005
1
0.995
0.99
0.985
0.98
0.975
0.97
0.965
0.96
0
200
400
600
800
1000
1200

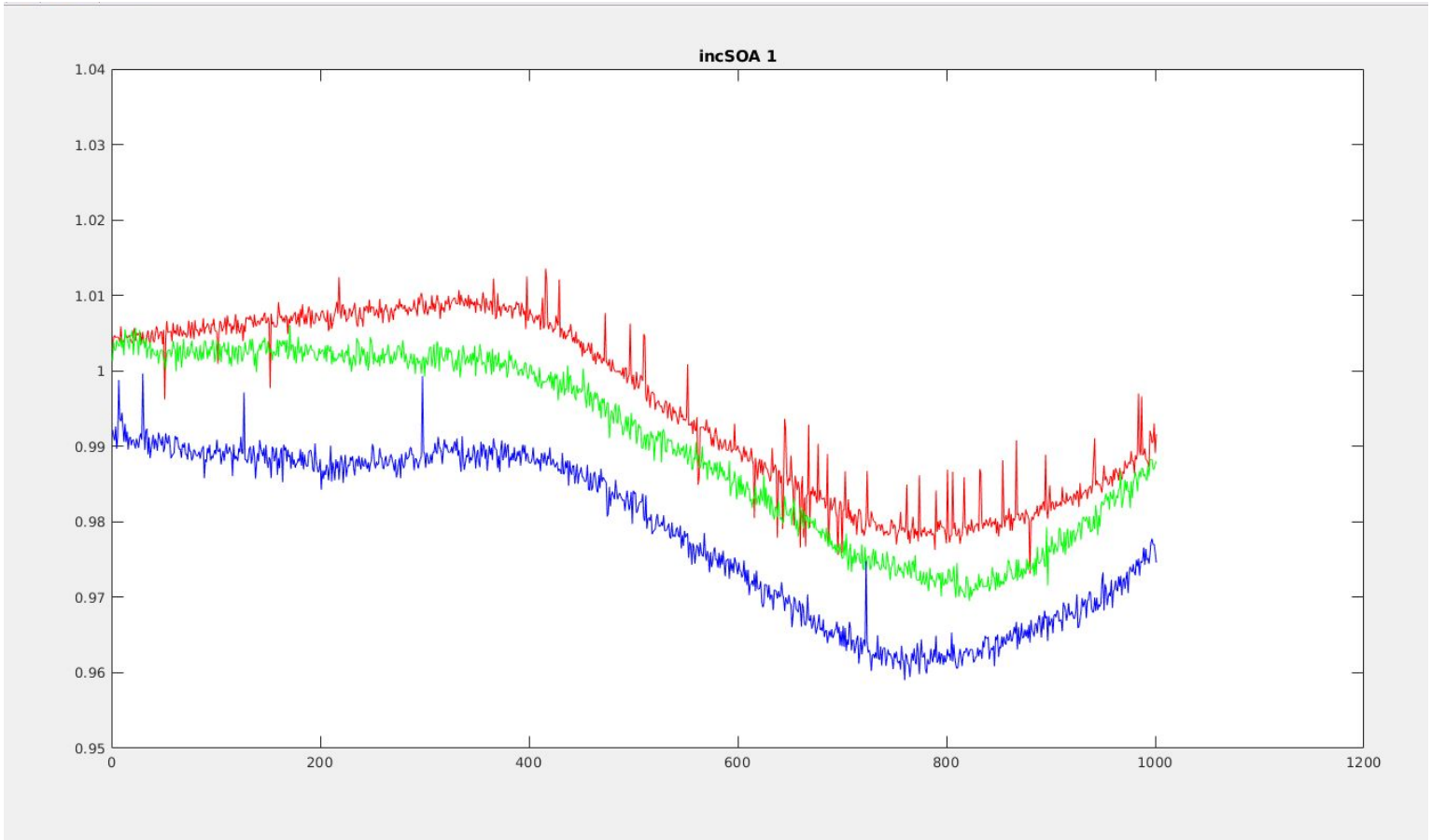

SOA 3
1.01
1.005
1
0.995
0.99
0.985
0.98
0.975
0.97
0.965
0.96
0
200
400
600
800
1000
1200
X 818
Y 0.9924

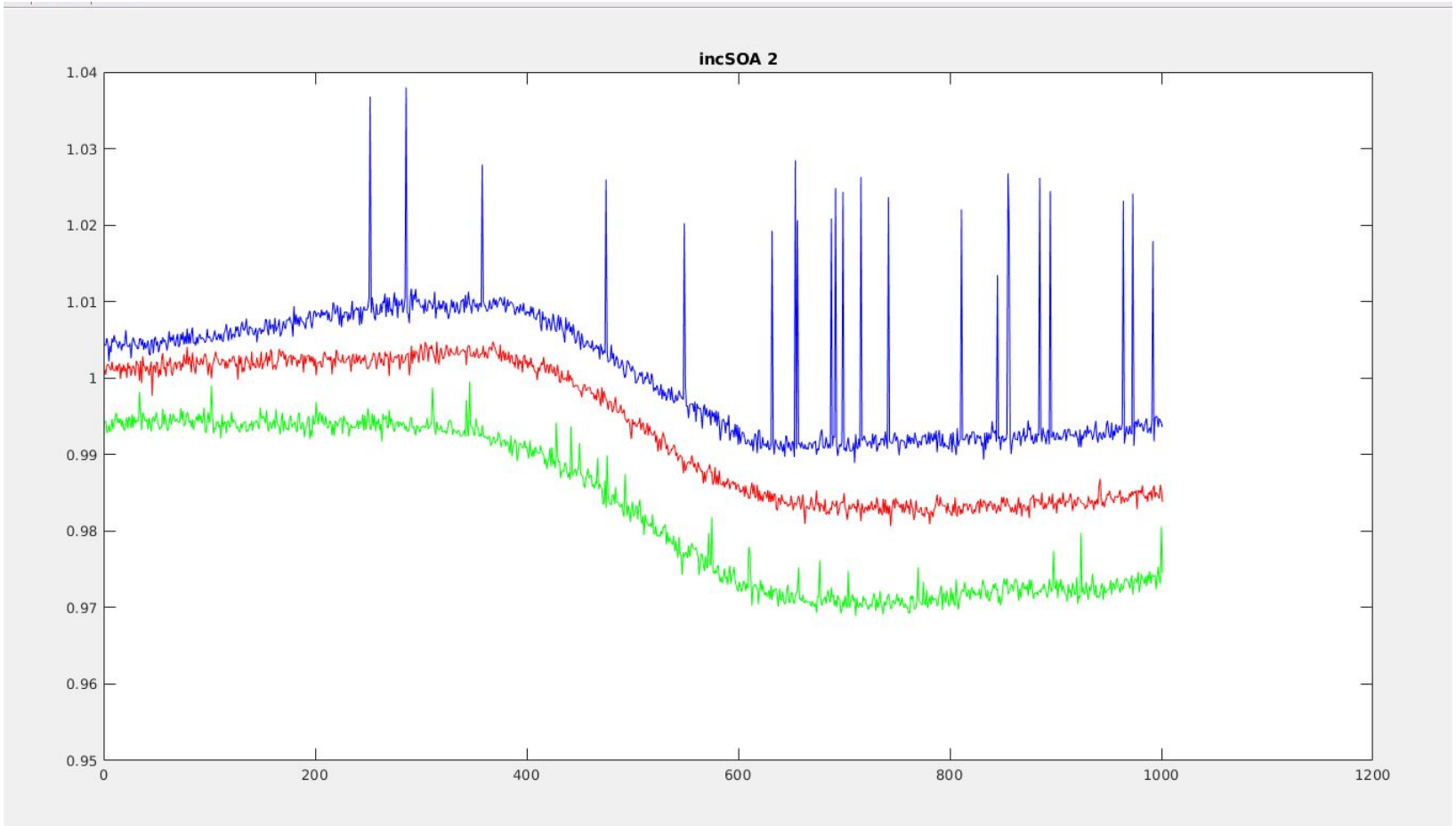
New Task

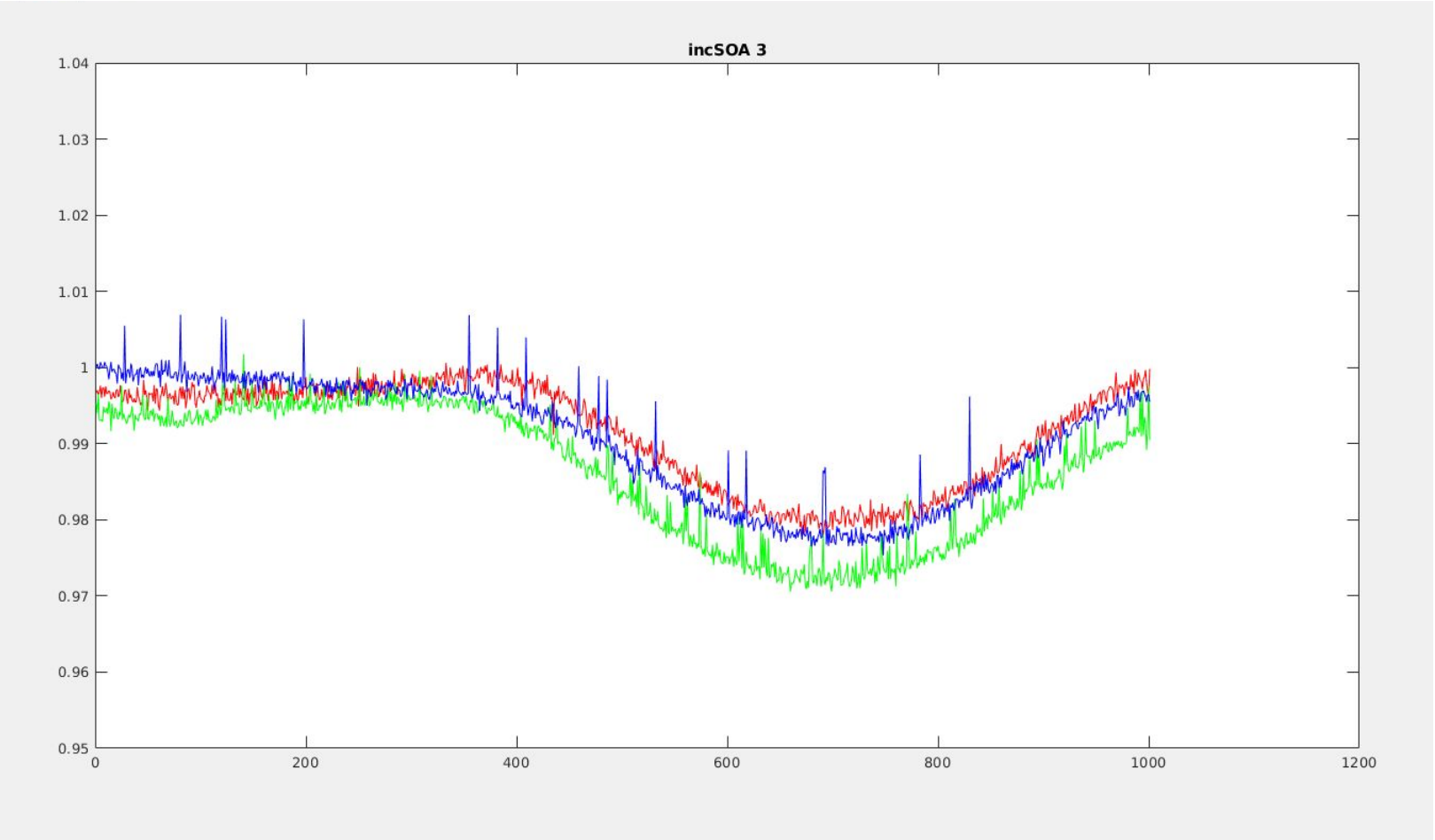
What to do:

Make 3 more graphs:

1. 3 graphs would be like I made for correct response. But this time this would be for incorrect trials. A trial would be added in this because of 2 cases:
   a. If the person says there is no second target but there is
   b. If the person gives the direction of second target wrong


incSOA 1
1.04
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
0.95
0
200
400
600
800
1000
1200


incSOA 2
1.04
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
0.95
0
200
400
600
800
1000
1200

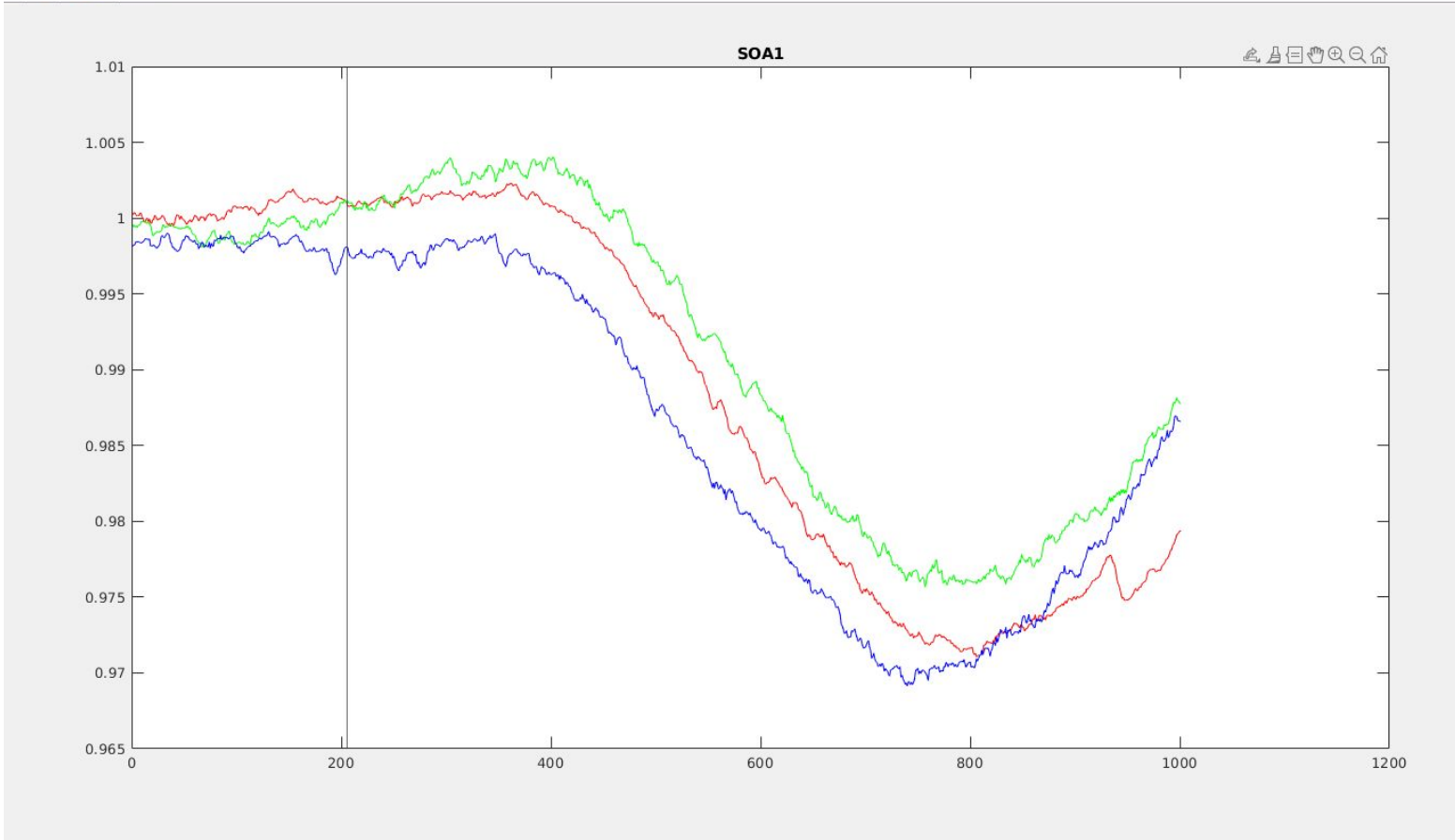

incSOA 3
1.04
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
0.95
0
200
400
600
800
1000
1200

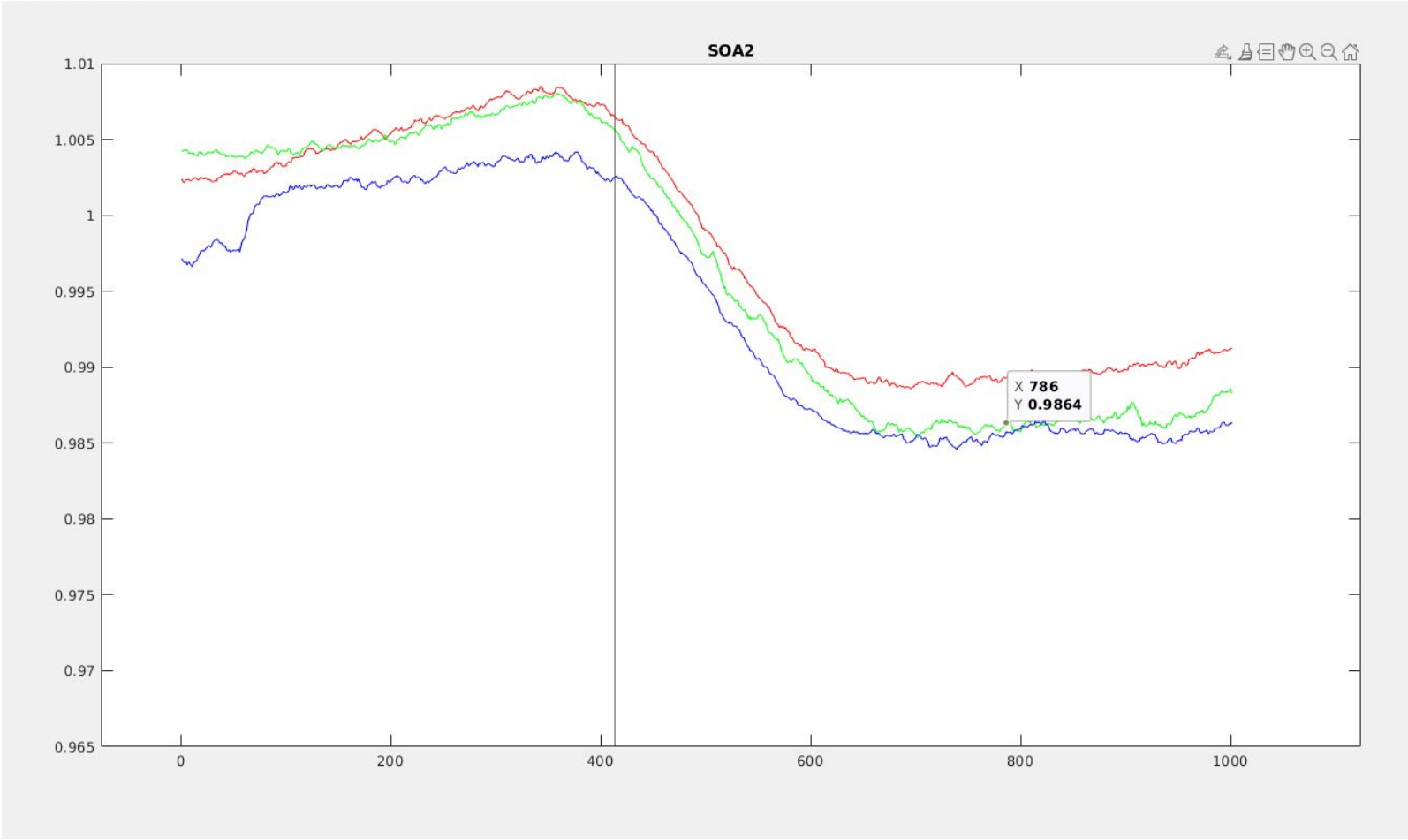
New Task

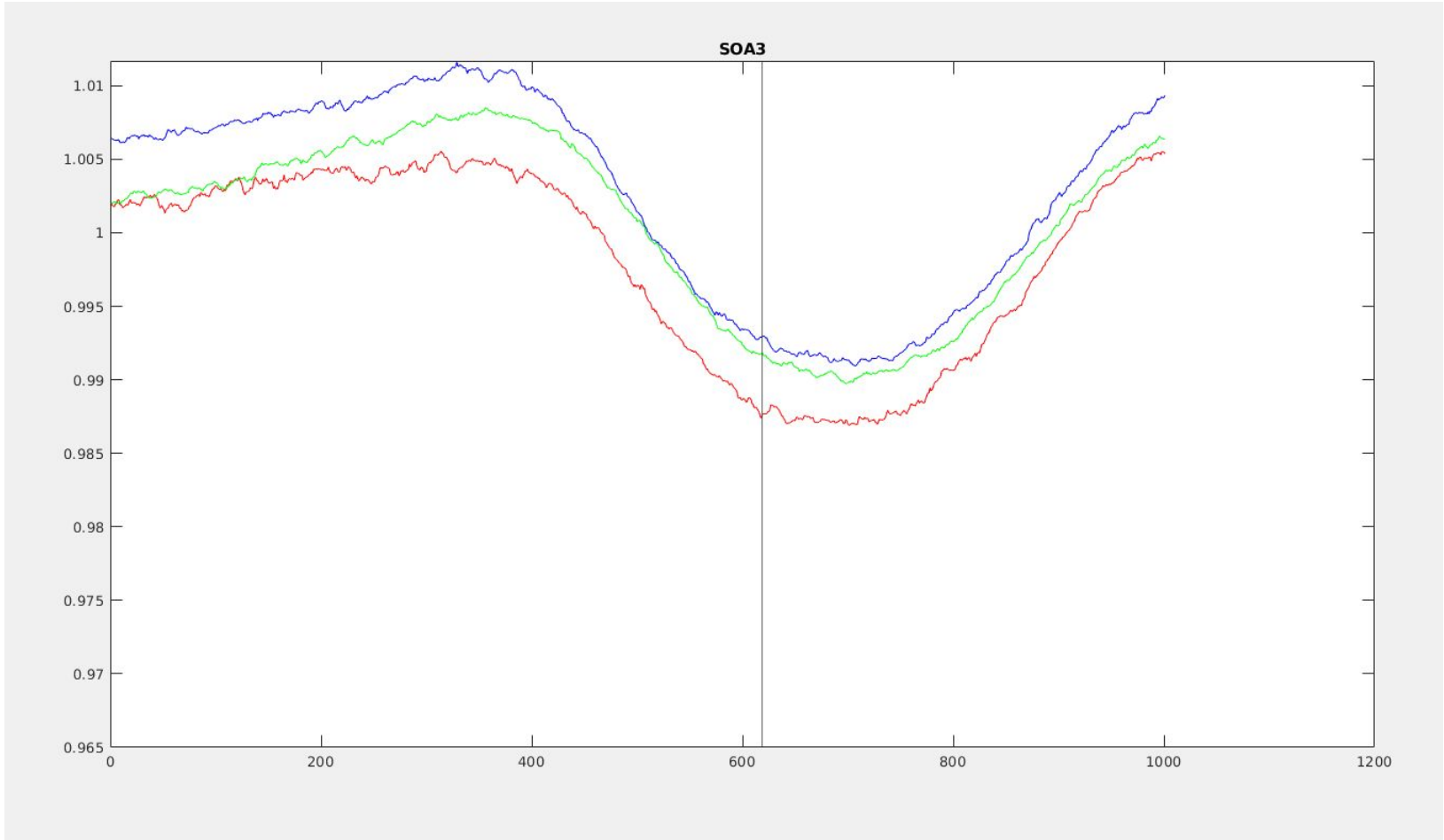
What to do:

1. Draw a vertical line representing the onset of T2, for different SOA's it will be different
   a. SOA1 - onset for T2 starts at 165/0.8=206
   b. SOA2 - onset for T2 starts at 330/0.8=413
   c. SOA3 - onset for T2 starts at 495/0.8=619

2. Smoothing of graph using convolution : What I did is used a inbuilt function conv for doing this part taking the filter value as 01*ones(1,10) and multiplying each value of PD with it and then plotting the graph

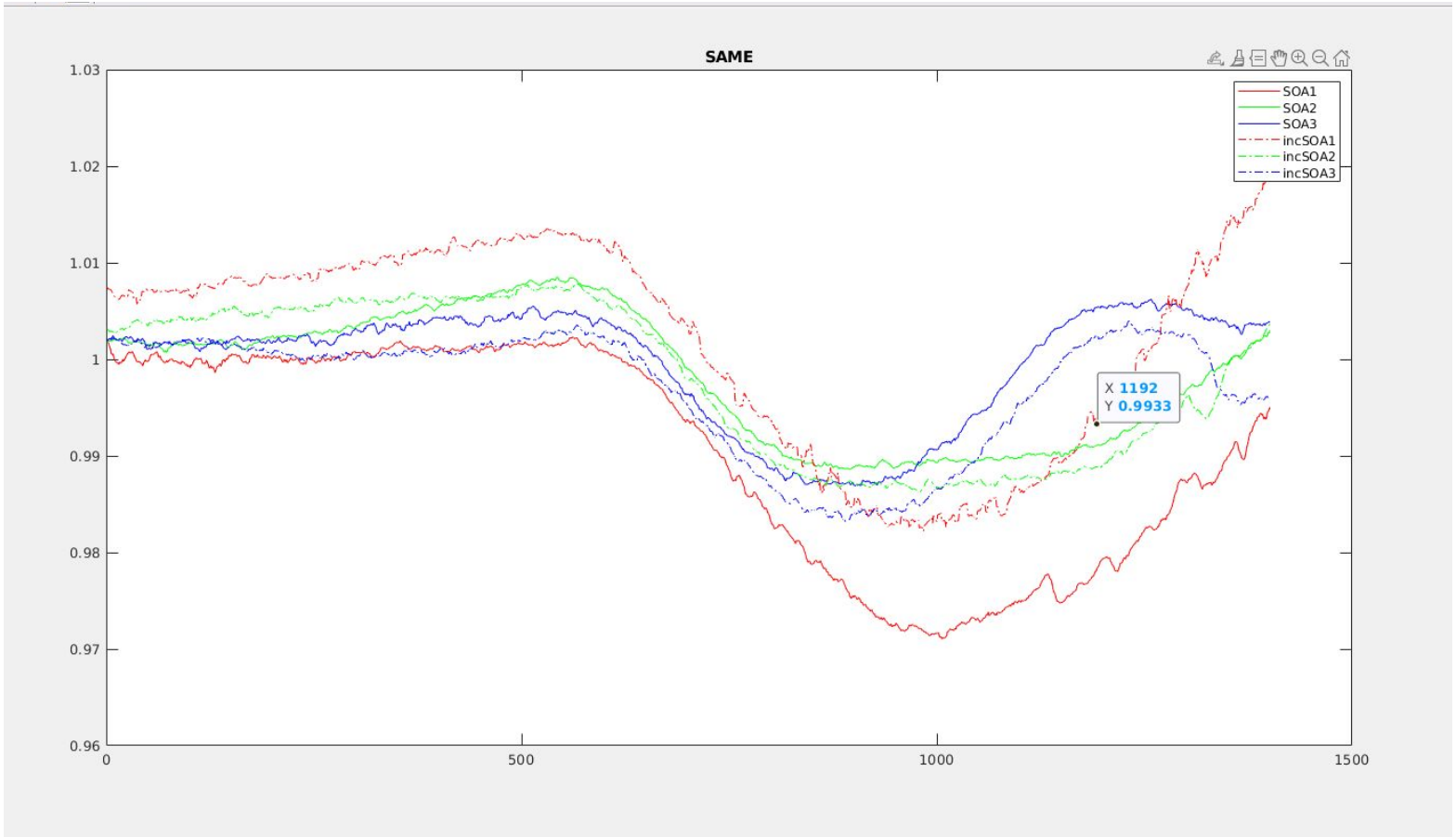

SOA1
1.01
1.005
1
0.995
0.99
0.985
0.98
0.975
0.97
0.965
0
200
400
600
800
1000
1200

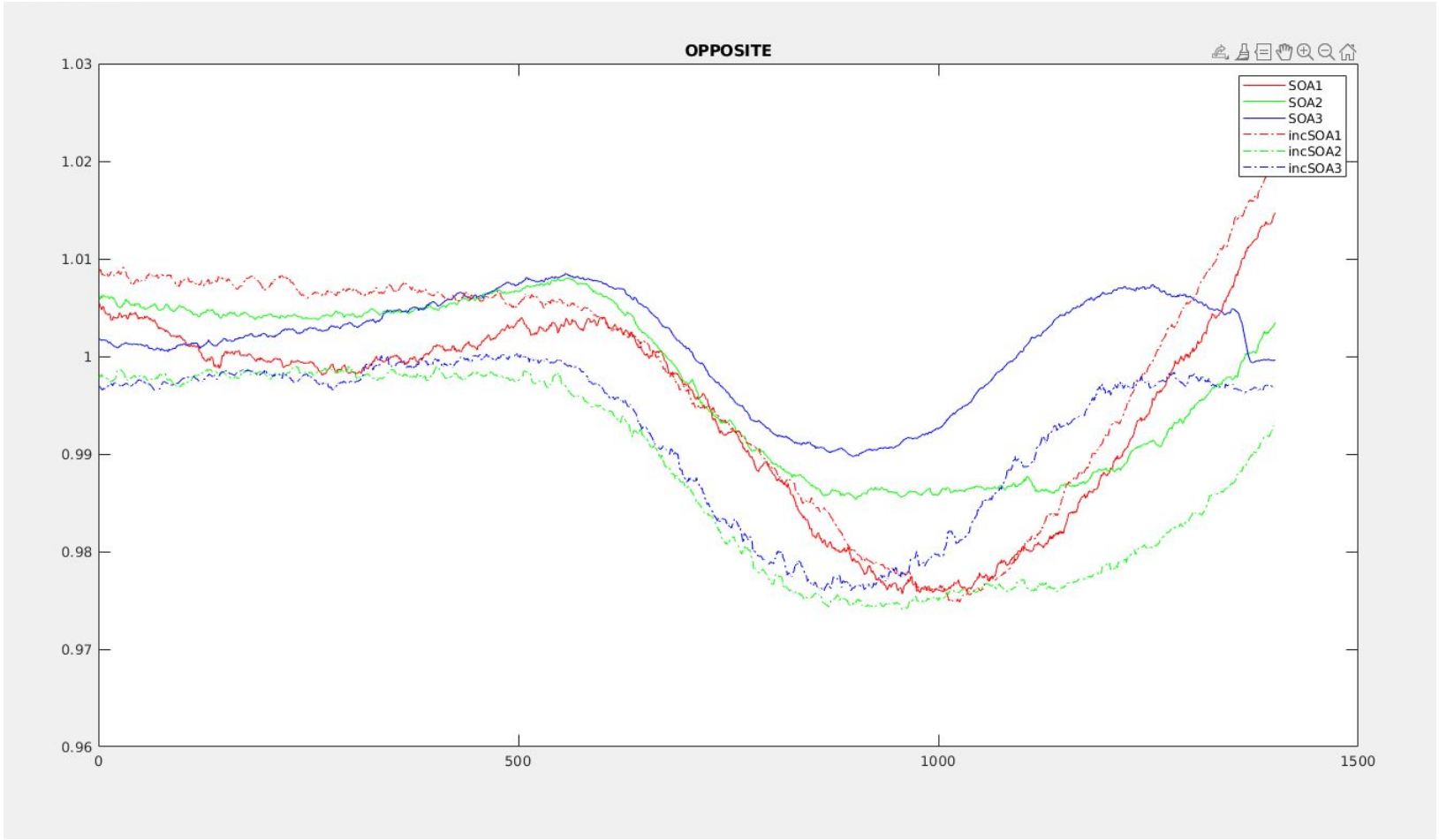

SOA2
1.01
1.005
1
0.995
0.99
0.985
0.98
0.975
0.97
0.965
0
200
400
600
800
1000
X 786
Y 0.9864

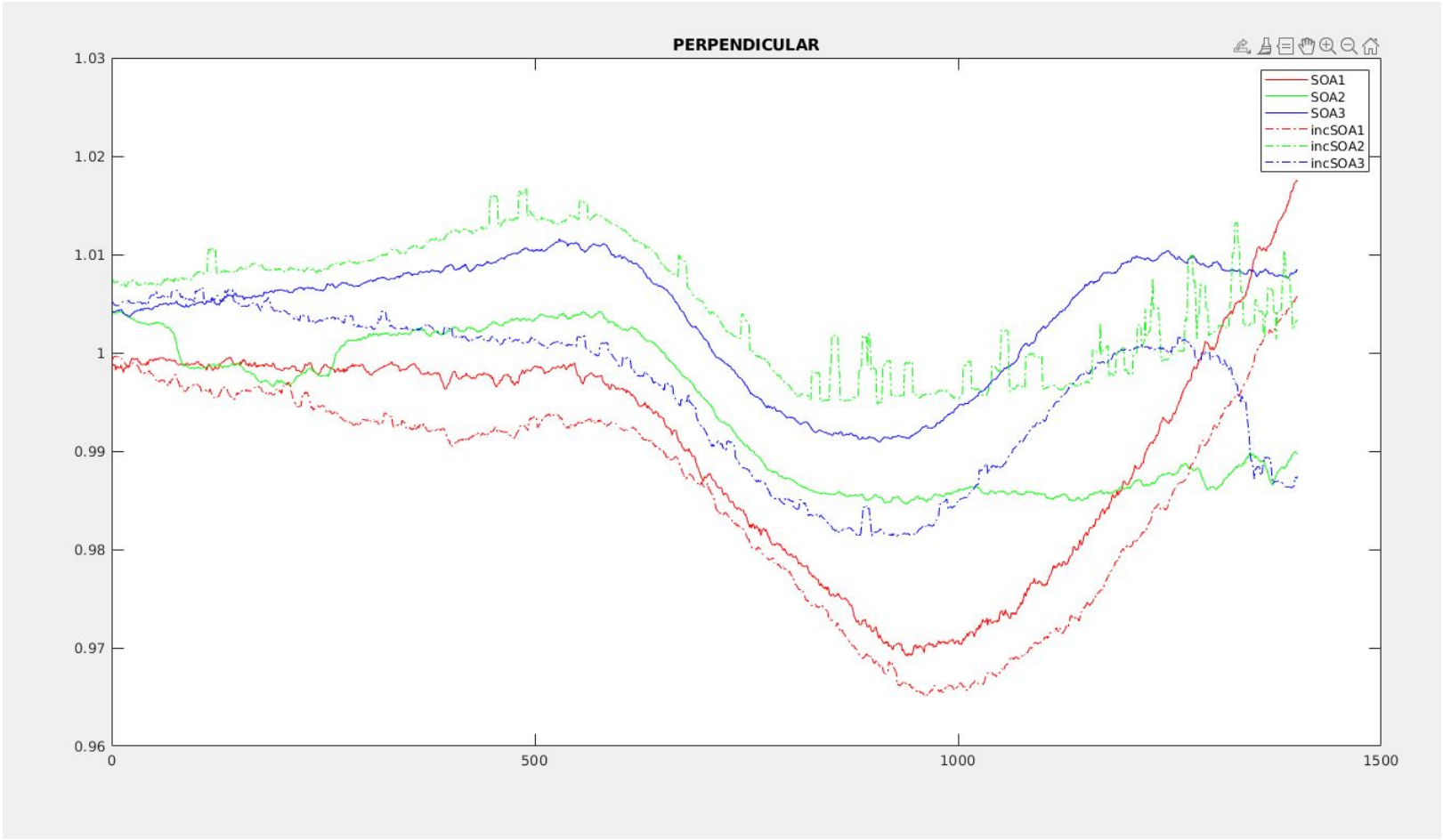

SOA3
1.01
1.005
1
0.995
0.99
0.985
0.98
0.975
0.97
0.965
0
200
400
600
800
1000
1200

TASK for next meeting 24th July 2020:

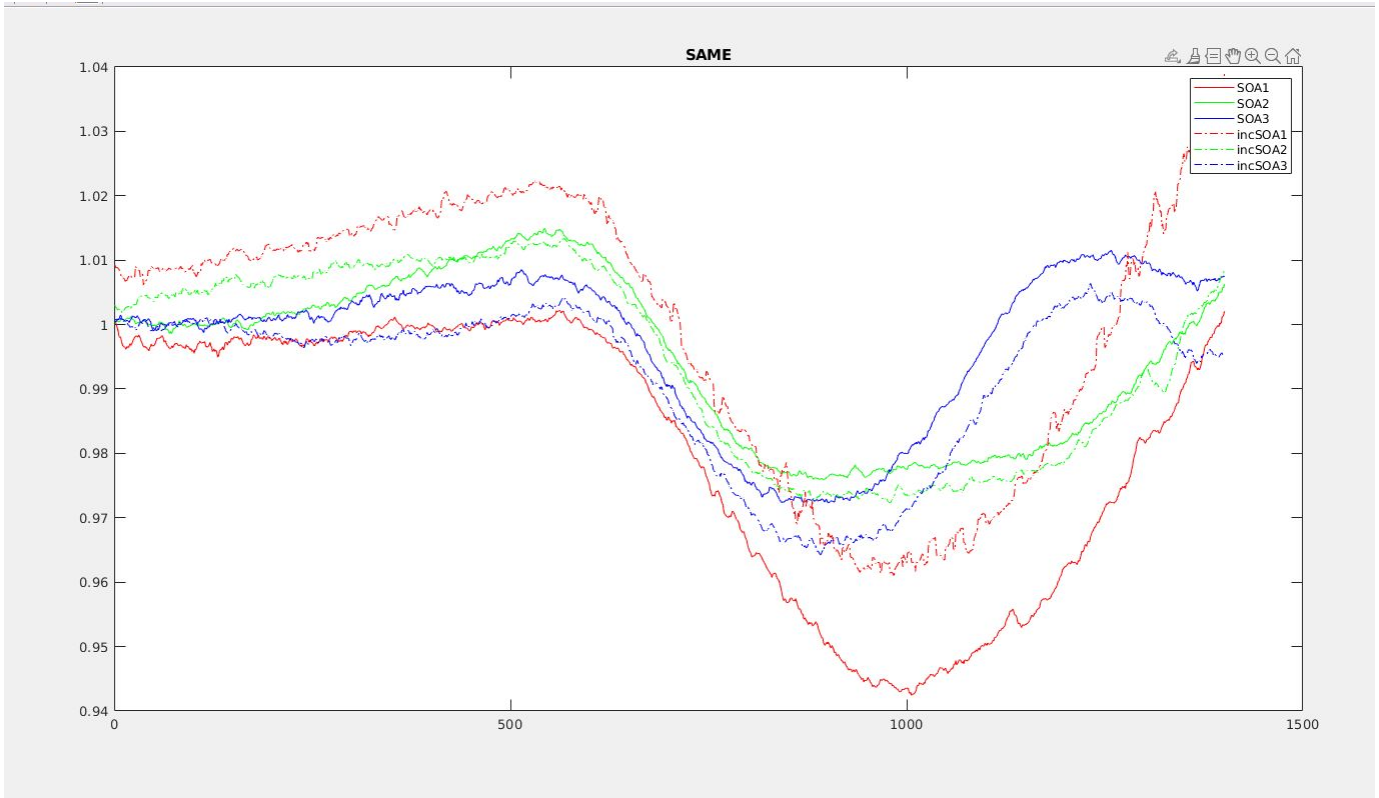
- Plot data for a range of >1000 points, until 2000 points after T1 onset.
- Also plot data for 200points before T1 onset.
- Plot the data with respect to the direction that is
    - Figure 1 should have data for same condition with SOA1, 2 & 3 shown with different curves for correct (solid lines) and incorrect (dashed lines) data trials.
    - Similarly figure 2 for opposite and figure 3 for perpendicular direction.
- Plot data as described above for Pupil area also (which will be present in the field of PA in the data structure)
- Also plot data for condition with No T2: both correct and incorrect trials (we discussed this long back)

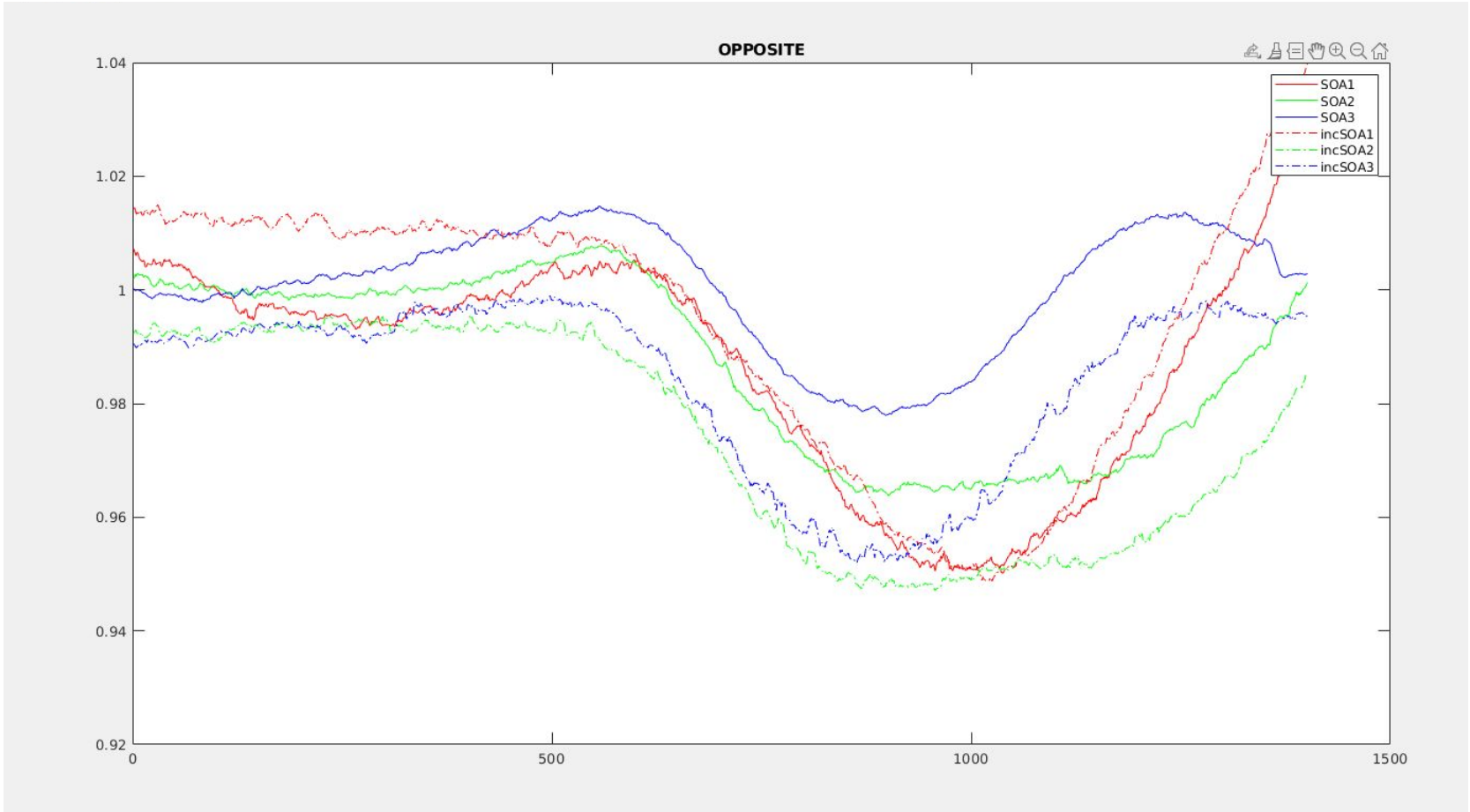

SAME
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
X 1192
Y 0.9933
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
0
500
1000
1500

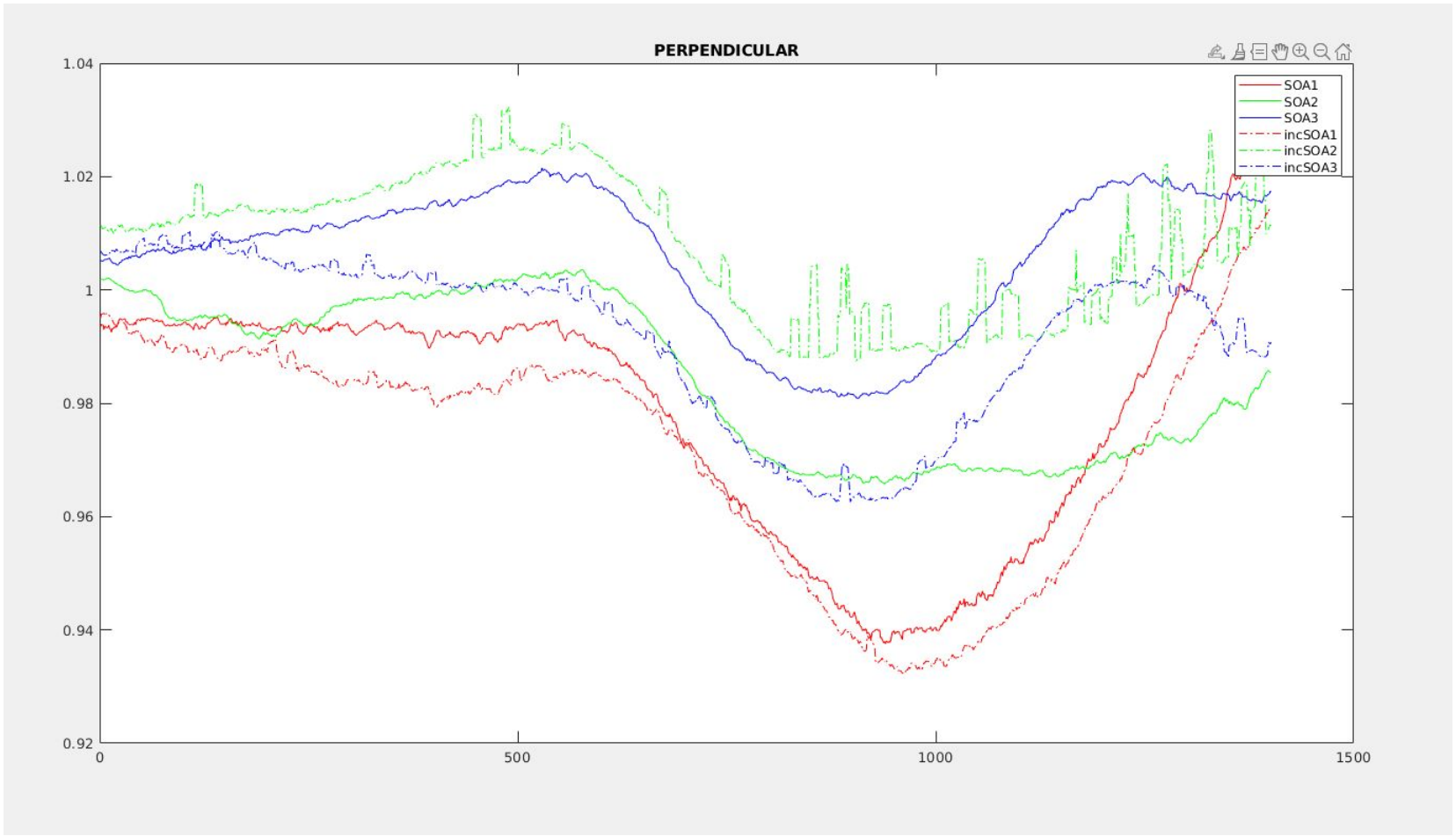

OPPOSITE
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
0
500
1000
1500


PERPENDICULAR
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
1.03
1.02
1.01
1
0.99
0.98
0.97
0.96
0
500
1000
1500

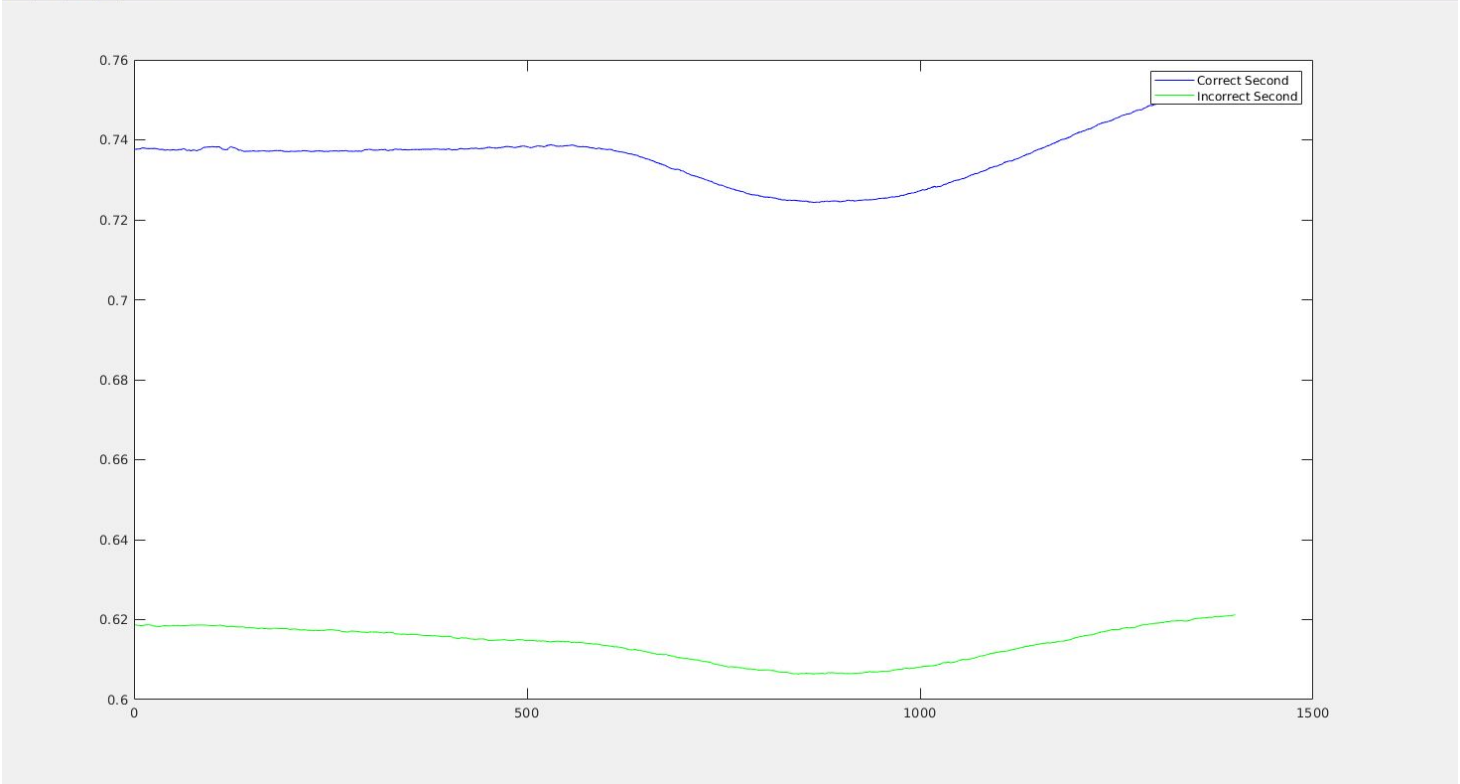
Now graph for PA




OPPOSITE
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
1.04
1.02
1
0.98
0.96
0.94
0.92
0
500
1000
1500


PERPENDICULAR
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
1.04
1.02
1
0.98
0.96
0.94
0.92
0
500
1000
1500

For Second Target



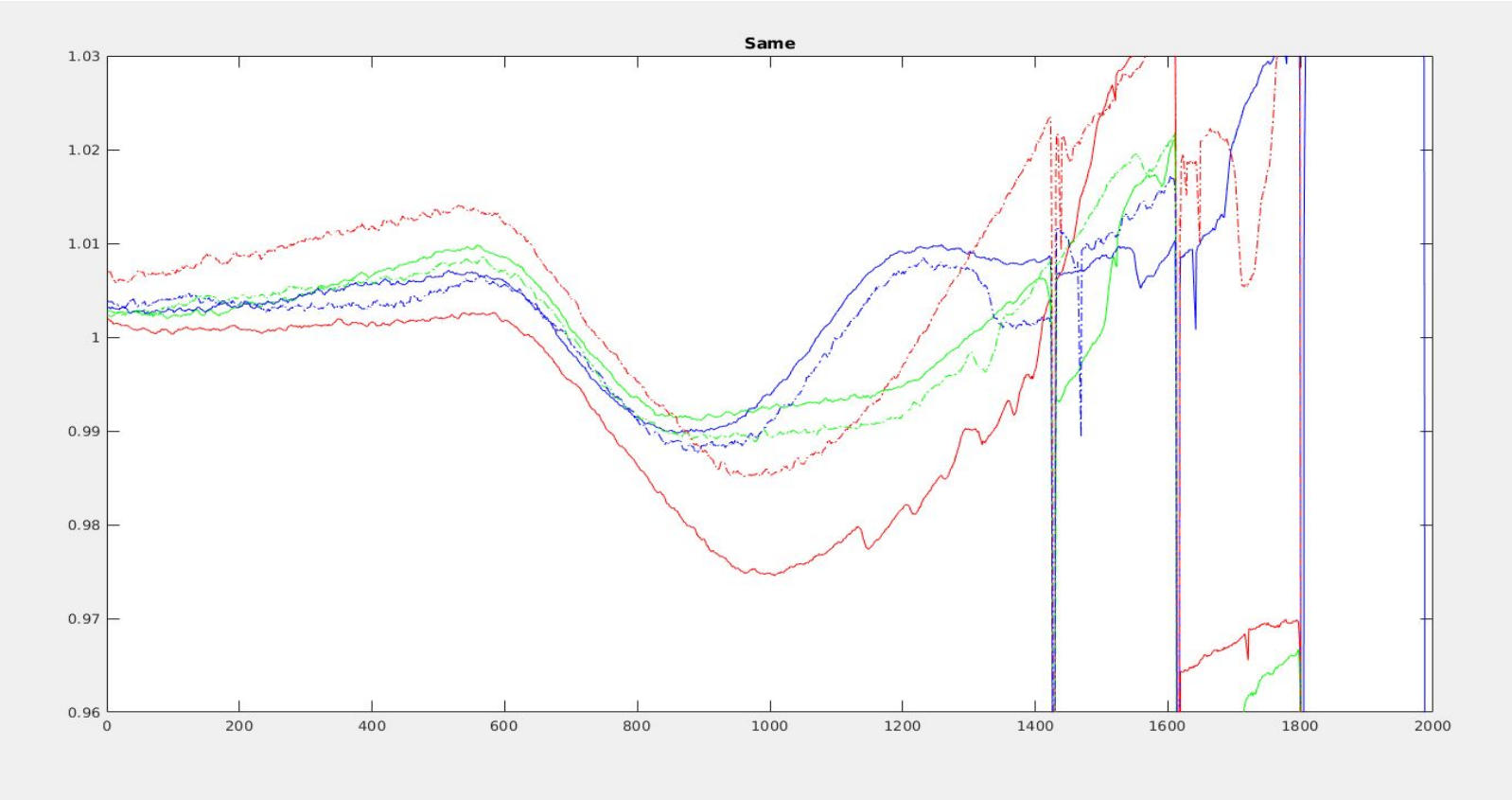
Next Steps:

1. Plot the same graphs as in from slide 27-29 (PD) by averaging in the way I explained, so that we can include all the trials for time points 200 points before T1 Onset to 2000 points after T1 onset.
2. For slides 27-29 re-plot the error trials only when the second answer (yes/No) is incorrect.
3. Share the names of participants (N = 14) that were included in the analysis.
4. Also plot the error trials for perpendicular condition especially for SOA 3 for all participants to point out the noisy participant and remove them.
5. Plot vertical line marking onset of T1 and Soa1, soa2, soa3 (color code them)

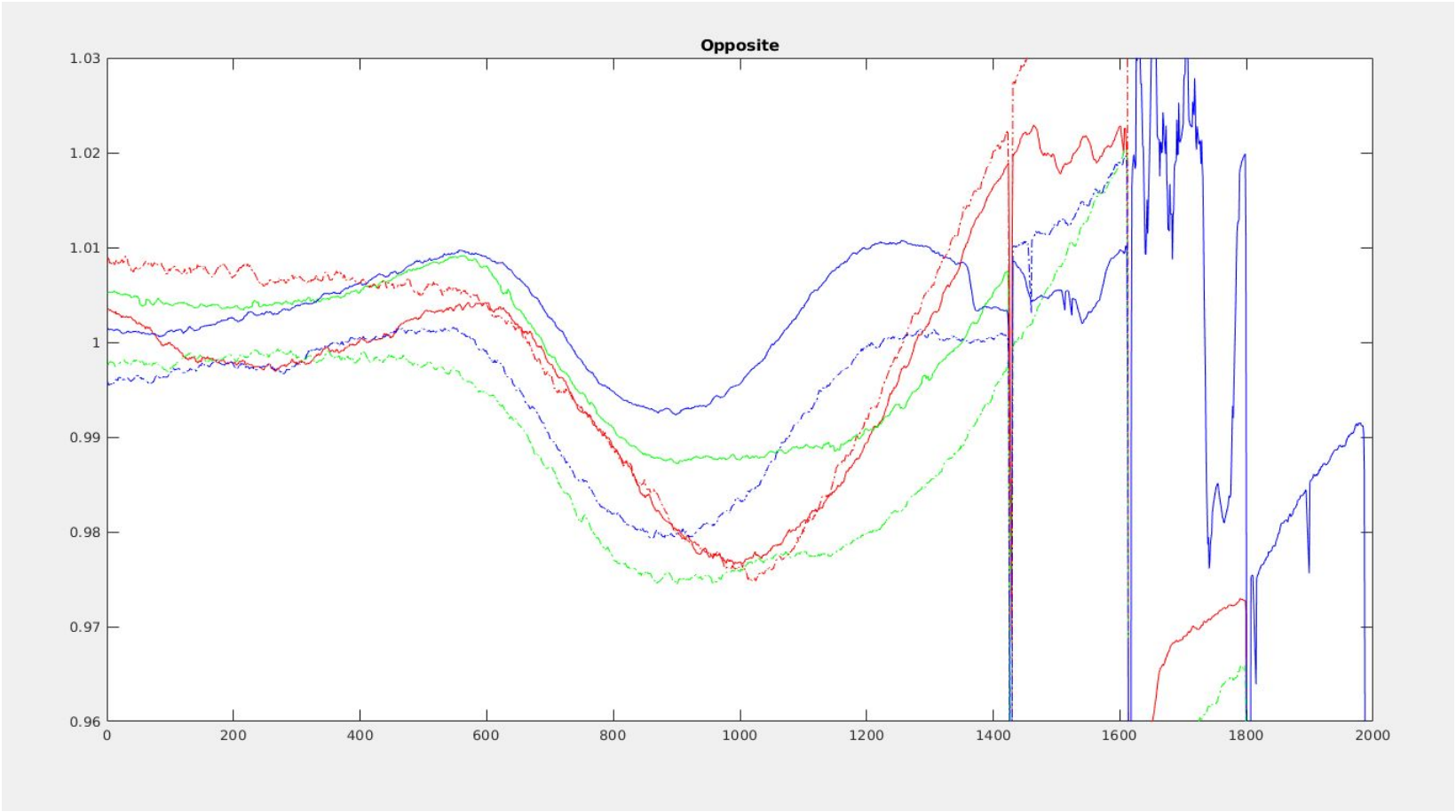
Participant Including (Name, participant_number) :

1. Mayank → 1
2. Swarnima → 2
3. Diksha → 3
4. RahulK → 4
5. Devesh → 5
6. Vivek → 7
7. SatyapalY → 10
8. Deepika → 11
9. Mohit → 12
10. Pragati → 15
11. Pragya Shukla → 16
12. Mridul → 18
13. Milind Baudh → 19
14. Arun → 21
15. Aman Kumar

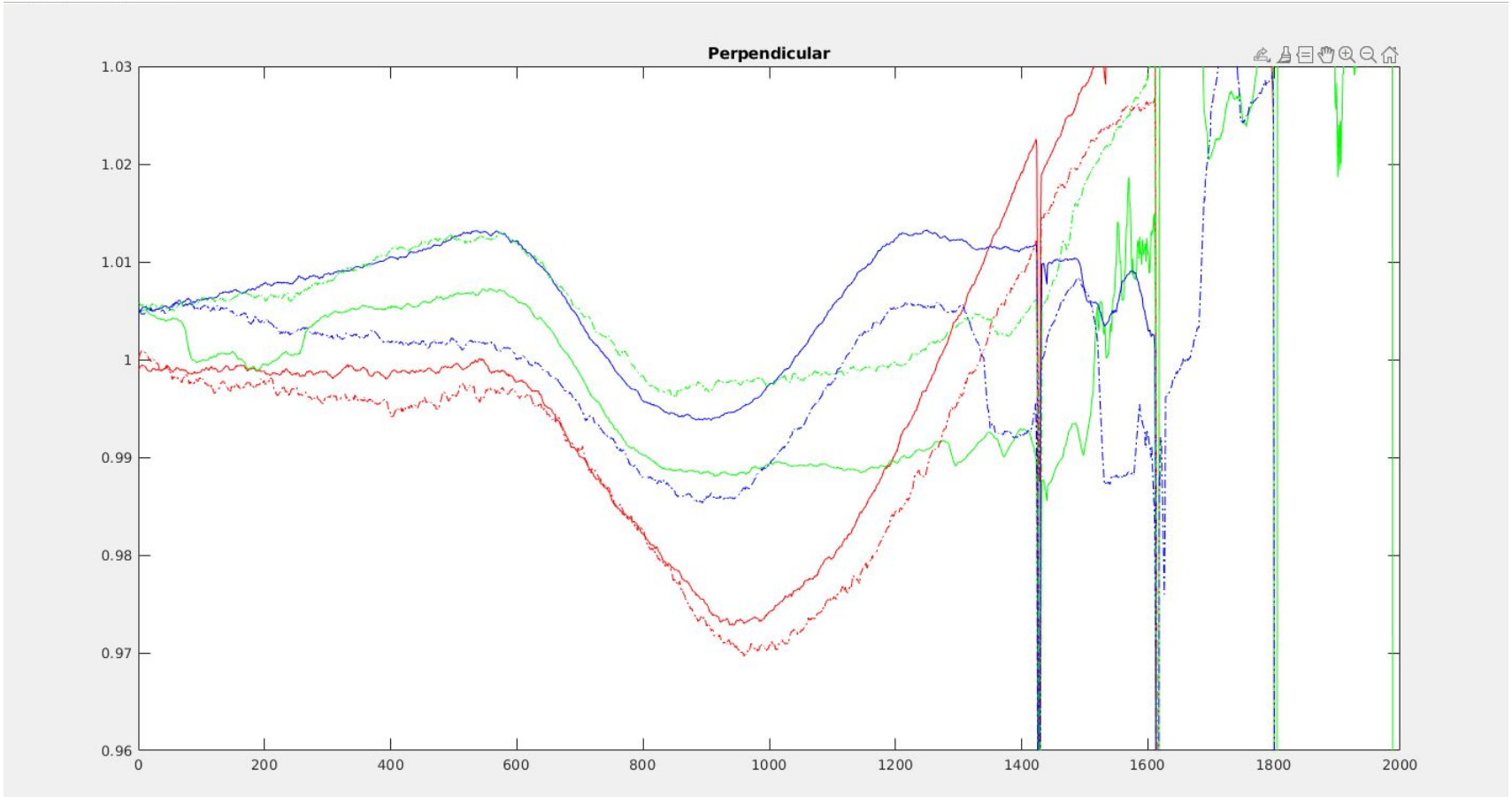
Zoomed Image

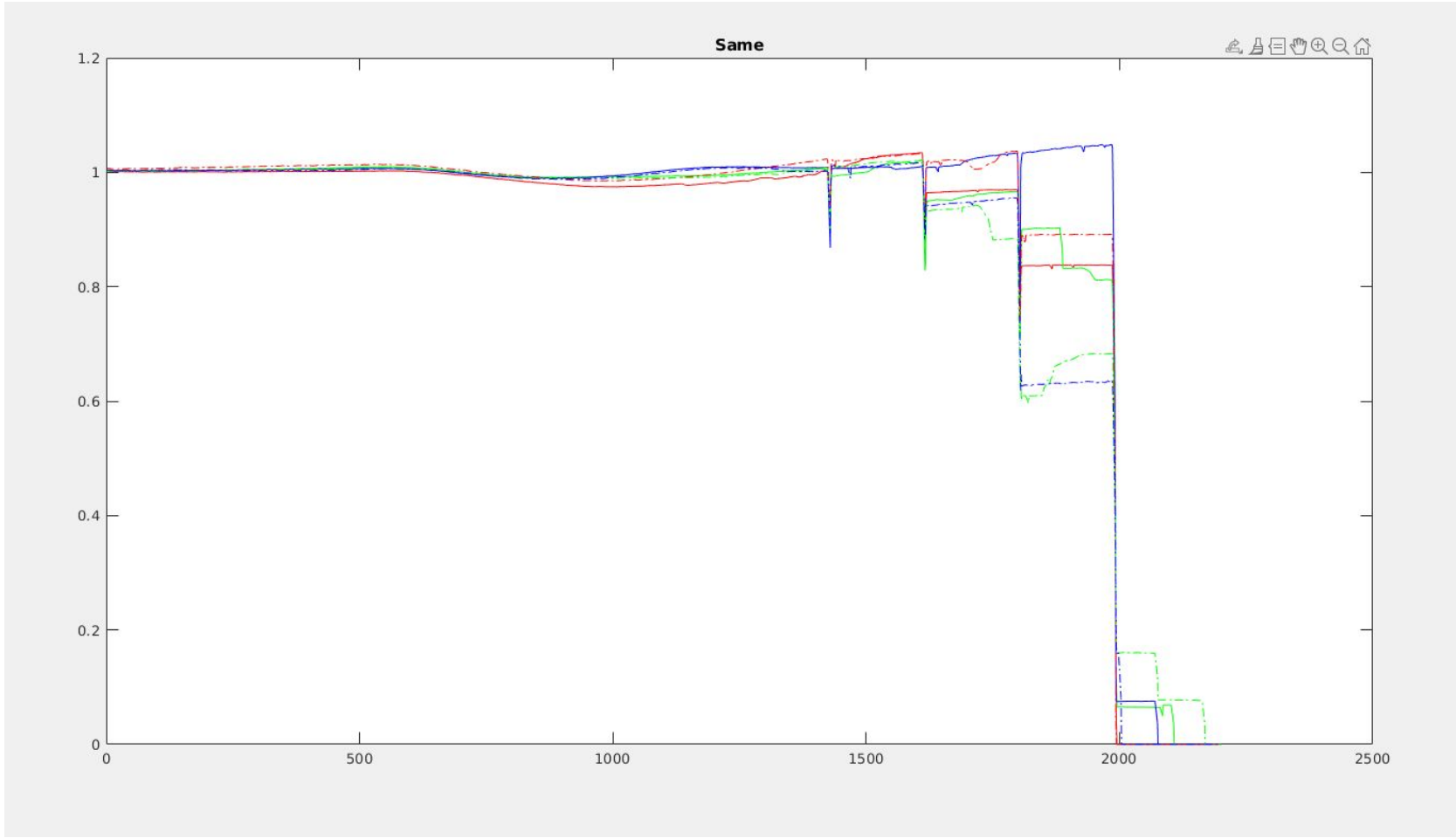


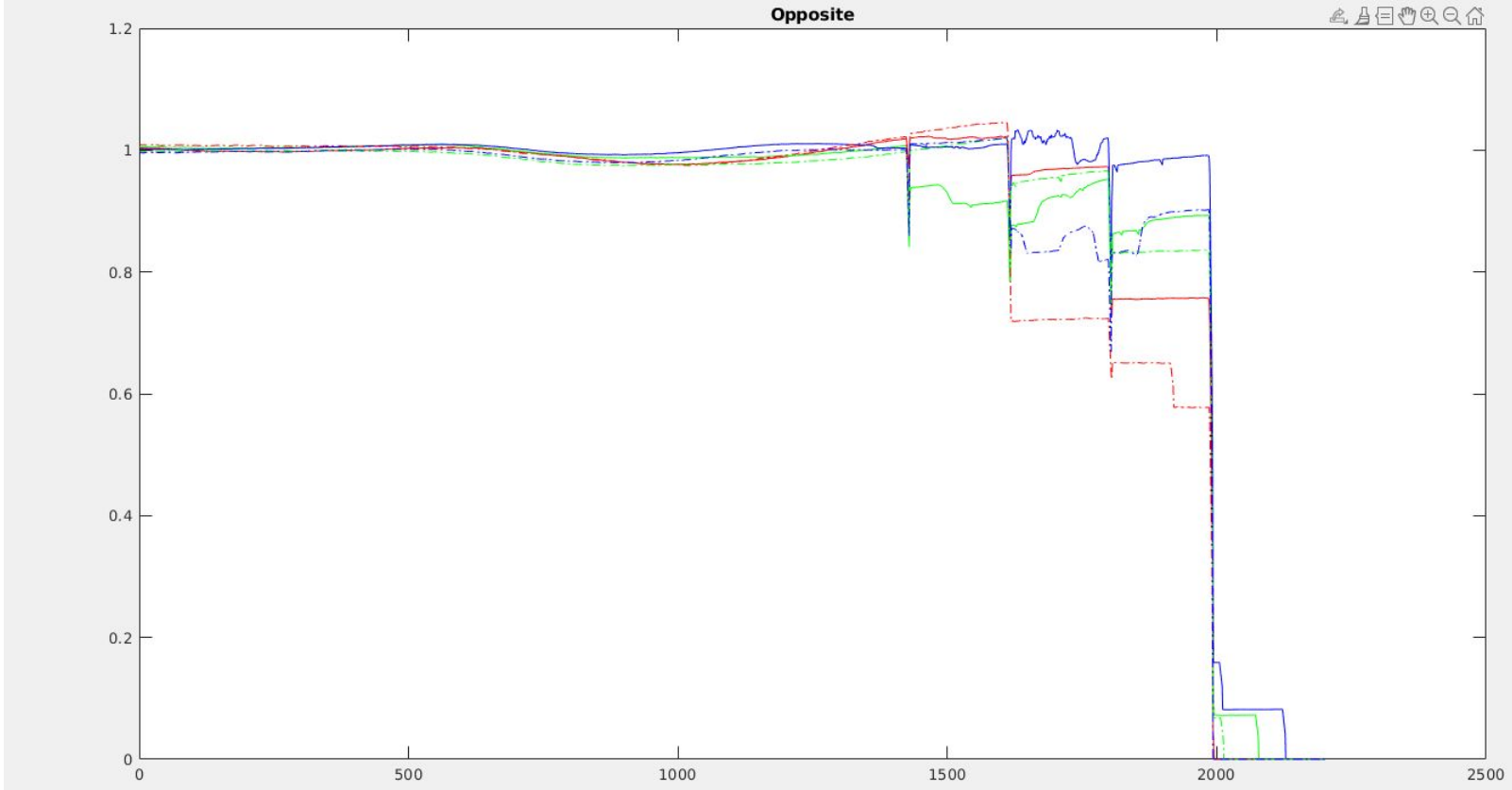
Zoomed Image

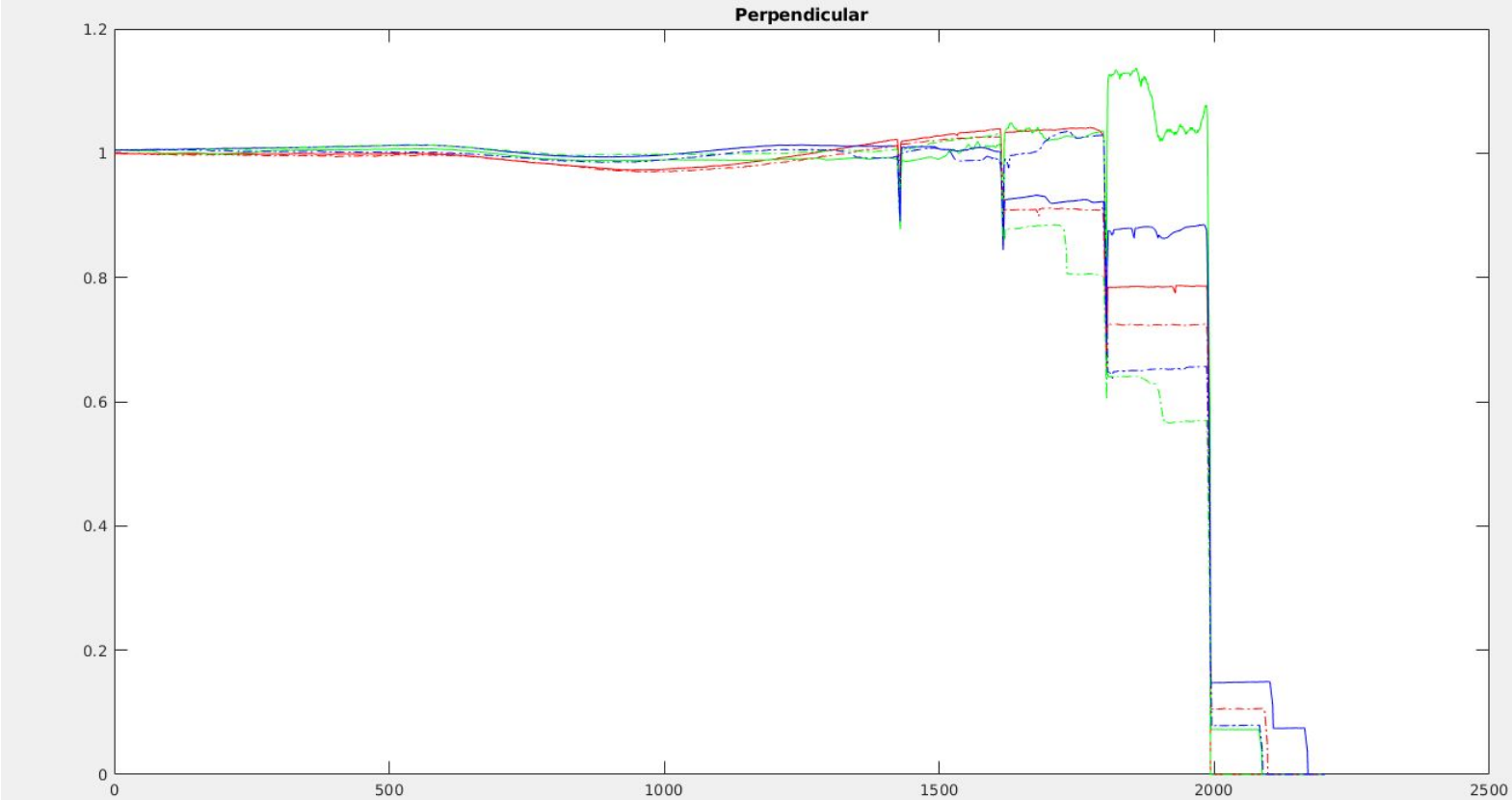


Zoomed Image

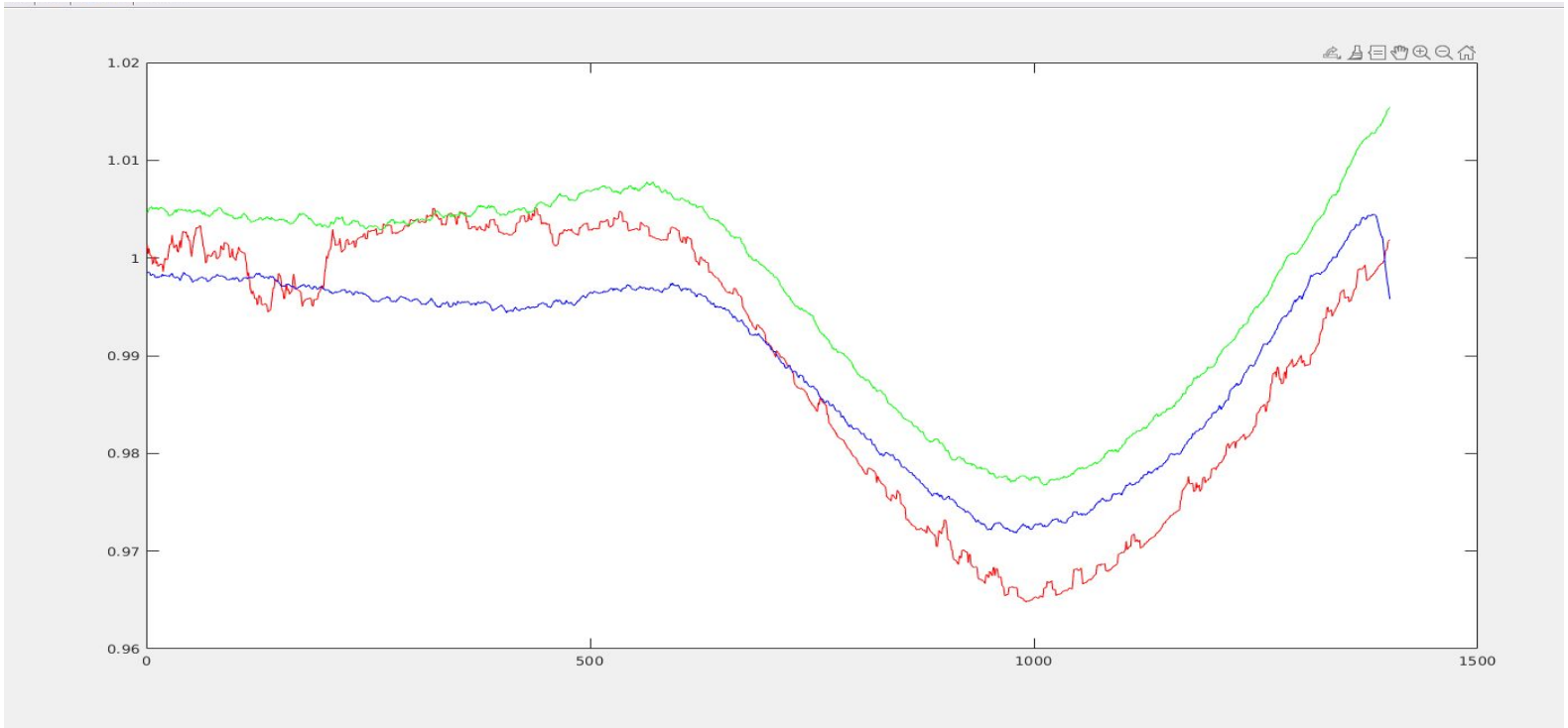


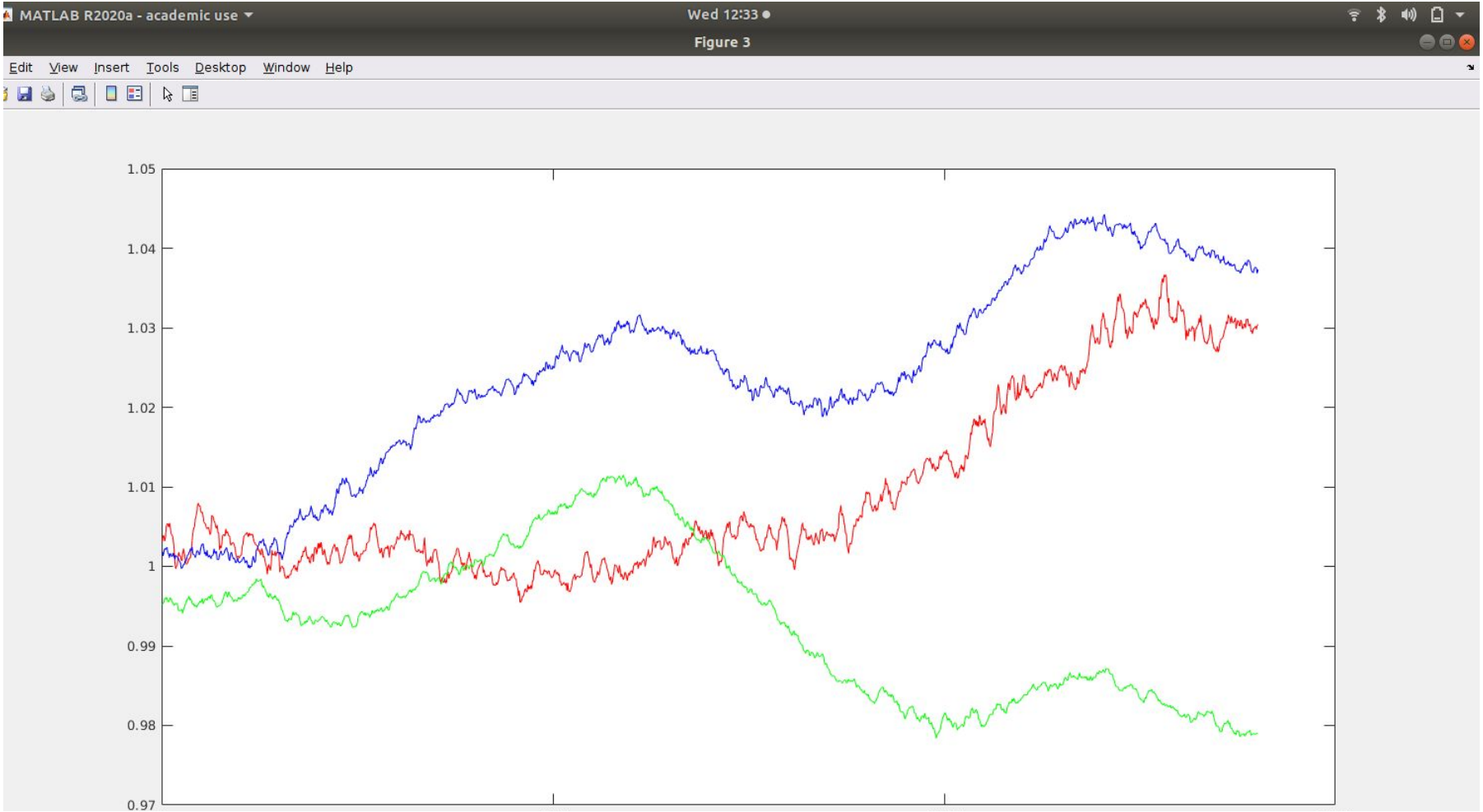

Same
1.2
1
0.8
0.6
0.4
0.2
0
0
500
1000
1500
2000
2500

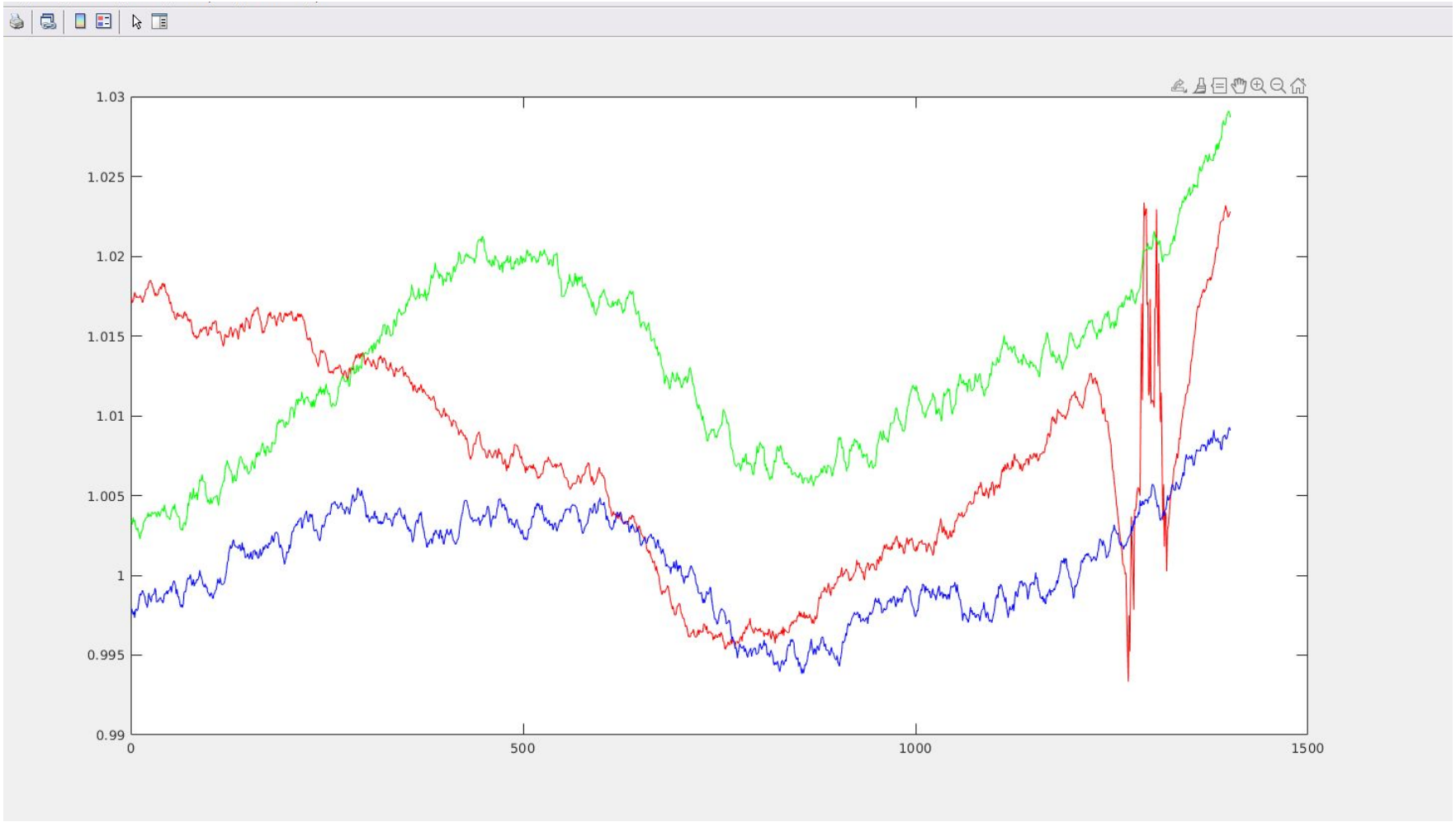

Opposite
1.2
1
0.8
0.6
0.4
0.2
0
0
500
1000
1500
2000
2500


Perpendicular
1.2
1
0.8
0.6
0.4
0.2
0
0
500
1000
1500
2000
2500

As the results for others were not amazing since the no of participants getting that error are really low like 4,5 that's why the result is not that good. This is for SOA1(When the participant doesn't see the second target)
R-Same G-Opposite B-Perpendicular



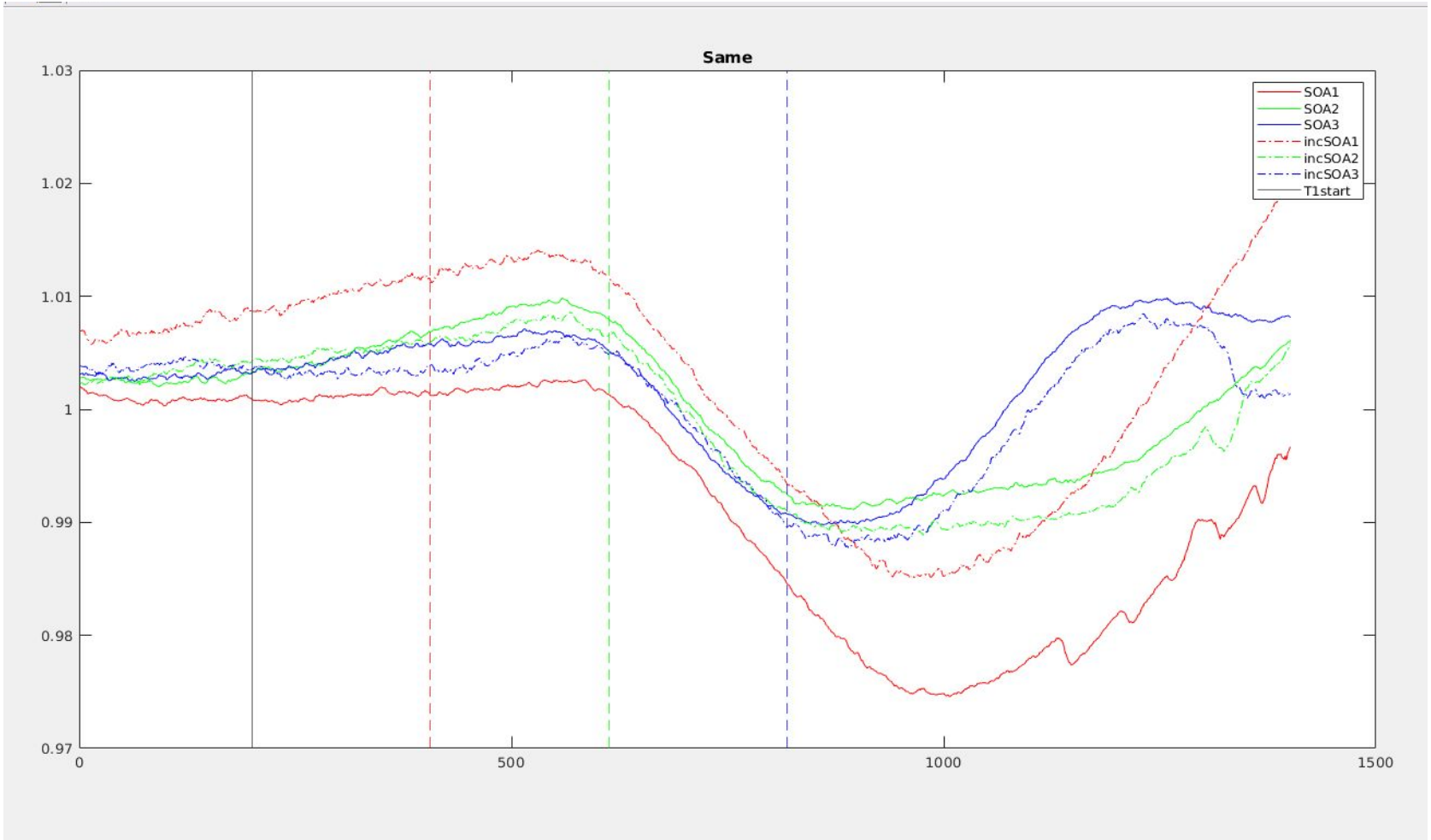

MATLAB R2020a - academic use
Wed 12:33
Figure 3
Edit View Insert Tools Desktop Window Help
1.05
1.04
1.03
1.02
1.01
1
0.99
0.98
0.97

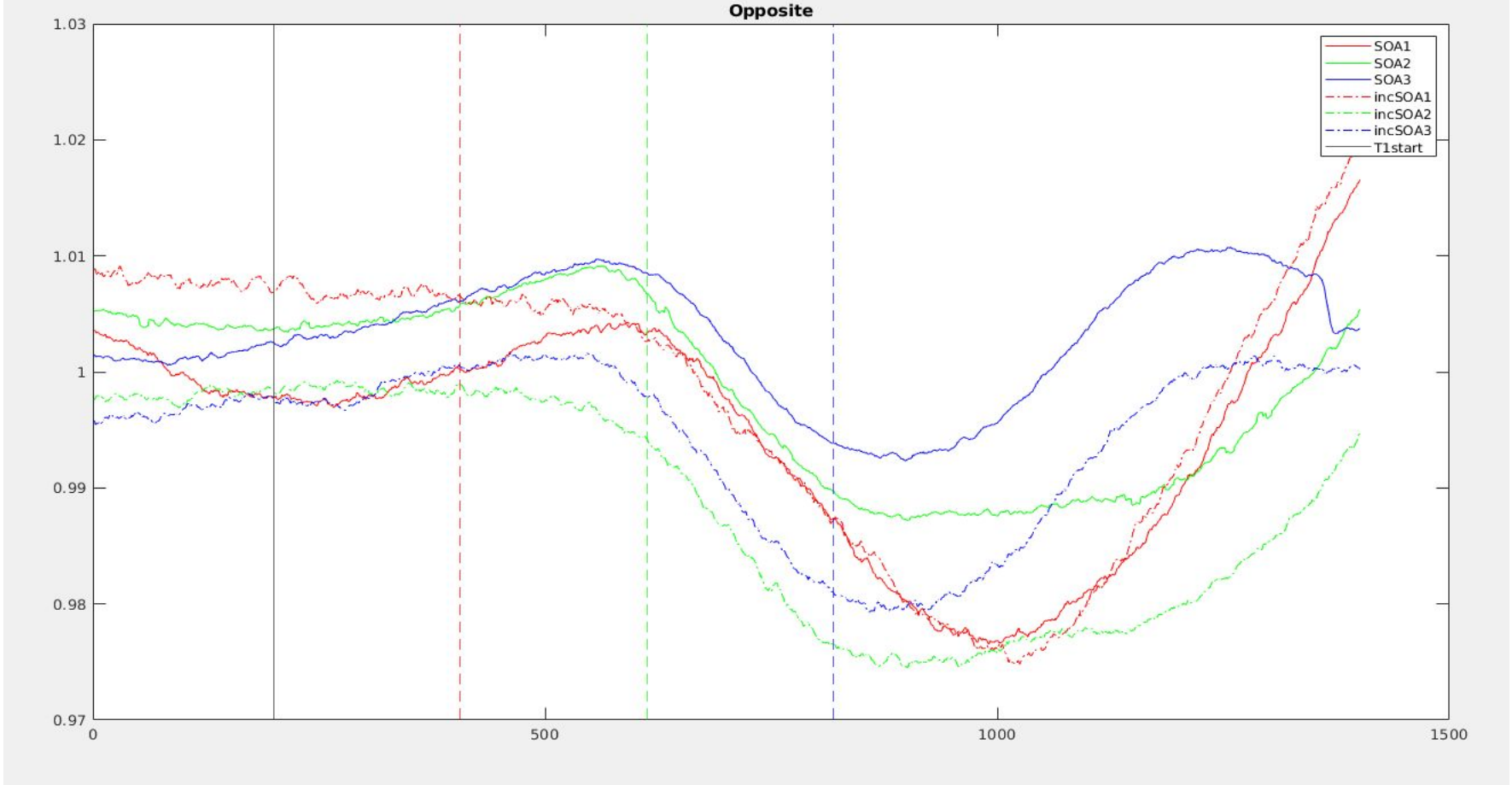

1.03
1.025
1.02
1.015
1.01
1.005
1
0.995
0.99
0
500
1000
1500

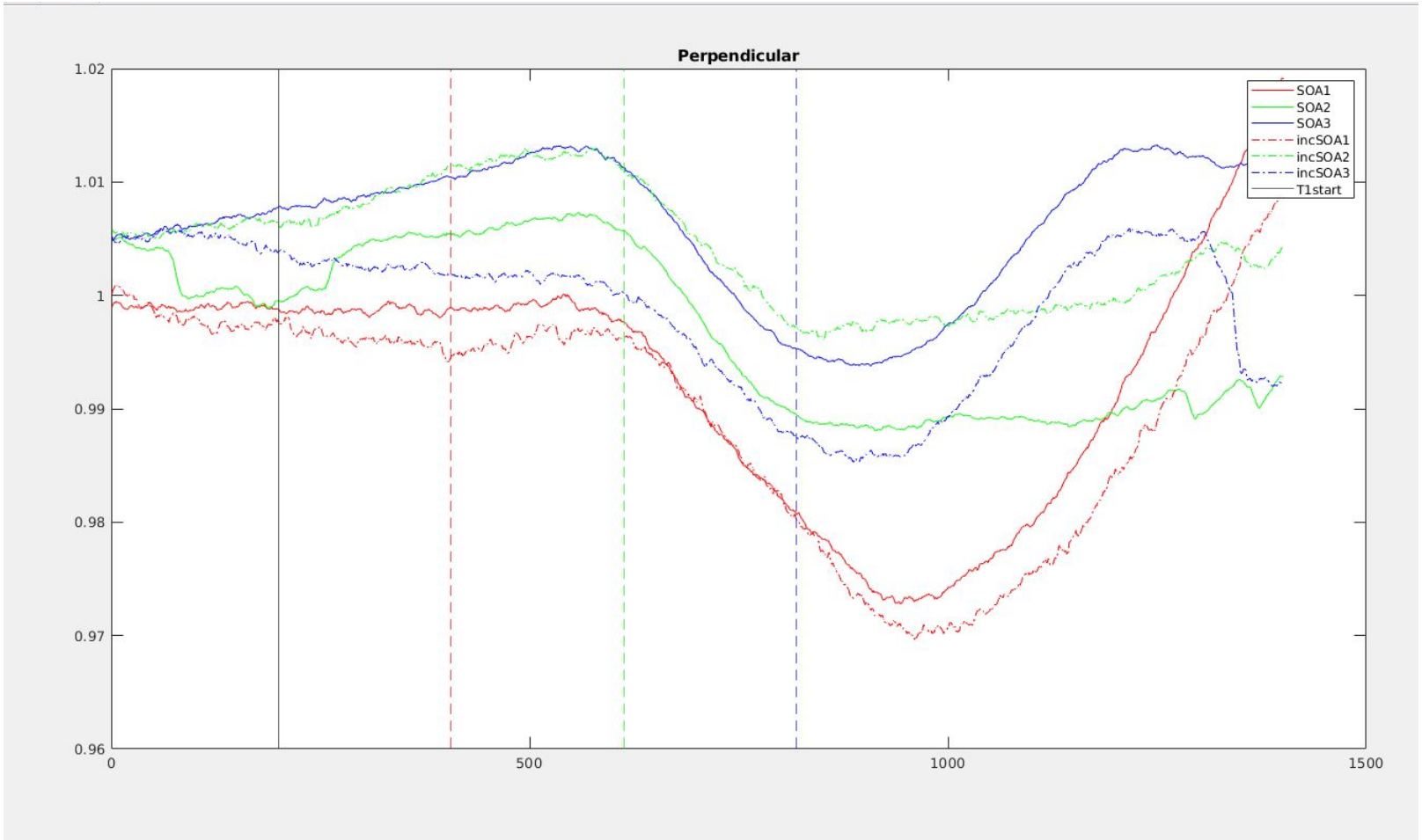
Next Task:

1. Draw vertical Lines in our graph to represent T1_start and T2_start. Since T2_start would different for SOA1, SOA2 and SOA3, so represent vertical lines according to the color of that SOA.
2. Chop the graphs till 1400th point
3. Check for results of perpendicular incorrect for SOA2
4. Subtract the normalized part instead of dividing
5. Take average of all SOA's and directions and then compare the graph with when their is only one target

NOTE : The error made in 46,47 and 48th slide includes both when the participant is not able to see the second target and when he guesses the direction of second target wrong.


Same
1.03
1.02
1.01
1
0.99
0.98
0.97
0
500
1000
1500
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
T1start


Opposite
1.03
1.02
1.01
1
0.99
0.98
0.97
0
500
1000
1500
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
T1start


Perpendicular
1.02
1.01
1
0.99
0.98
0.97
0.96
0
500
1000
1500
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
T1start

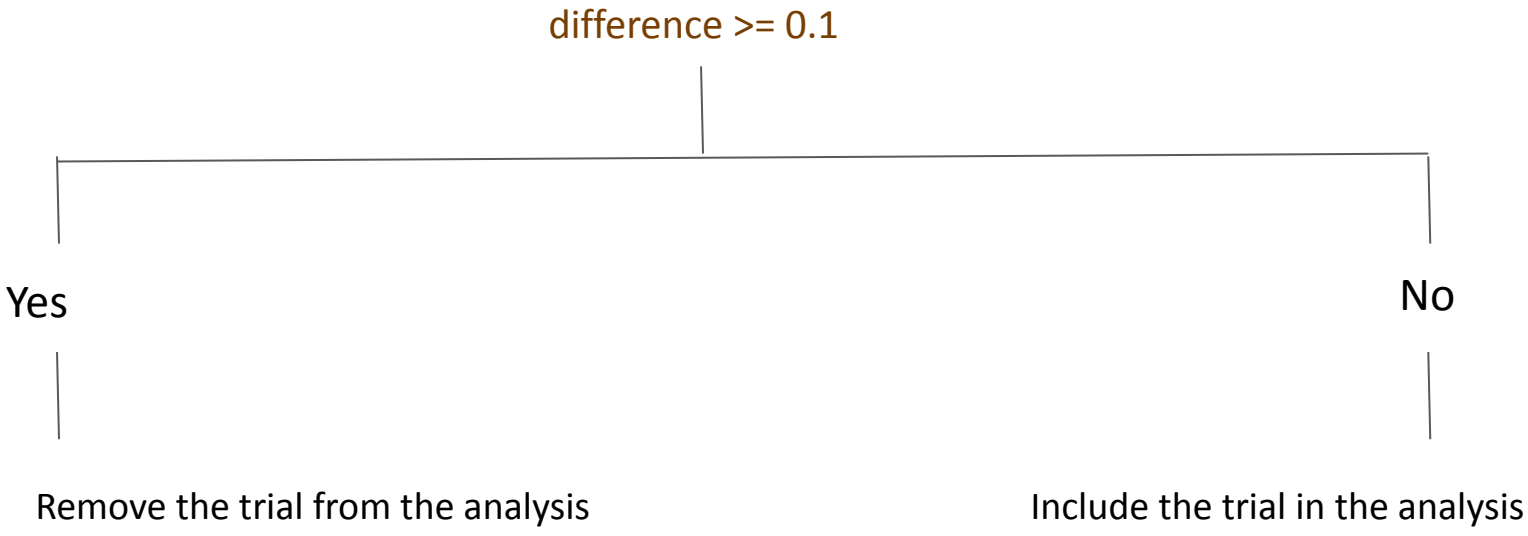
Next Steps:

1. Re- Plot 46, 47 & 48 graphs with incorrect trials only corresponding to incorrect second answer. e.g for same SOA1 condition plot incorrect trials when participant gave answer to second answer incorrectly as ‘NO’ despite second target was present. DO NOT include incorrect trials where participants have given incorrect answer for third question that is wrongly guessing the direction of second target.
2. Implement subtraction method for normalization as discussed in previous meetings.
3. Analysis on previously excluded participants (because of noisy signals) : Check if by excluding few noisy trials we can include previously excluded participants. The algorithm is mentioned in next slide.
4. If we can salvage some participants from step 3 then include them in the list shared in slide 35 and re-generate the plots in 46-48.
5. Exclude participants highlighted in red (in slide 35) from the analysis.

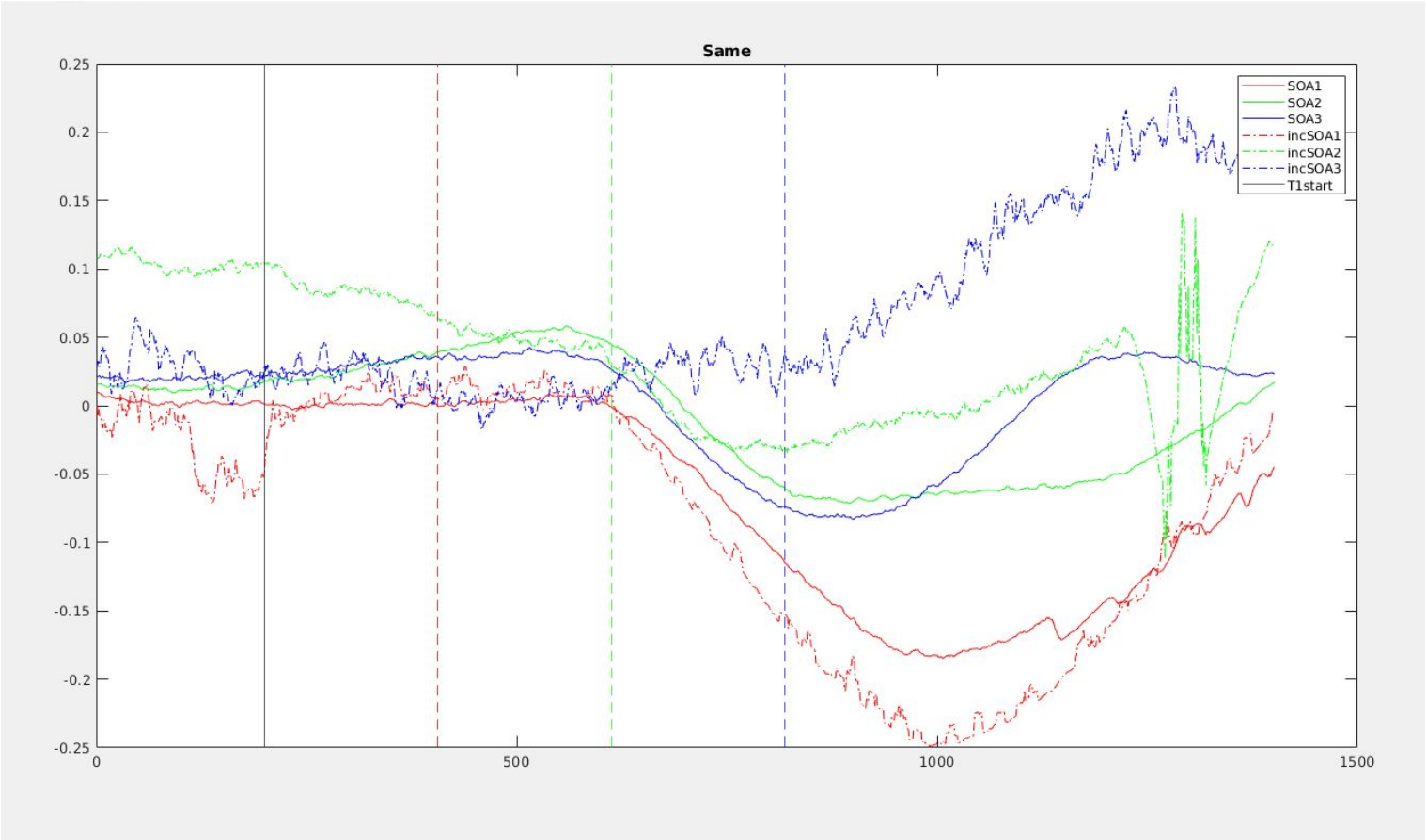
Trial-wise exclusion of noisy trials from previously excluded participants:

Assumption: there are only 1 or 2 trials which are noisy, so will remove only those trials.

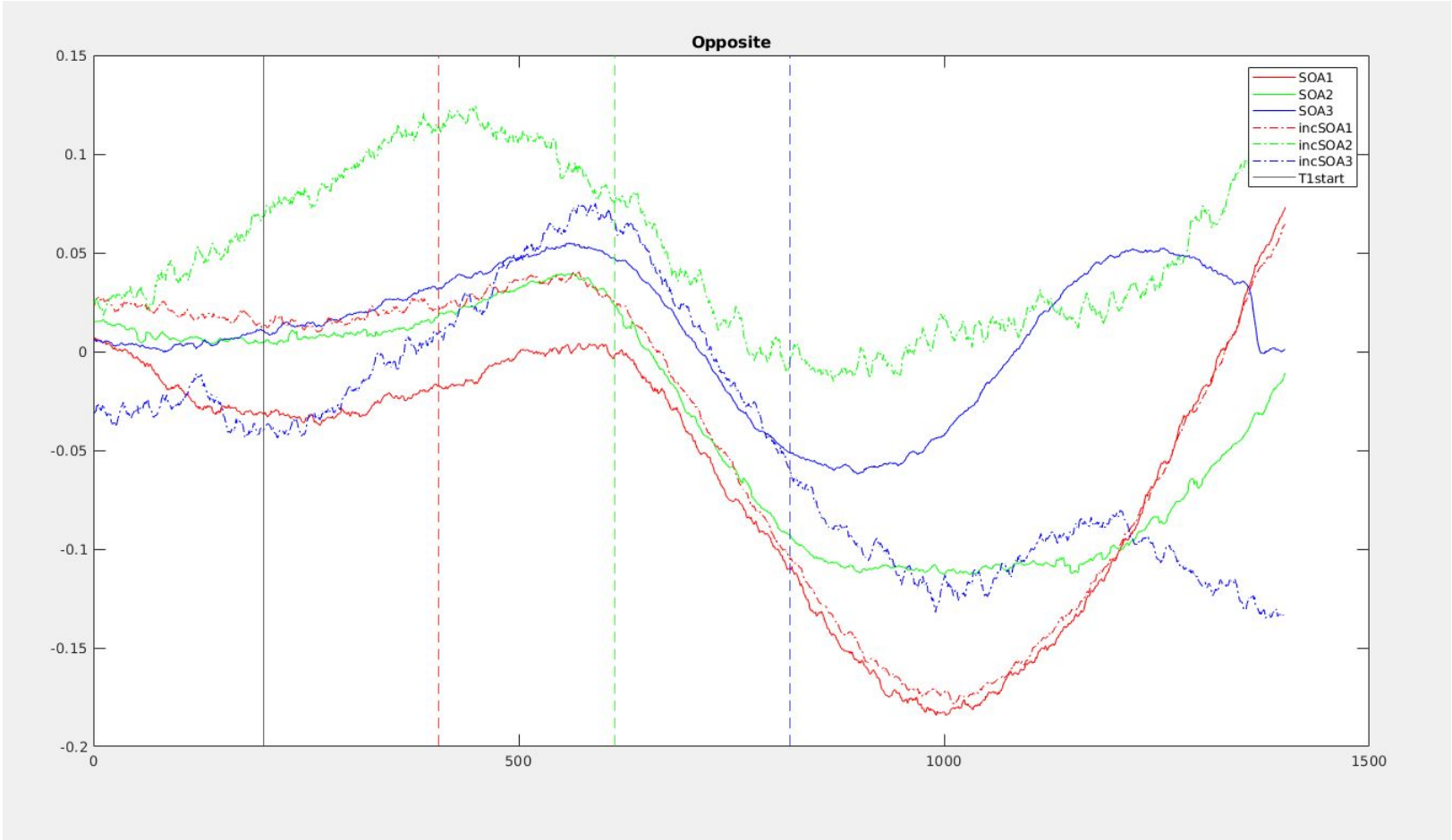
For each trial find difference between min and max value after normalization



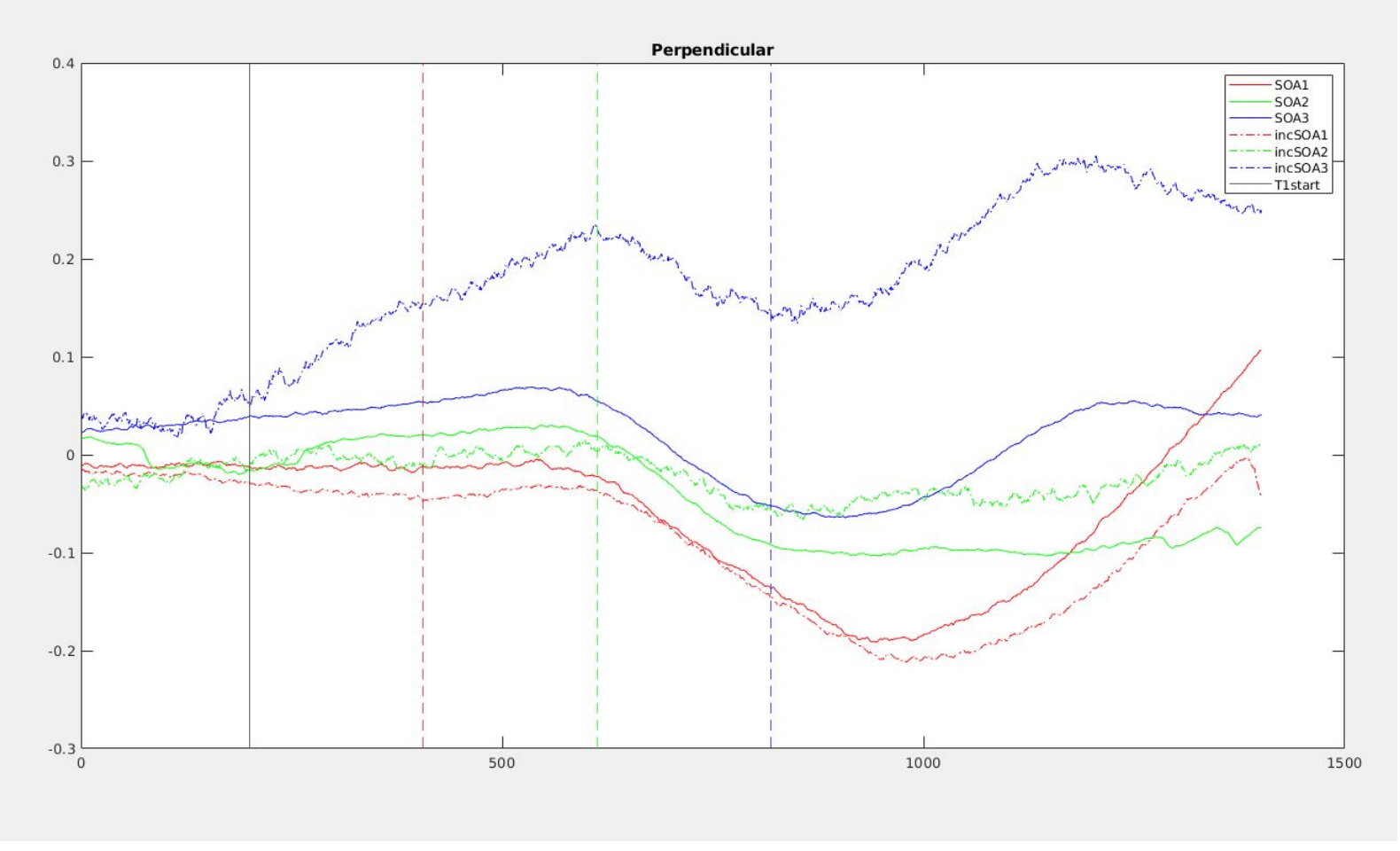
Baseline= Baseline-normalized_part
N = 12



Baseline= Baseline-normalized_part
N = 12



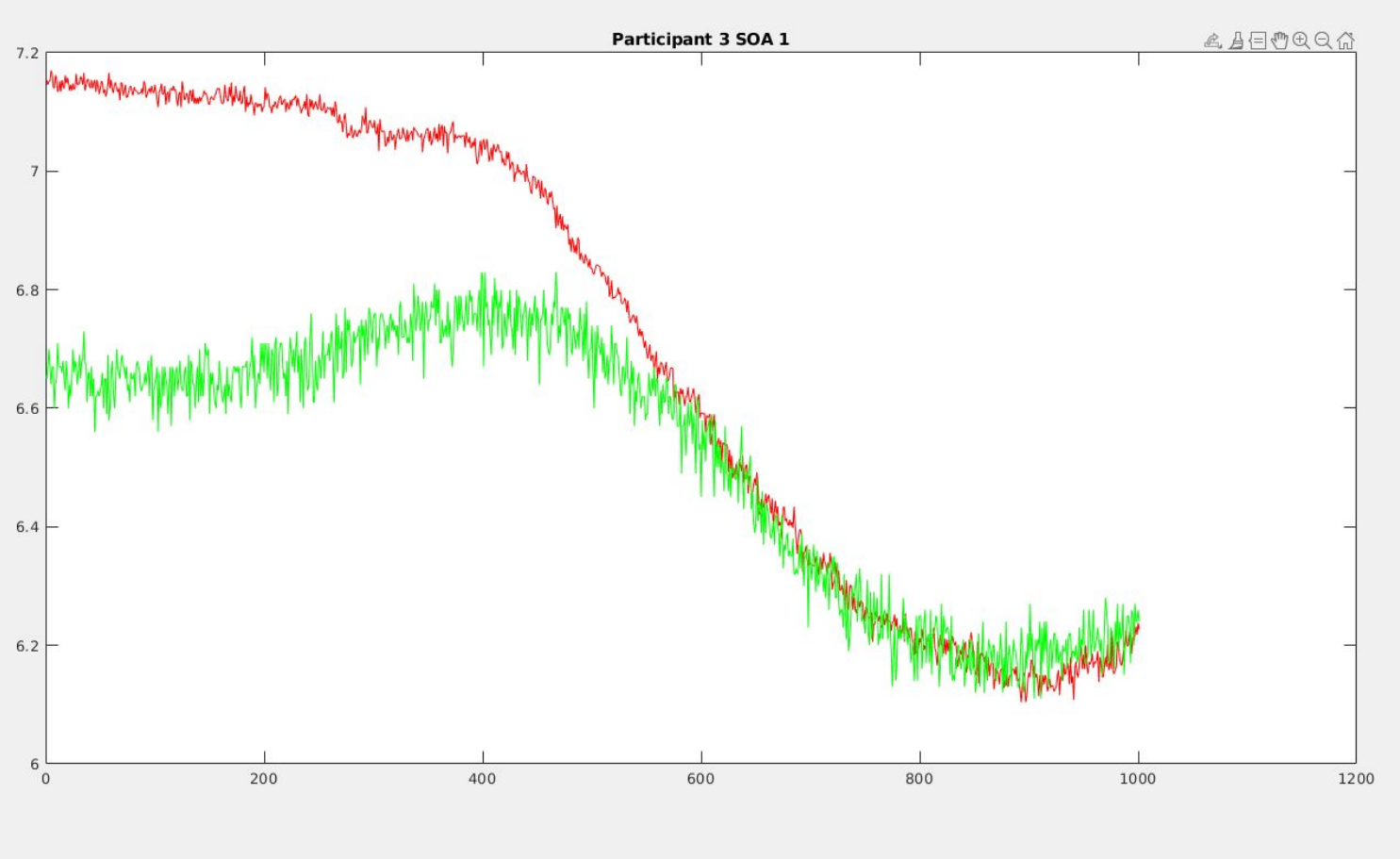
Baseline= Baseline-normalized_part
N = 12

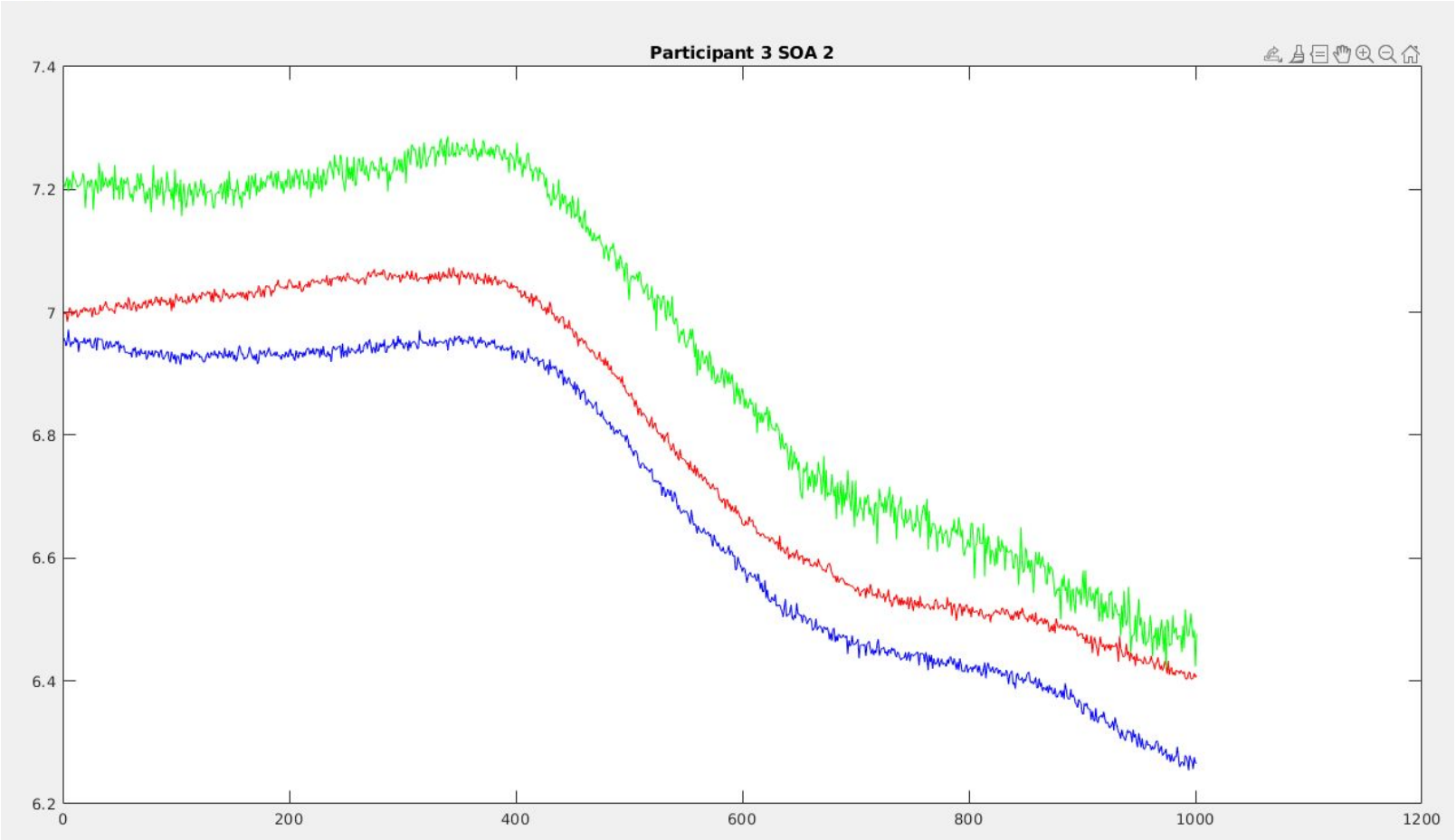


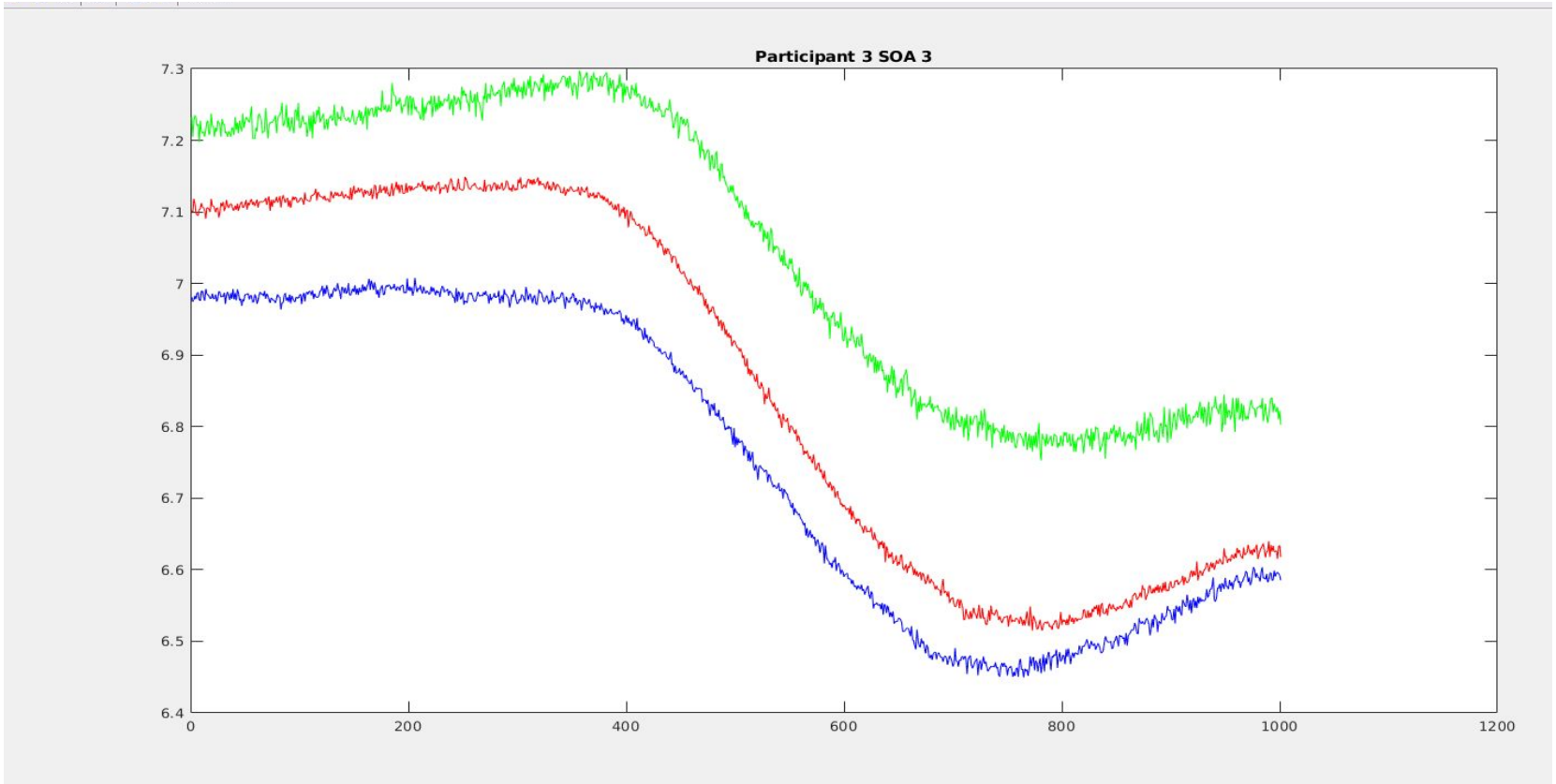
After doing the process written in 50th slide : (Participant_Number , No of DataEvents having max and min <0.1) :

8 → 134
9→ 24
10→ 229
14 → 138
17 → 117
20 → 213
21 → 181

Ma’am my personal opinion is that this won’t be a correct method to check for noisy trials because when I saw this method I thought why not check it for all participants and the number of trials for participant 3 that were following this criteria was 33 only but the participant graph is not noisy at all (Next slide) this is my previous graphs that were used by me to check if they can be included or not

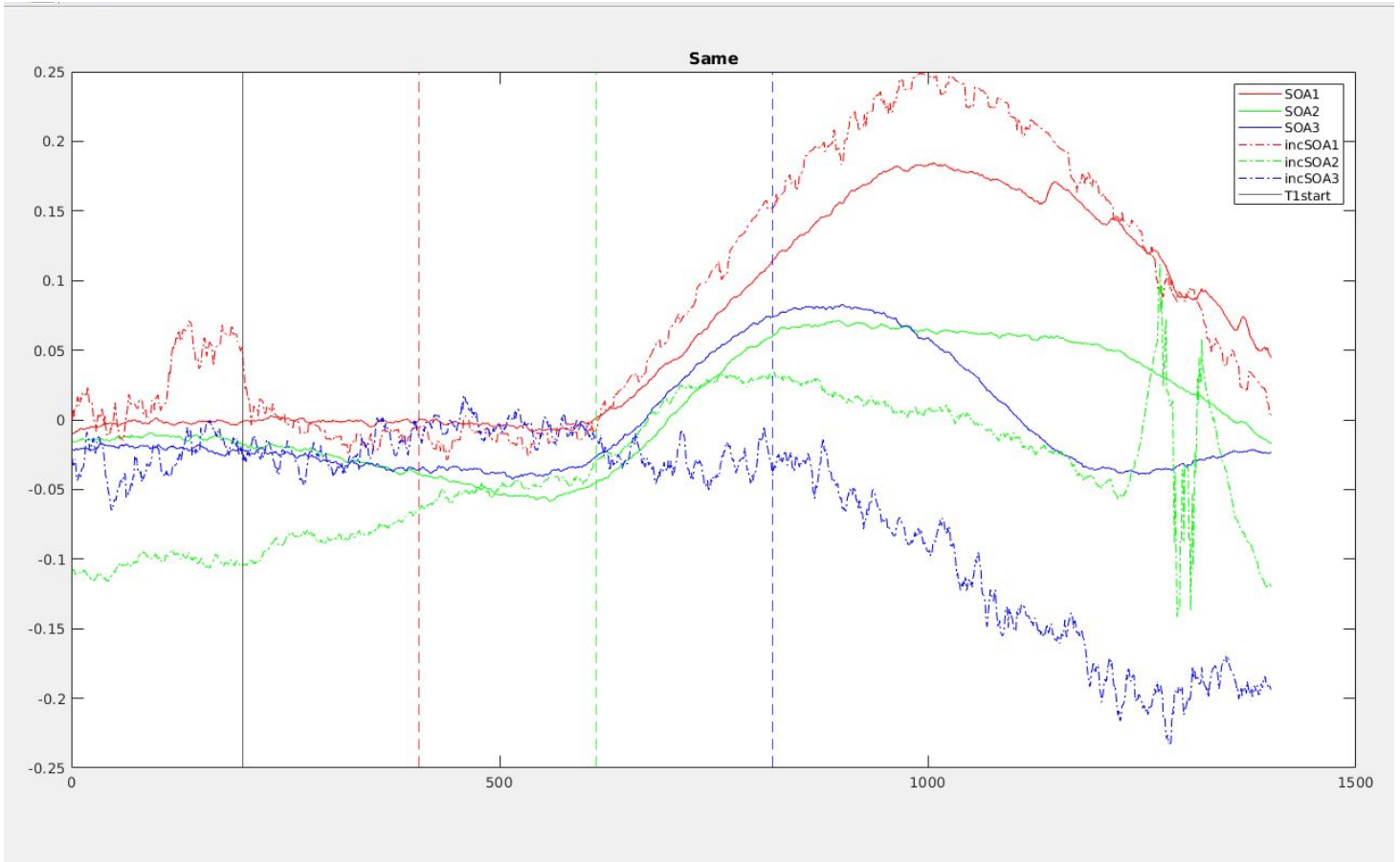

Participant 3 SOA 1
7.2
7
6.8
6.6
6.4
6.2
6
0
200
400
600
800
1000
1200


Participant 3 SOA 2
7.4
7.2
7
6.8
6.6
6.4
6.2
0
200
400
600
800
1000
1200

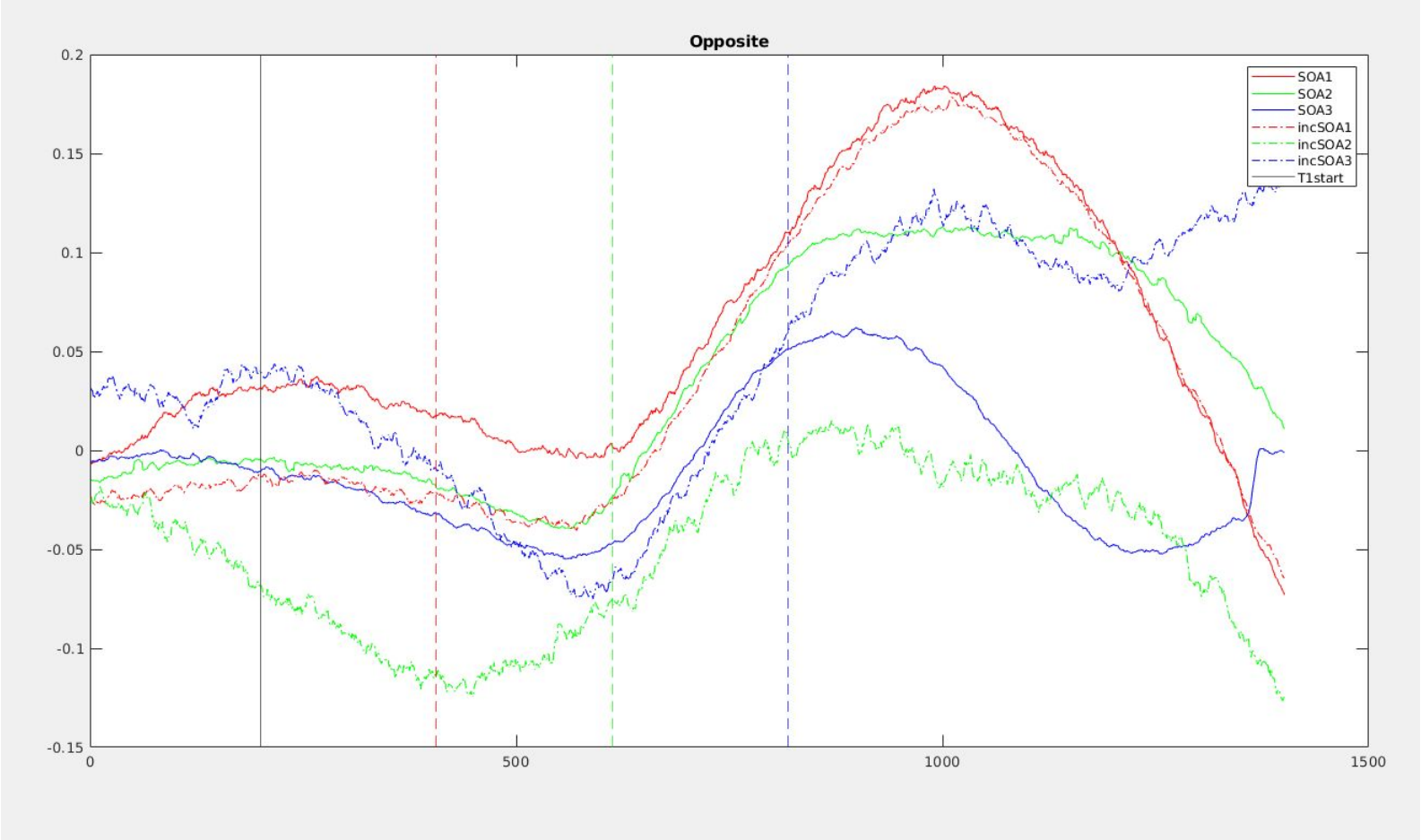

Participant 3 SOA 3
7.3
7.2
7.1
7
6.9
6.8
6.7
6.6
6.5
6.4
0
200
400
600
800
1000
1200

	Same	Opposite	Perpendicular
SOA1	C >I	C == I	C ==I
SOA 2	I >C	I >C	I>C
SOA3	I >C	I <C	I >C

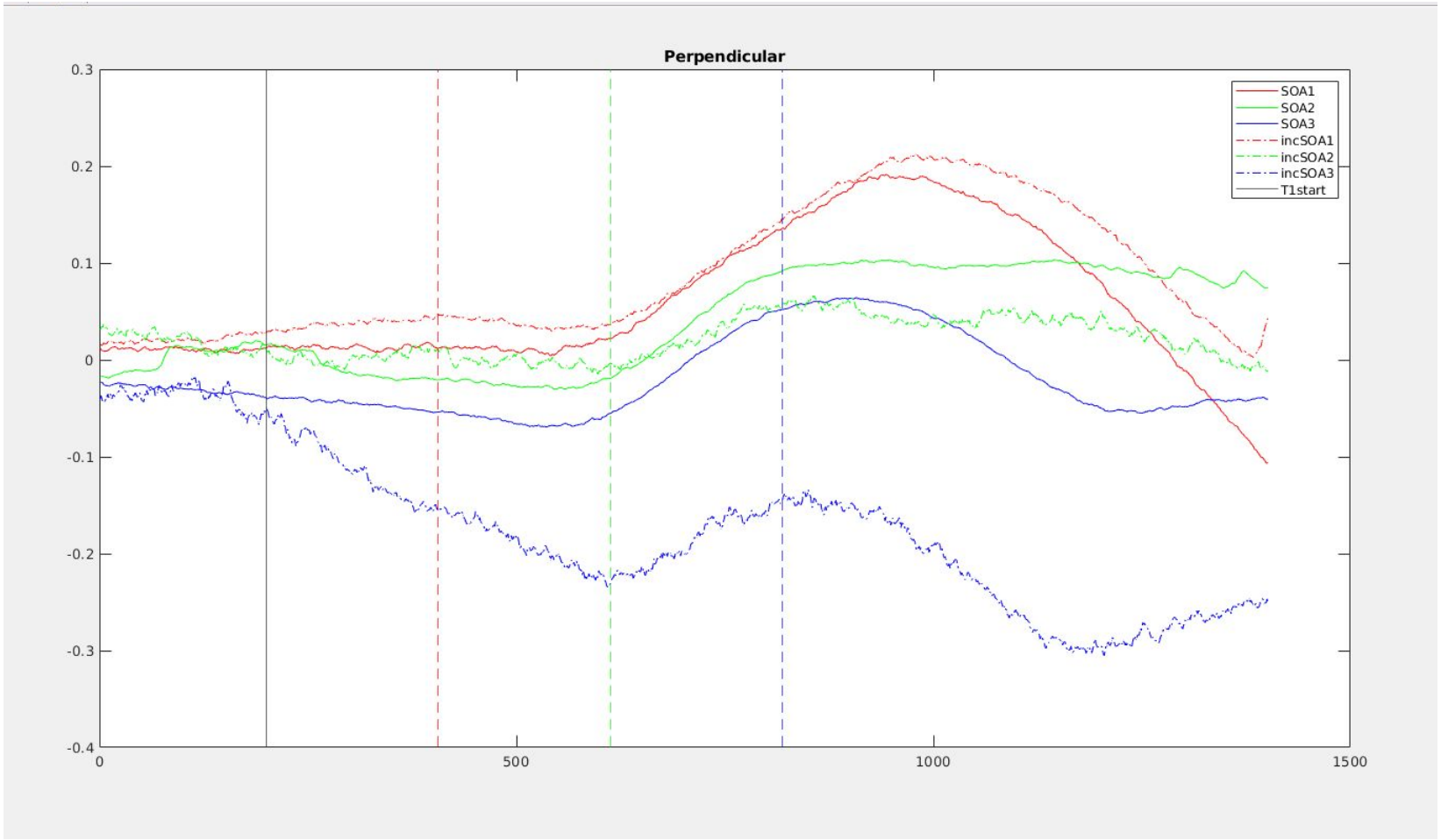
Baseline= normalized_part-Baseline
N = 12



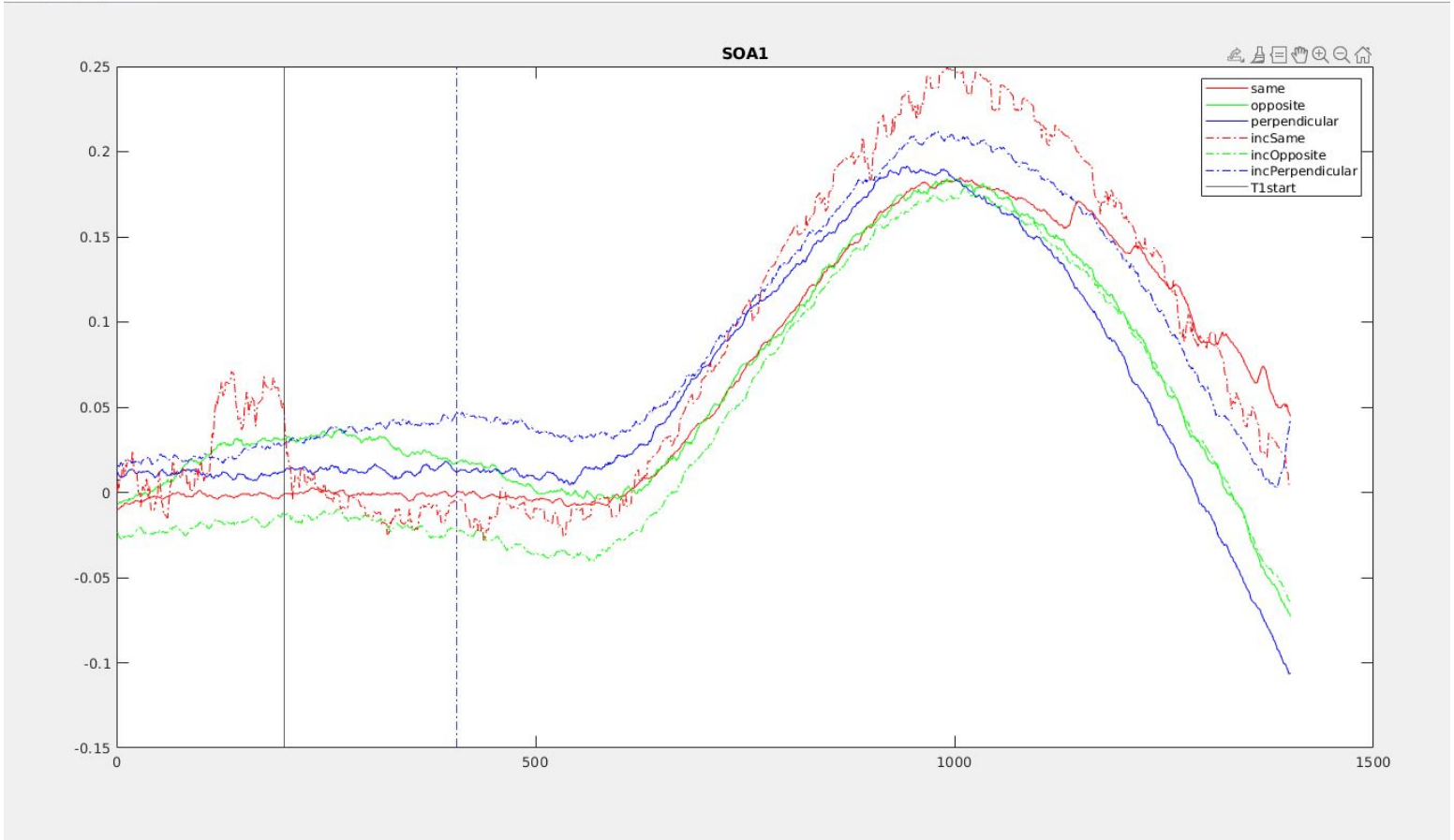
Baseline= normalized_part-Baseline
N = 12



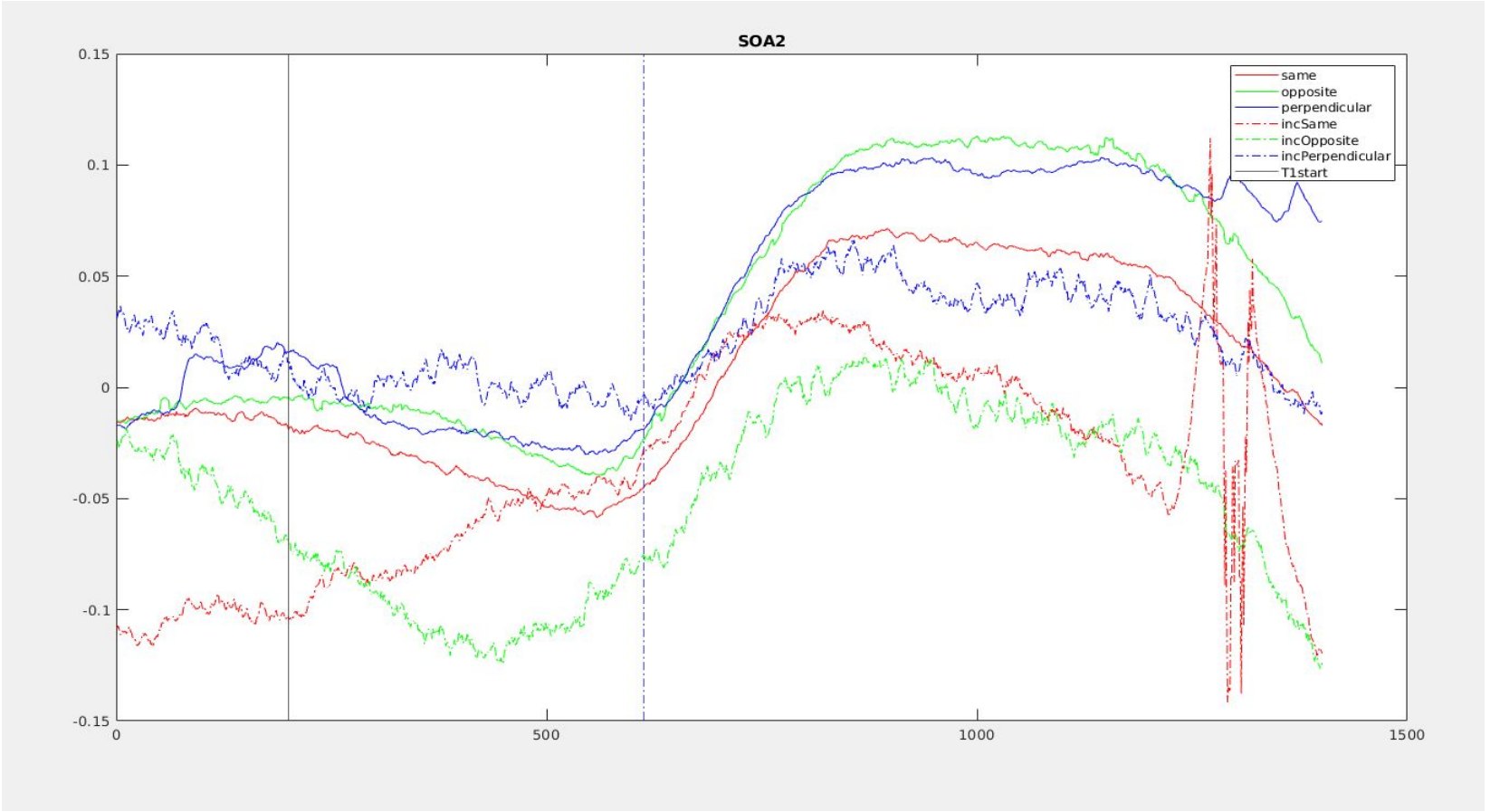
Baseline= normalized_part-Baseline
N = 12



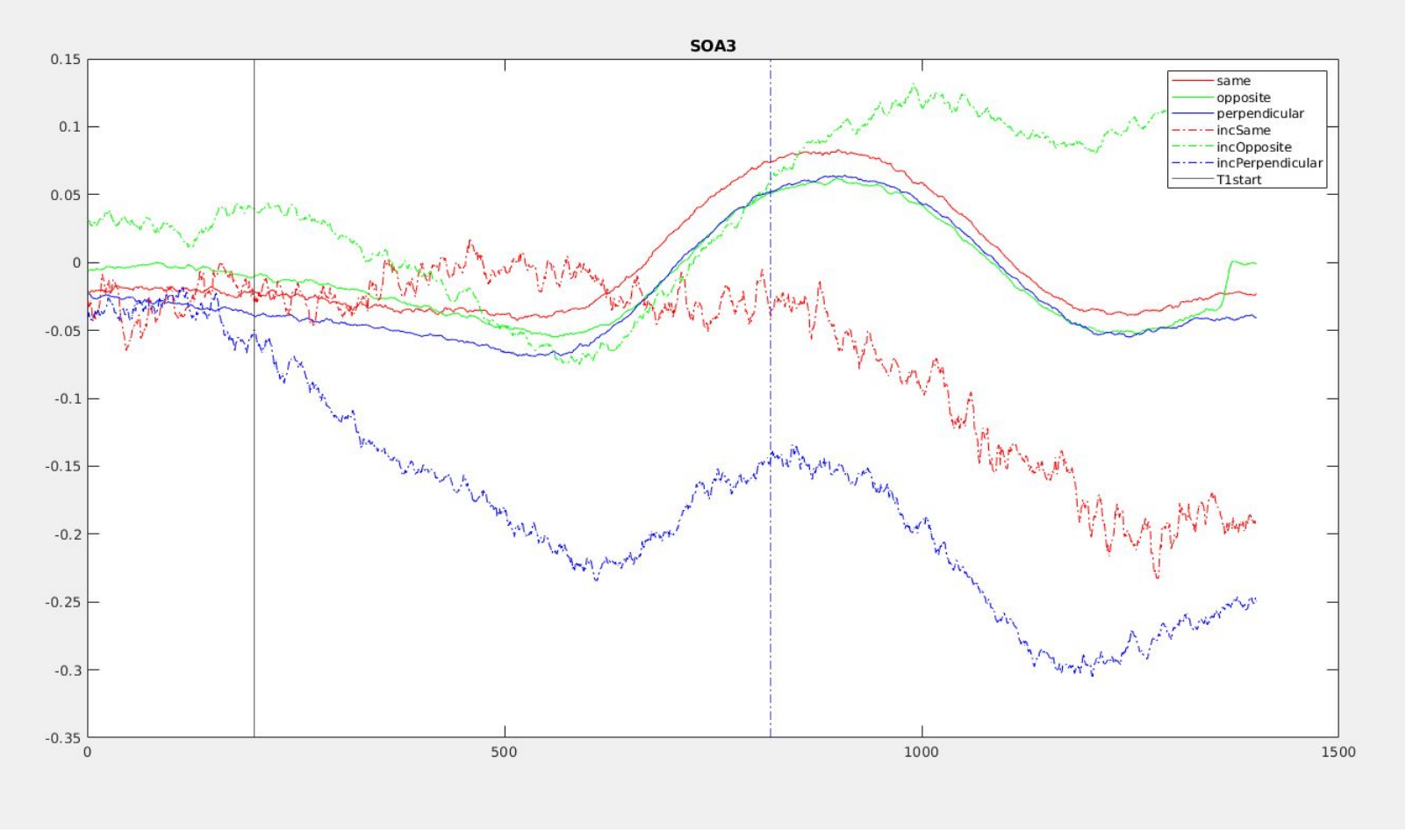
Baseline= normalized_part-Baseline
N = 12



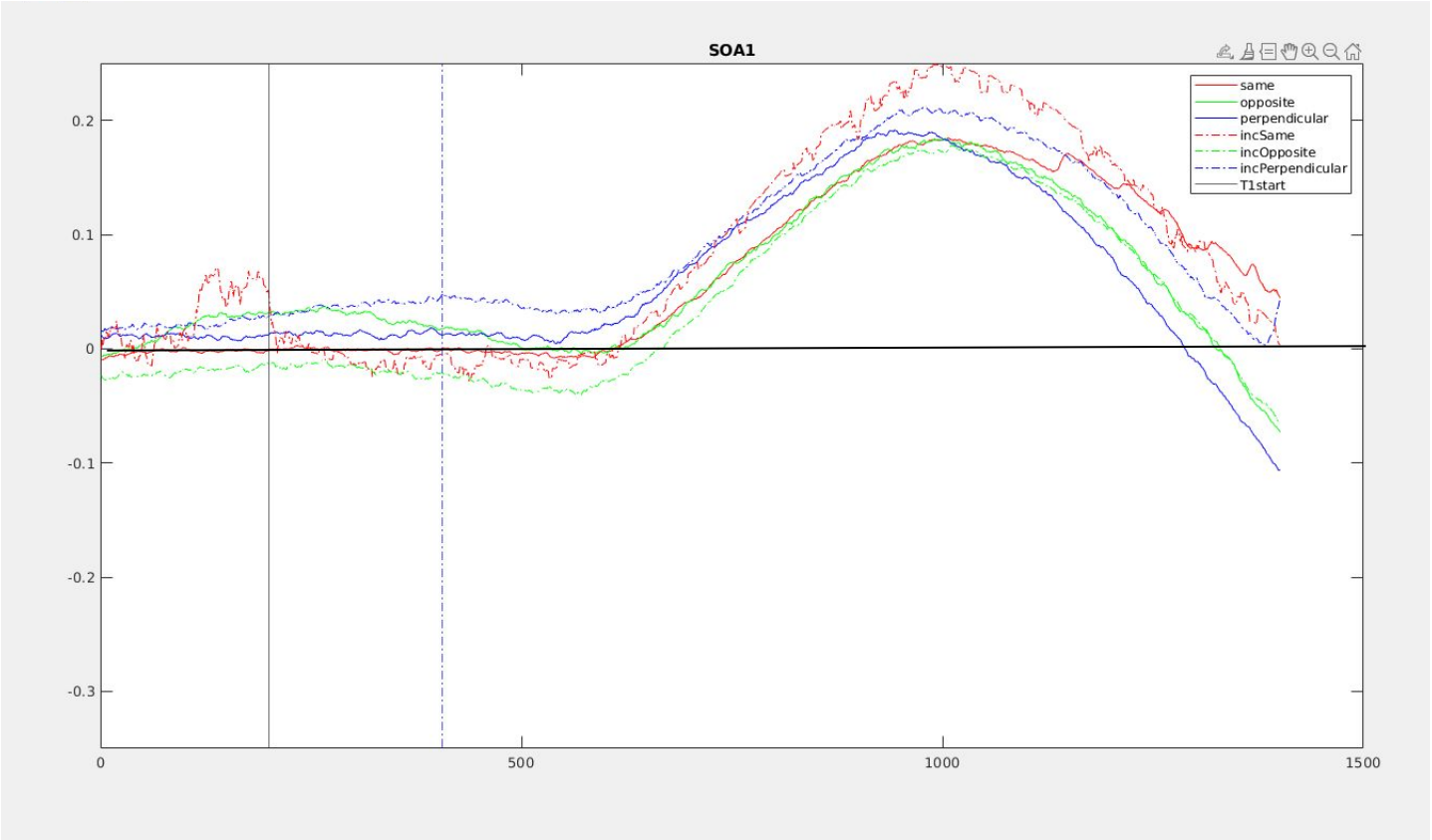
Baseline= normalized_part-Baseline
N = 12

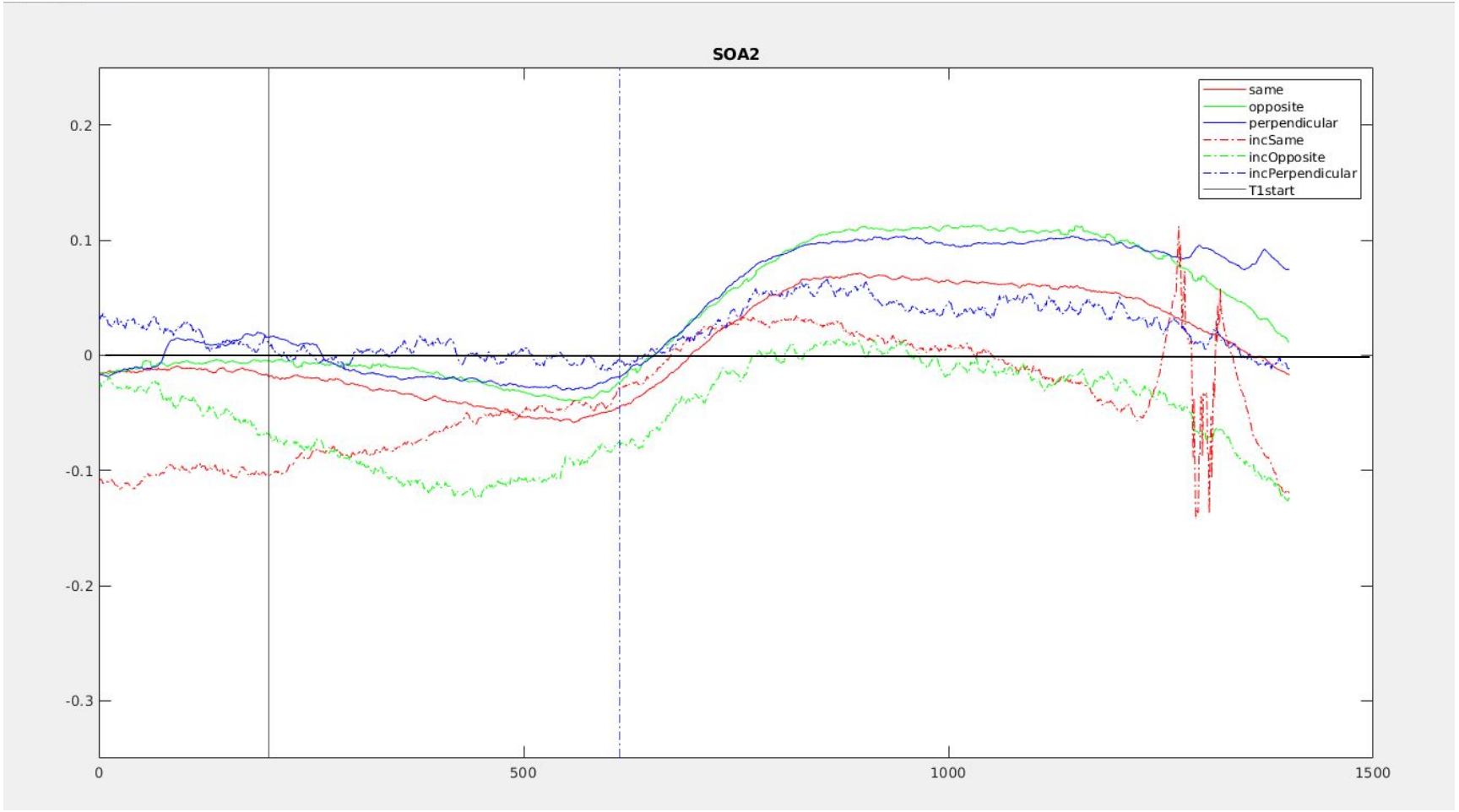


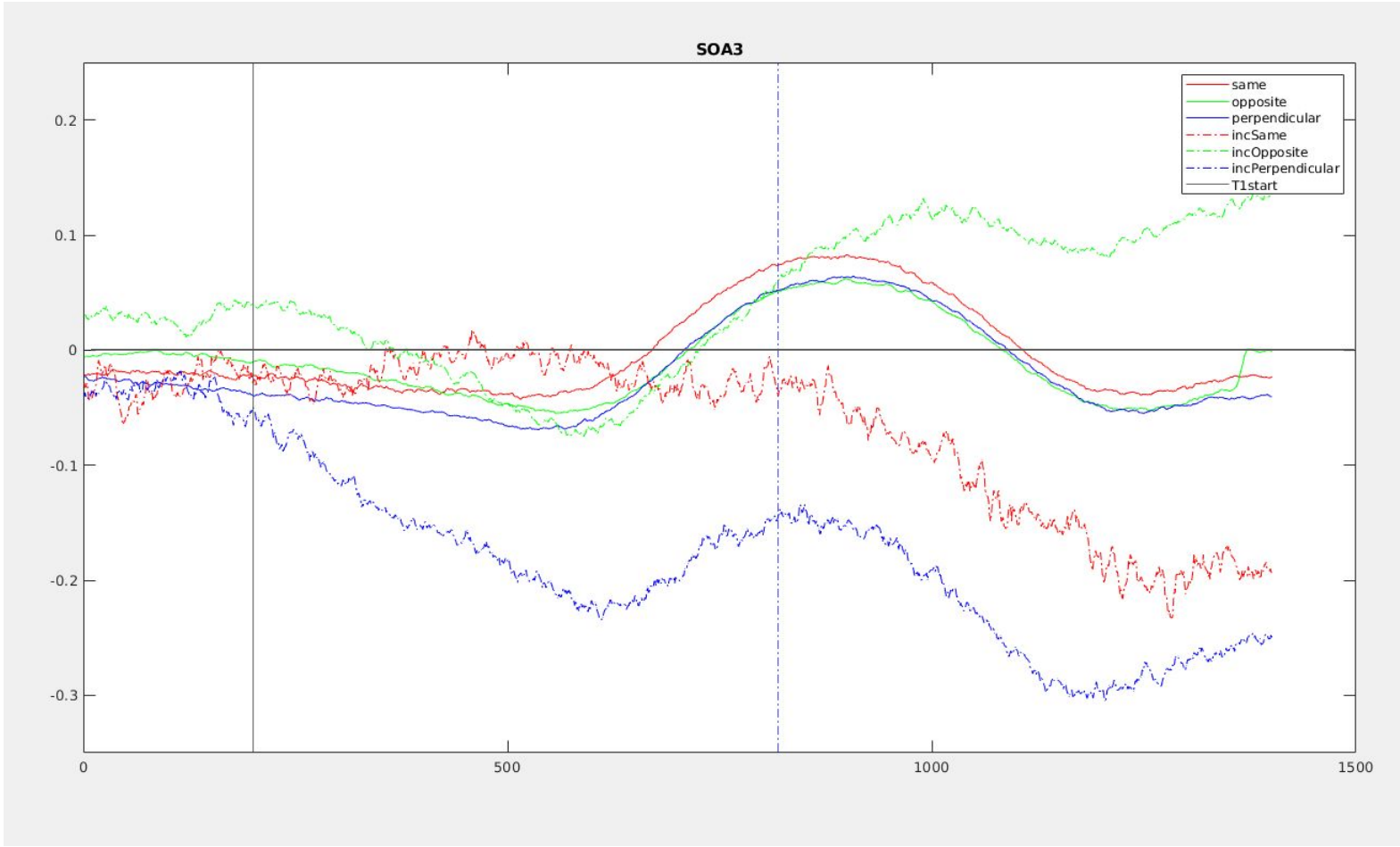
Baseline= normalized_part-Baseline
N = 12



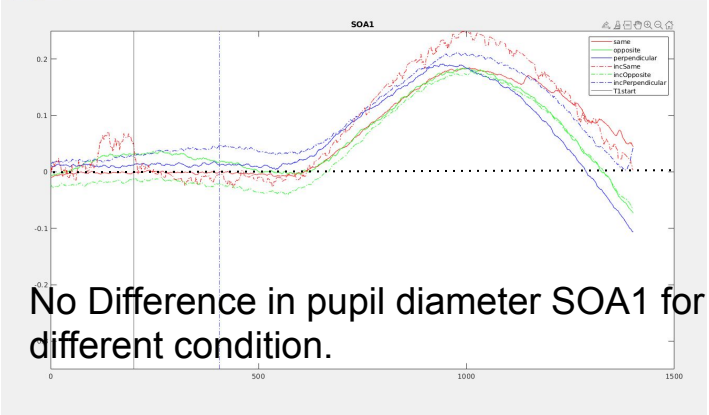
Changed limits of 62,63,64 to [-0.35 0.25]




SOA2
same
opposite
perpendicular
incSame
incOpposite
incPerpendicular
T1start
0.2
0.1
0
-0.1
-0.2
-0.3
0
500
1000
1500

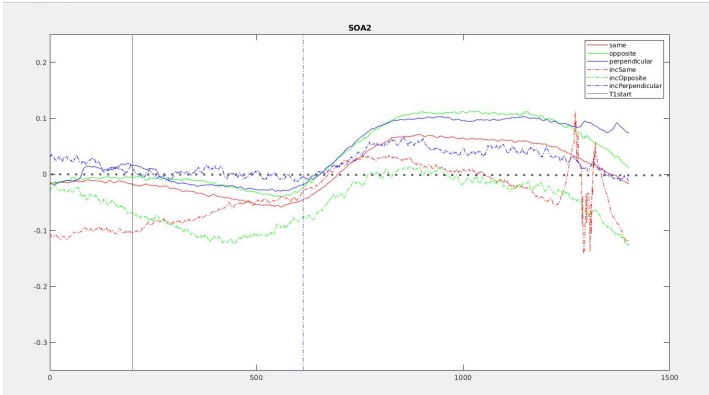

SOA3
same
opposite
perpendicular
incSame
incOpposite
incPerpendicular
T1start
0.2
0.1
0
-0.1
-0.2
-0.3
0
500
1000
1500

SOA1

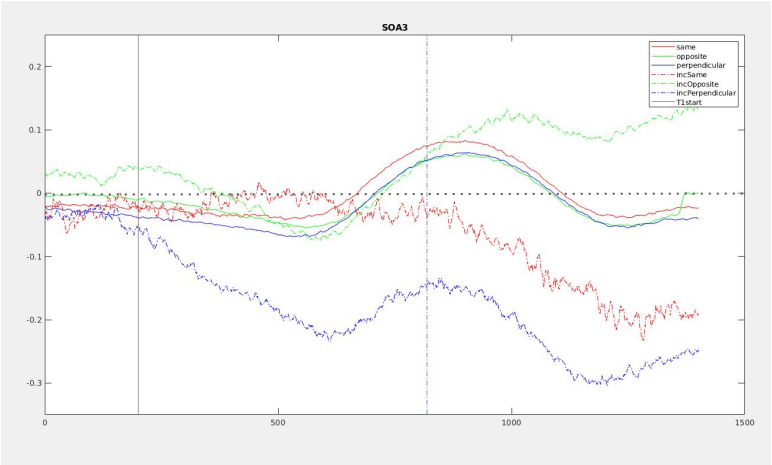


No Difference in pupil diameter SOA1 for different condition.

SOA2

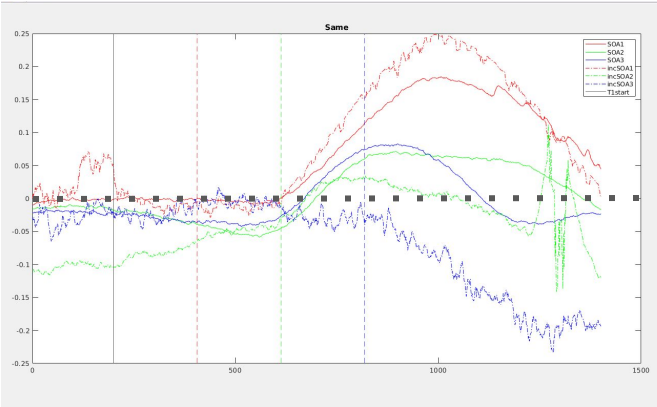


SOA3

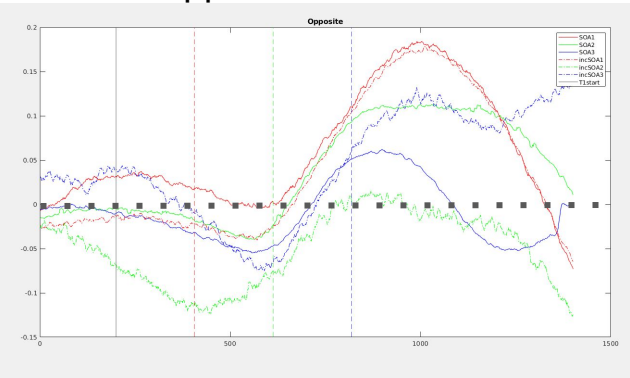


Change in pupil diameter:
SOA1 > SOA2 > SOA3

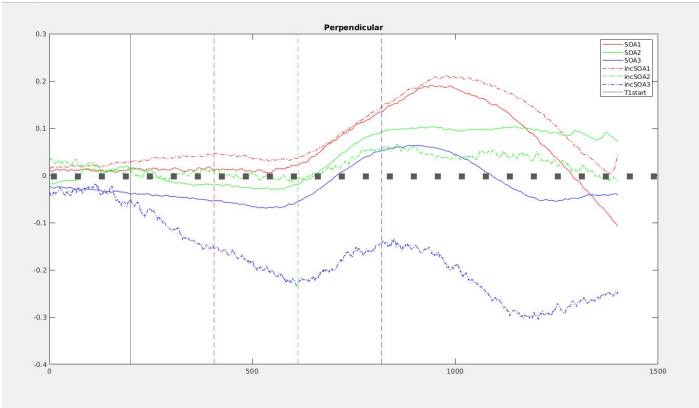
Same



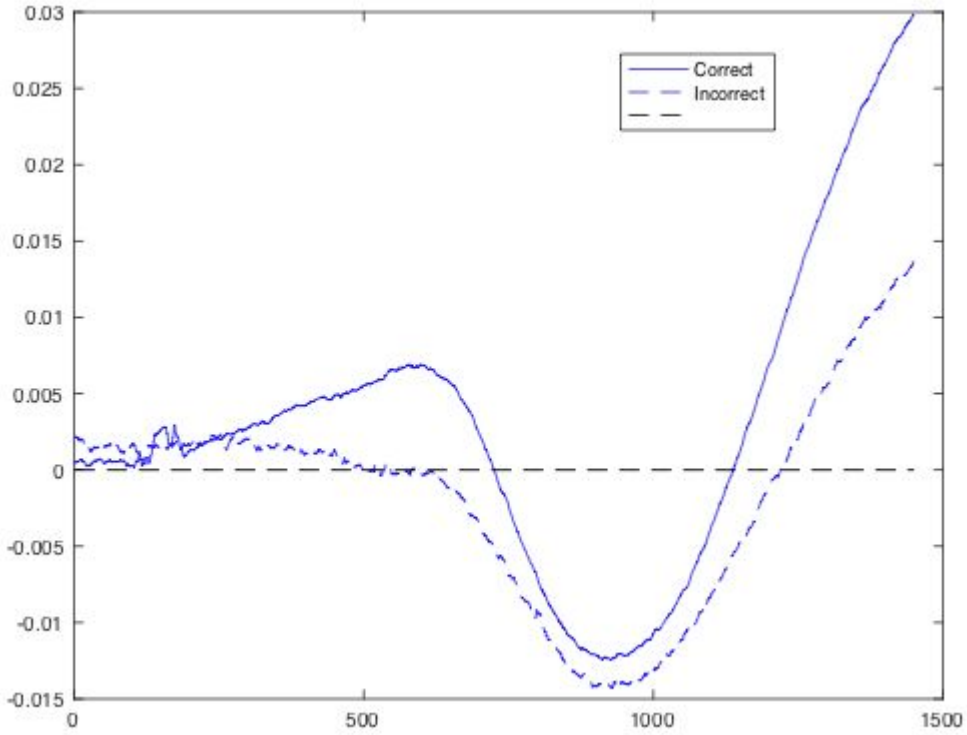
Opposite

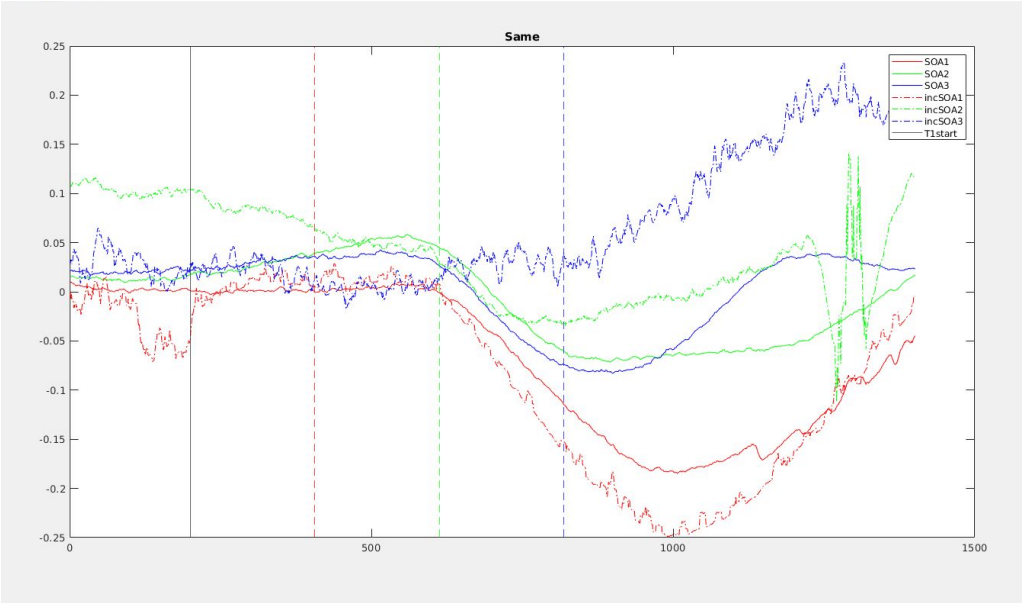


Perpendicular

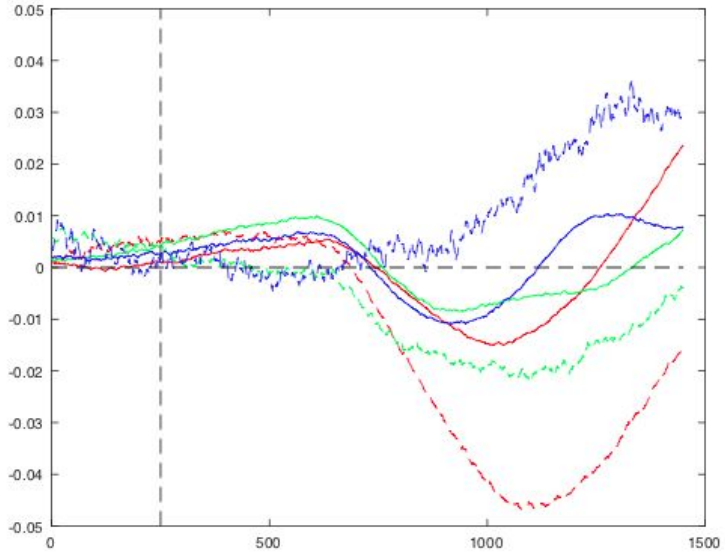


No T2


Correct
Incorrect
0.03
0.025
0.02
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
0
500
1000
1500


Same
SOA1
SOA2
SOA3
incSOA1
incSOA2
incSOA3
T1start
0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
-0.25
0
500
1000
1500

Same


0.05
0.04
0.03
0.02
0.01
0
-0.01
-0.02
-0.03
-0.04
-0.05
0
500
1000
1500

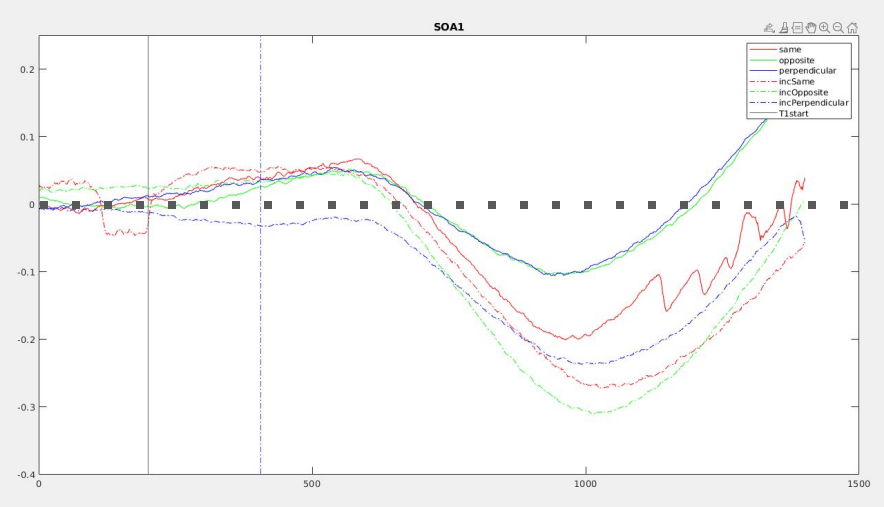
Next Step

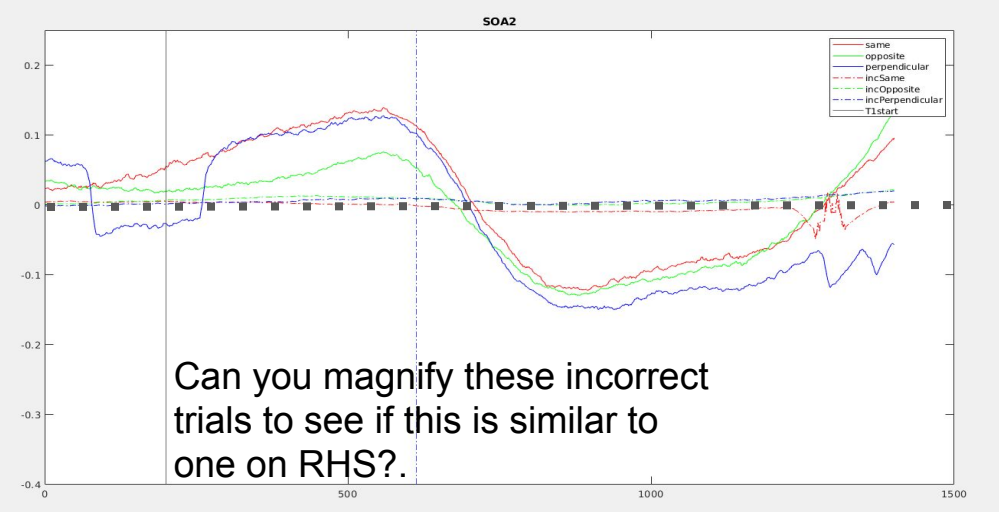
Change normalization method and period

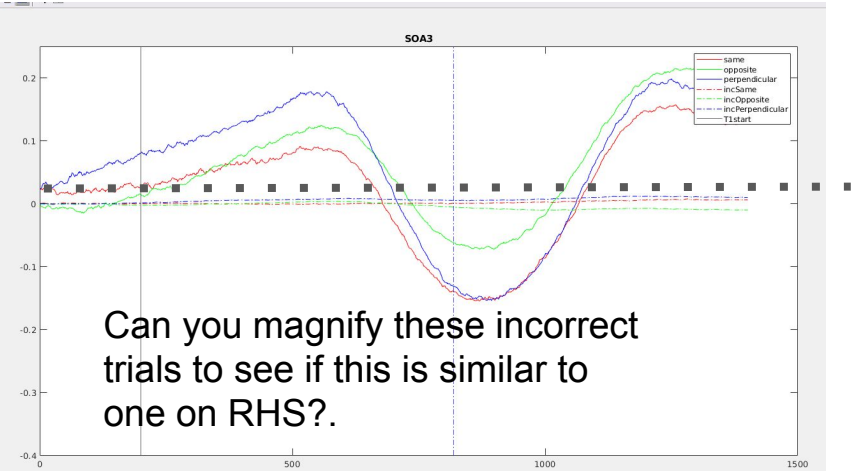
• Old: Signal - mean (First 250 points)
• New: Signal - mean (From first point until T1Onset)
mean (From first point until T1Onset)

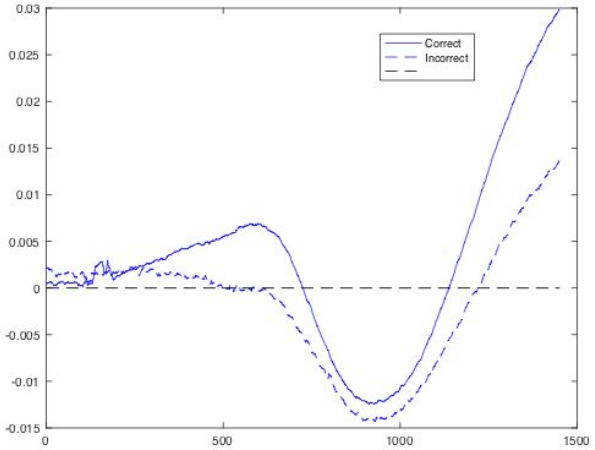
Mean method

• Old: mean of trials (participant) à mean across participants
• New: Append all trials (after normalization) for a condition across participant and then make the mean
e.g collect all correct trials for the Same condition SOA1 from all participant in one array and then take the average.
• Plot both condition as well as SOA wise like in slide #69 & 70 of shared google Presentation.
• Evaluate standard deviation of each condition and plot as an area across each curve as shown in next slide std

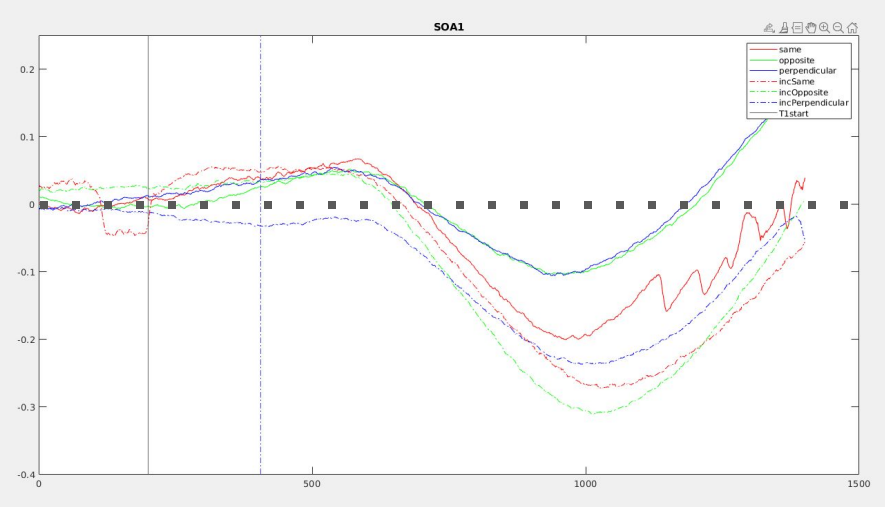


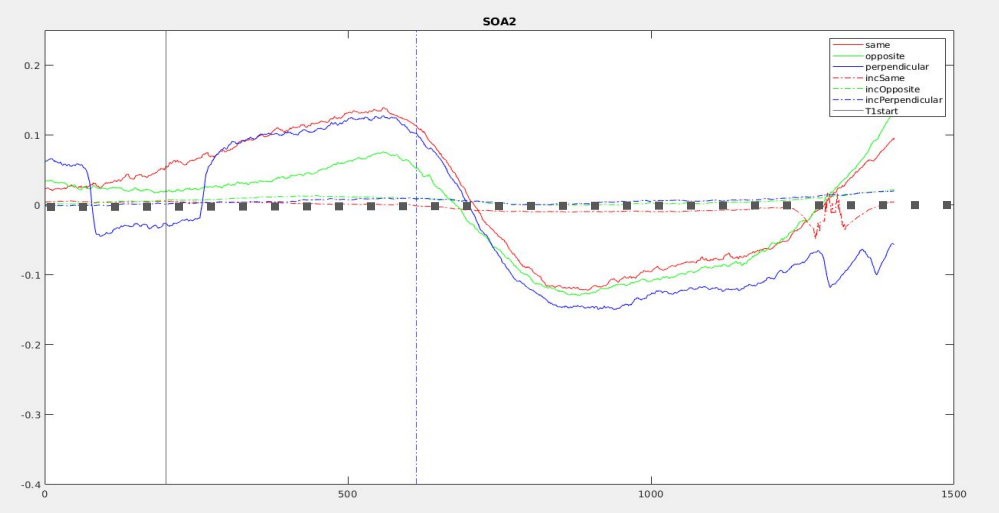


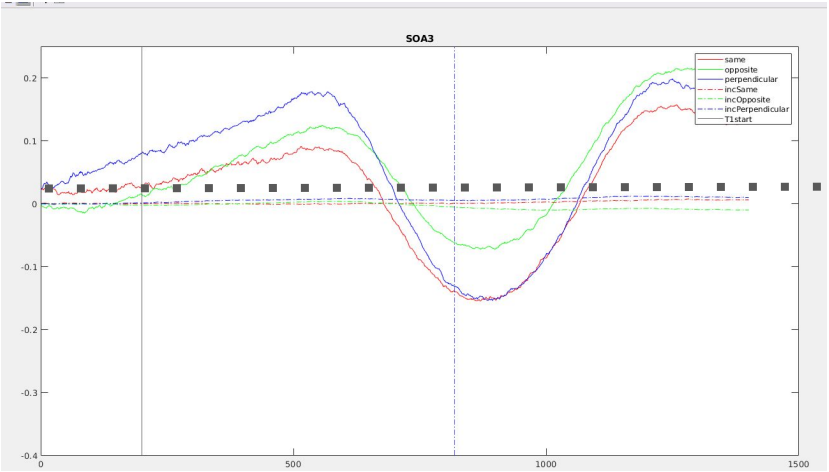


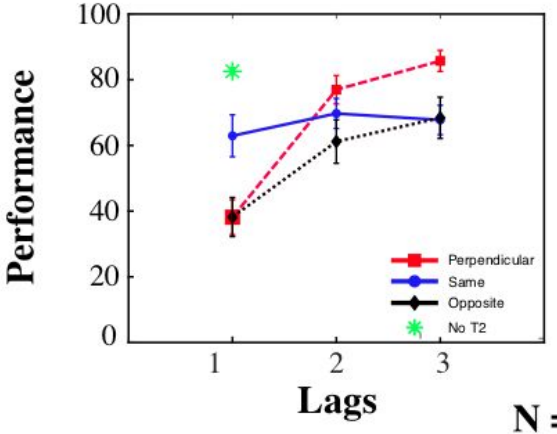


Baseline= Signal - mean (From first point unti





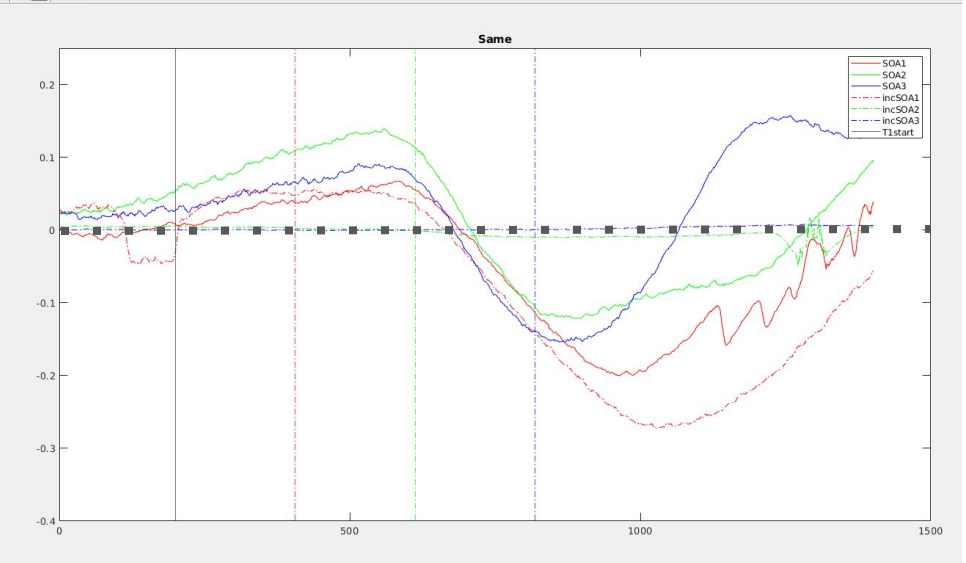


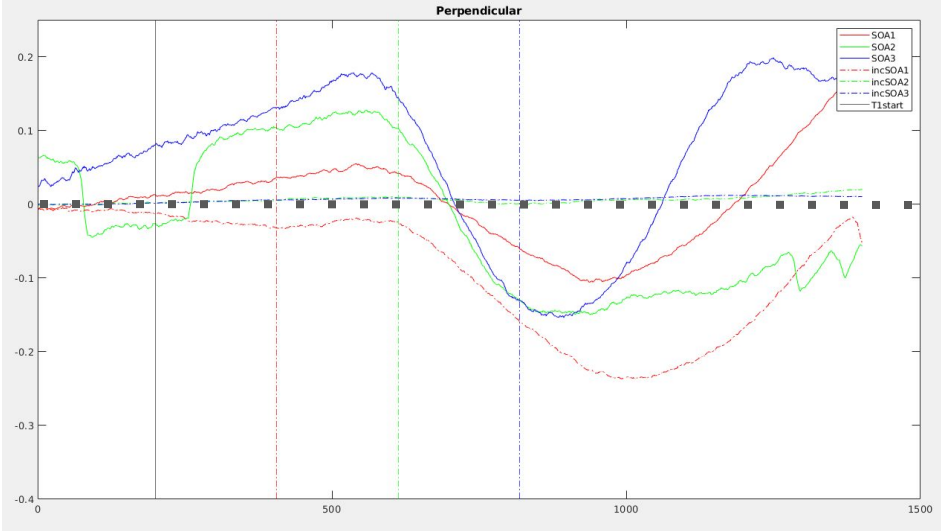


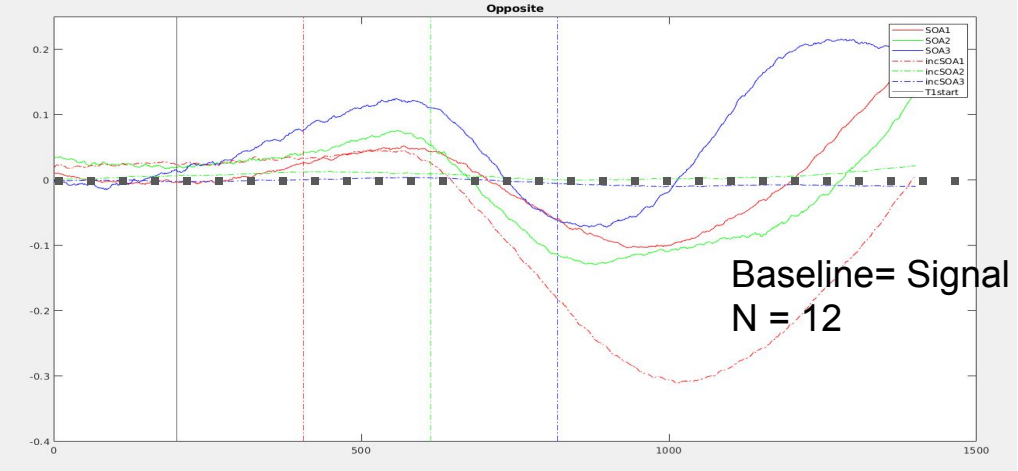
N = 19

Baseline= Signal - mean (From first point until

N = 12







Baseline= Signal - mean (From first point until

N = 12

	Same	Opposite	Perpendicular
SOA1	C >I	C >I	C > I
SOA 2	C<I	C<I	C<I
SOA3	C<I	C<I	C<I

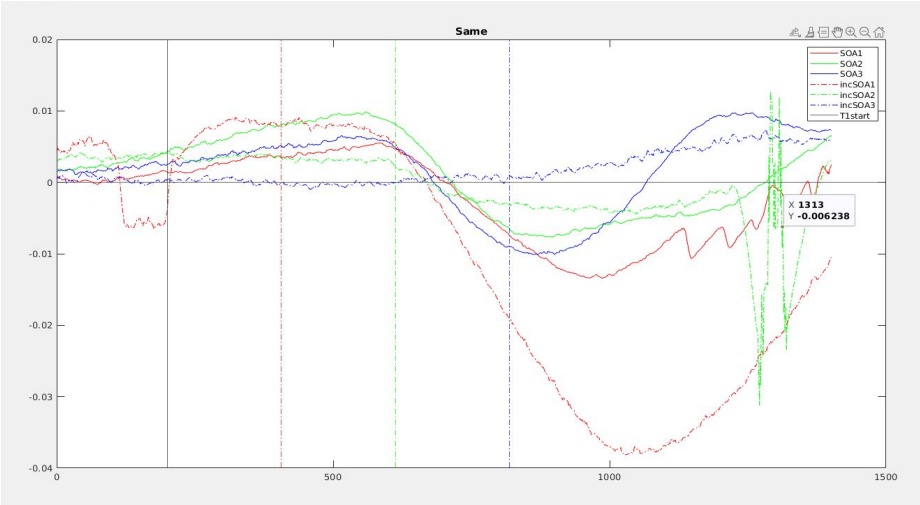
- SOA 2 & 3 the pupil size is close to zero… have to see if this magnitude is same as NoT2 correct condition?
- SOA 1 Incorrect > Correct
- Latency difference as a function of SOA.
-

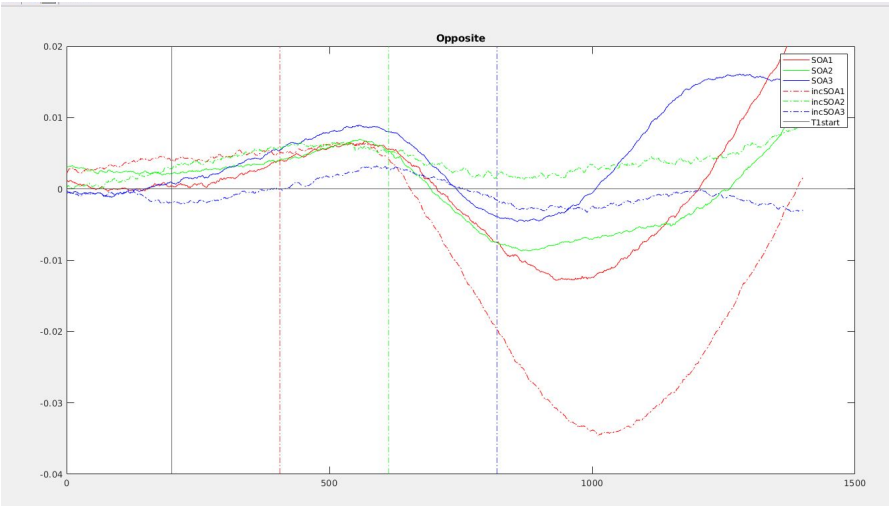
Observations

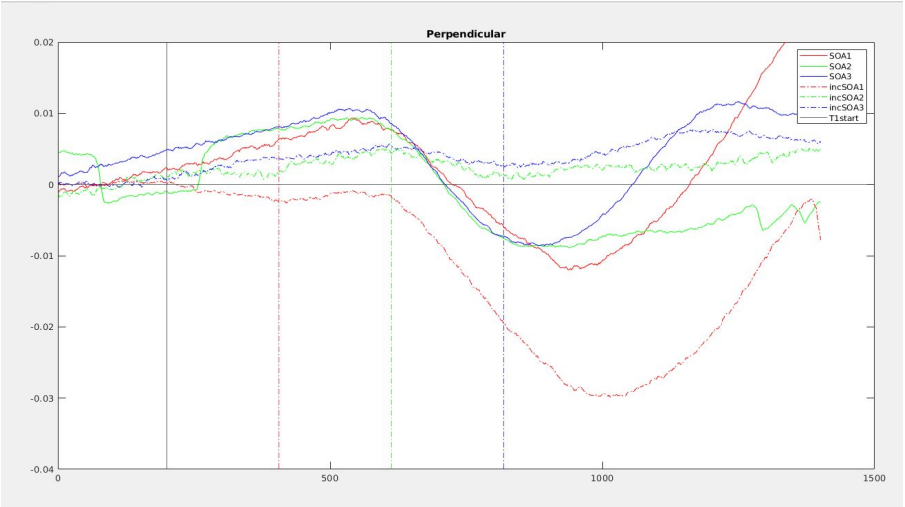
- Difference between correct and incorrect trials @SOA1
- This difference disappears for SOA2 & 3 close to zero.
- Latency different for different SOA for all directions: same, opposite and perpendicular
- @SOA1 max constriction for same followed by opposite and perpendicular

Based on #79 & 80:

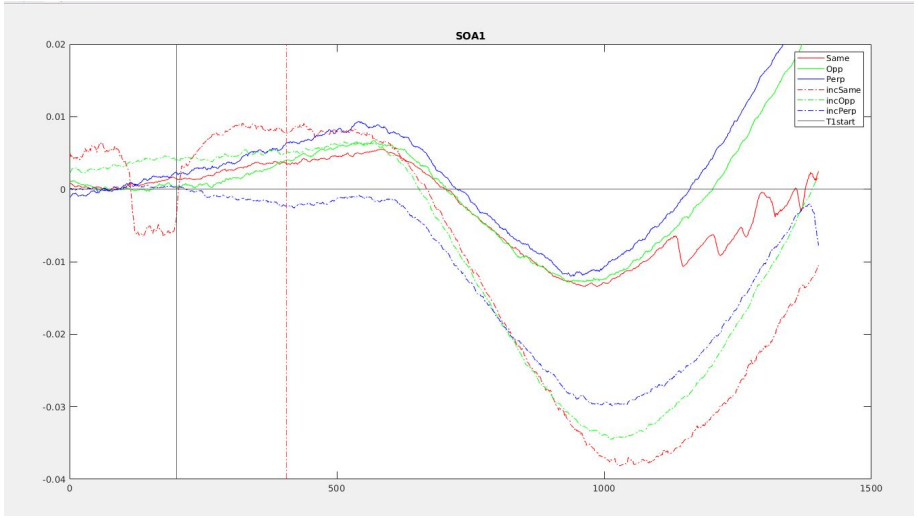
1. Pupil dia SOA1 < SOA2 == SOA3 == >
2. @SOA1 Correct dia > Incorrect diameter
3.

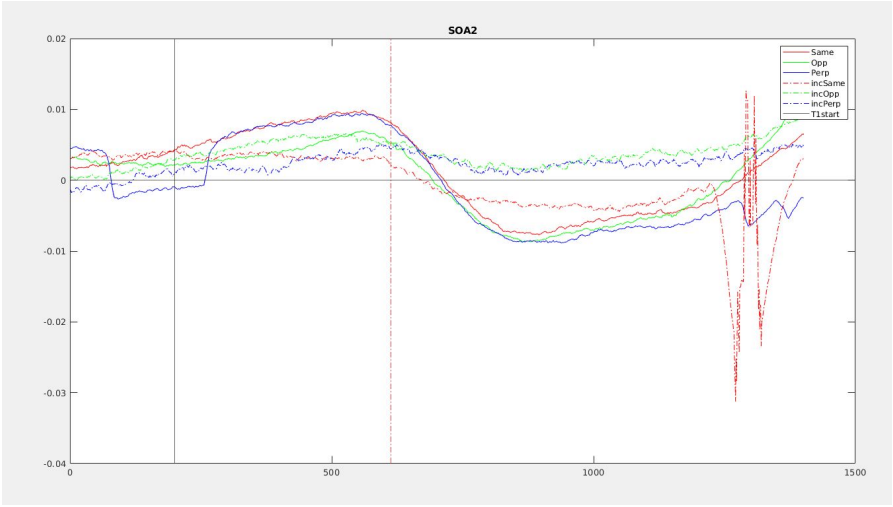


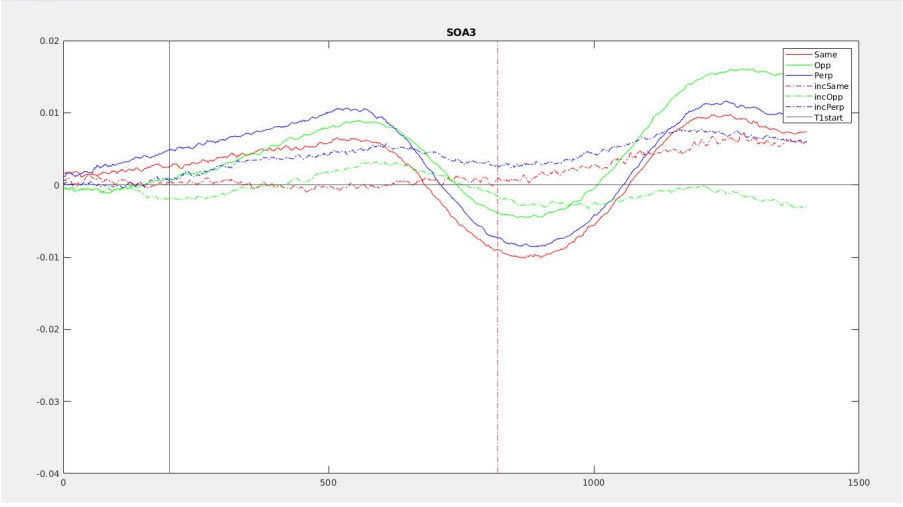




Baseline= Signal - mean (From first point until T1Onset)
N = 12







Baseline= Signal - mean (From first point until T1Onset)
N = 12

Next Step

Latency Calculation:

• Constrict latency: Time @ which we see constriction in pupil diameter.
• Dilate latency: Time @ which pupil starts dilating again.

Constrict latency calculation

• Collate data across participants for different conditions, e.g. for same soa1 there are S = 165 X 1400 elements, representing T1-200 to T1+ 1200 ms
• Take data from T1 onset
• Compare each adjacent column elements through this statistical test
• For I = 1:1400-1
• [h,p] = ttest(S(:,i),S(:,i+1))
• If h == 1; break: This will given you first point in time where pupil starts constricting.

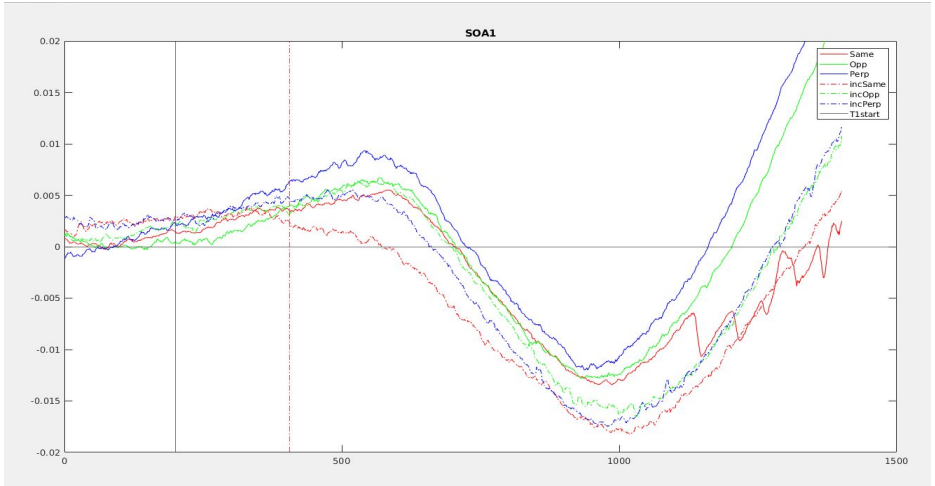
Next Step

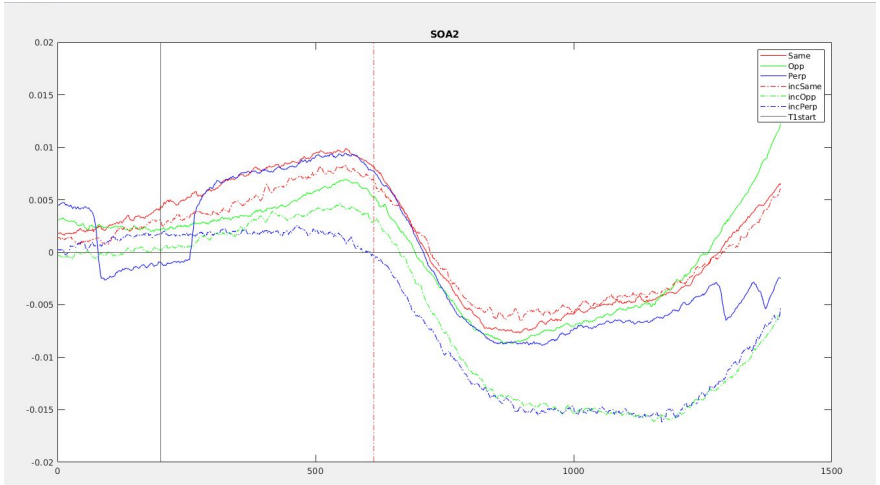
Dilate latency calculation

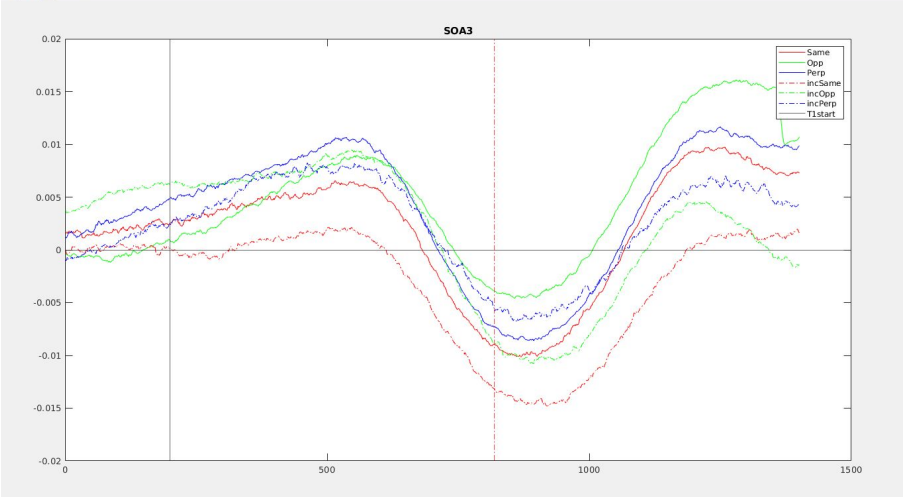
• Collate data across participants for different conditions, e.g. for same soa1 there are S = 165 X 1400 elements, representing T1-200 to T1+ 1200 ms
• Take data from minimum point after T1 onset
• Compare each adjacent column elements through this statistical test
• For i = min(S): (end-1)
• [h,p] = ttest(S(:,i),S(:,i+1))
• If h == 1; break: This will given you first point in time where pupil starts dilating .

• Replicate figure 2A & 2D of Zylberberg et al, 2012.
• Replicate slides 72 & 75 with errors in response 3 (misclassification into wrong direction)
• Compile the trends in the data.
• Also share the figure files as PDF in our shared folder.
• A plot from first point until T1 onset.

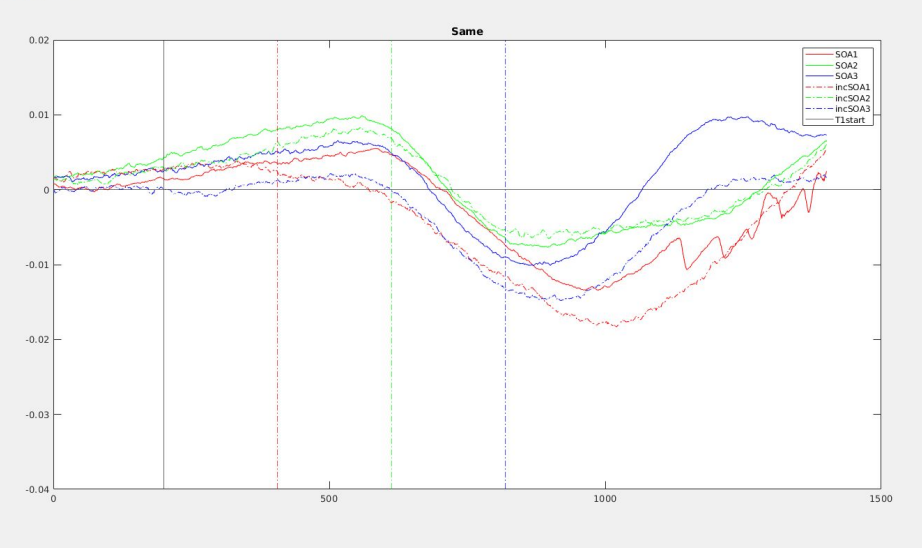
Next graphs are for when participant said that there is 2nd object but wrongly identified the direction of the 2nd object

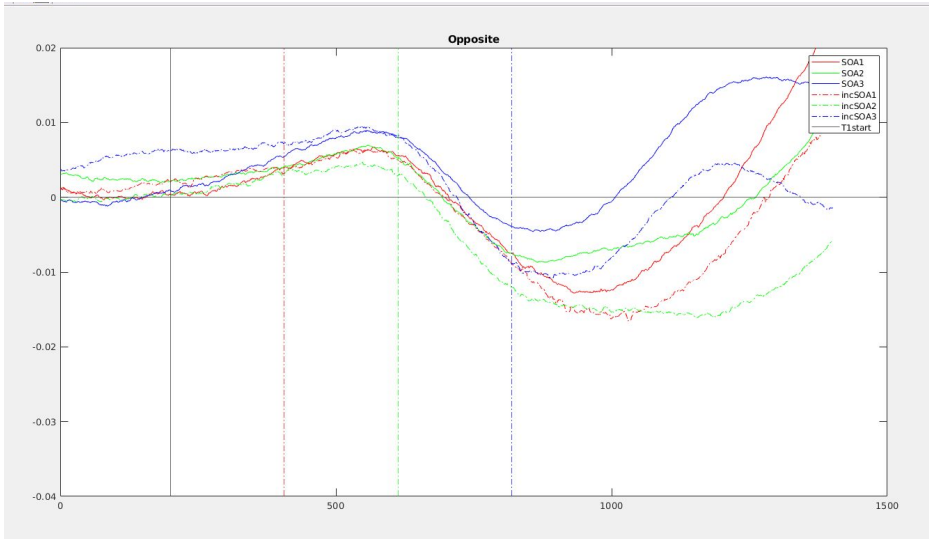


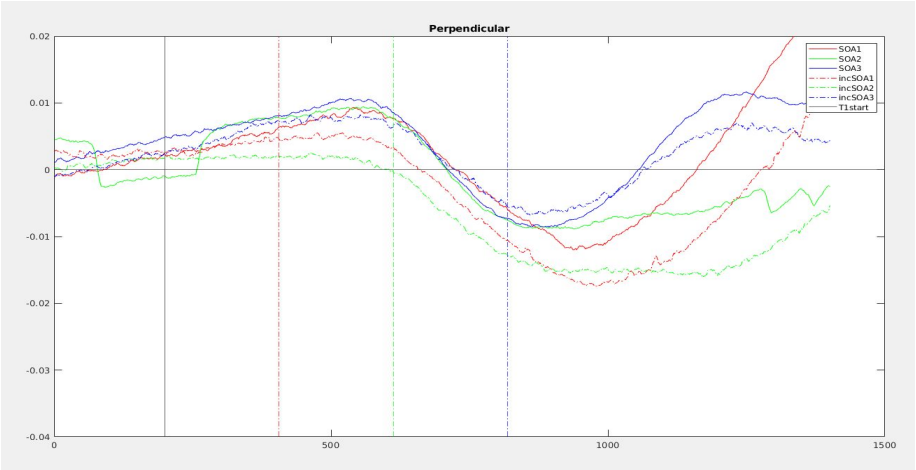




Baseline= Signal - mean (From first point until T1Onset)
N = 12



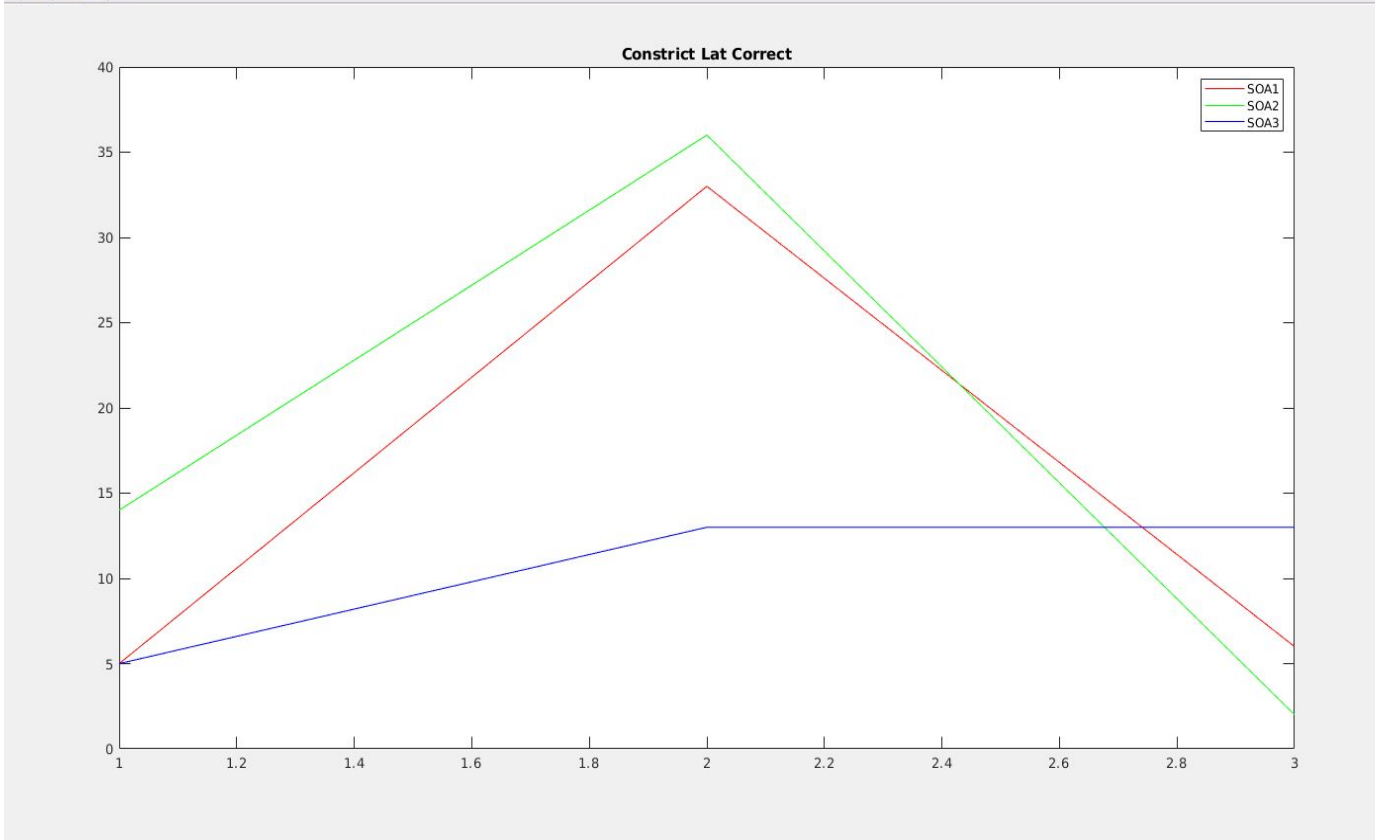




Baseline= Signal - mean (From first point until T1Onset)
N = 12

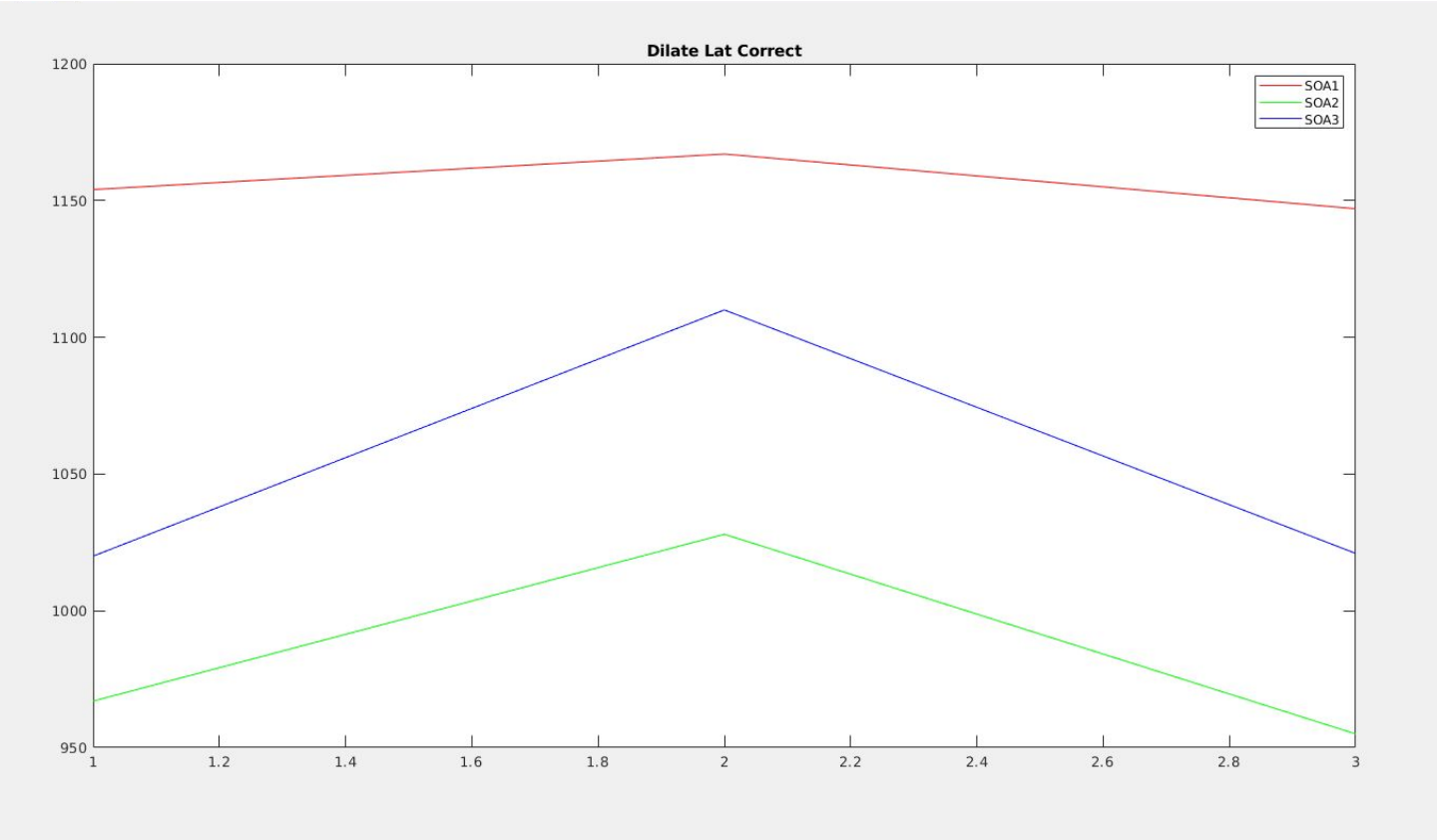
Constrict Latency Graph

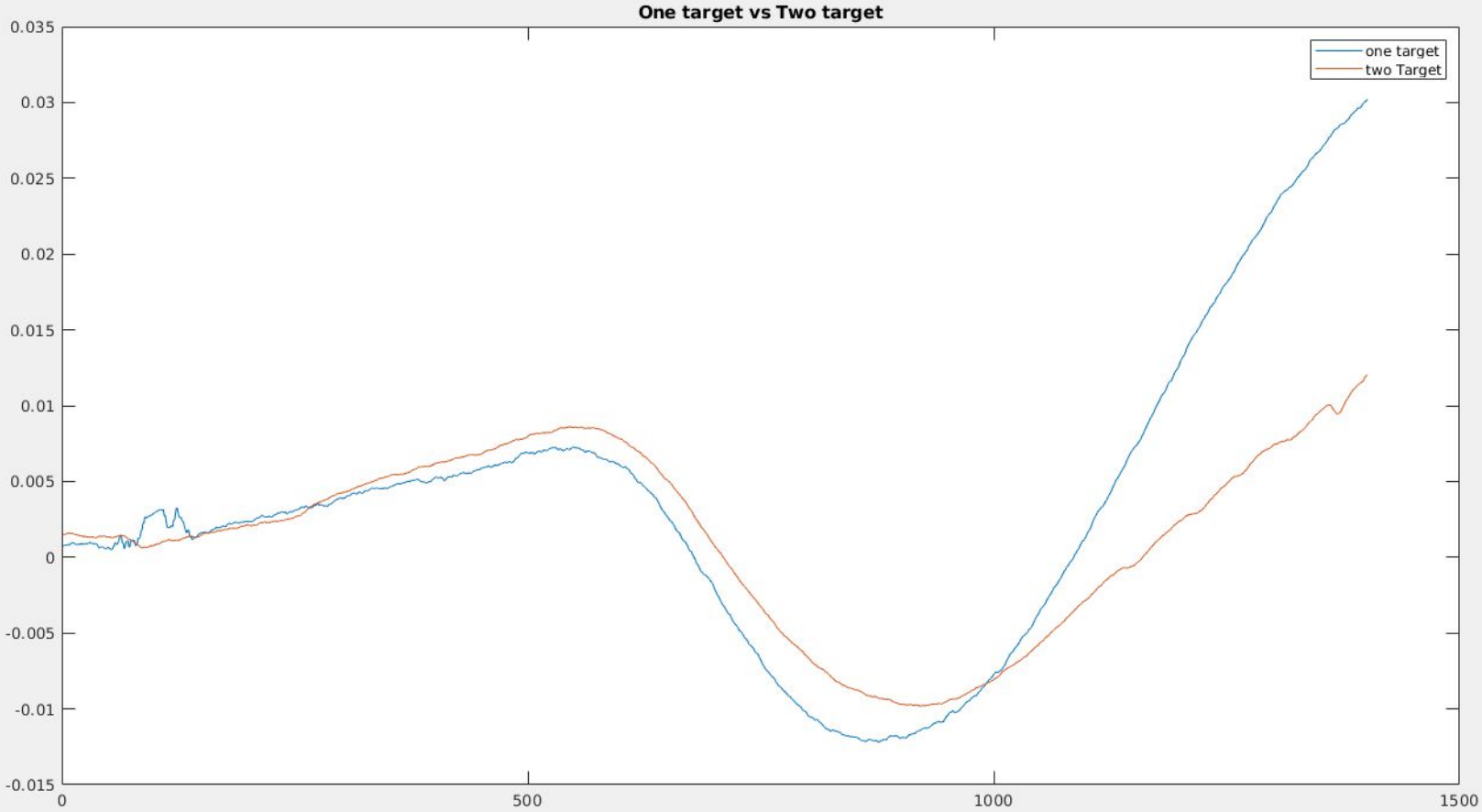
On y axis point where h==1
On x axis plotted 3 points (1,2,3) showing SOA

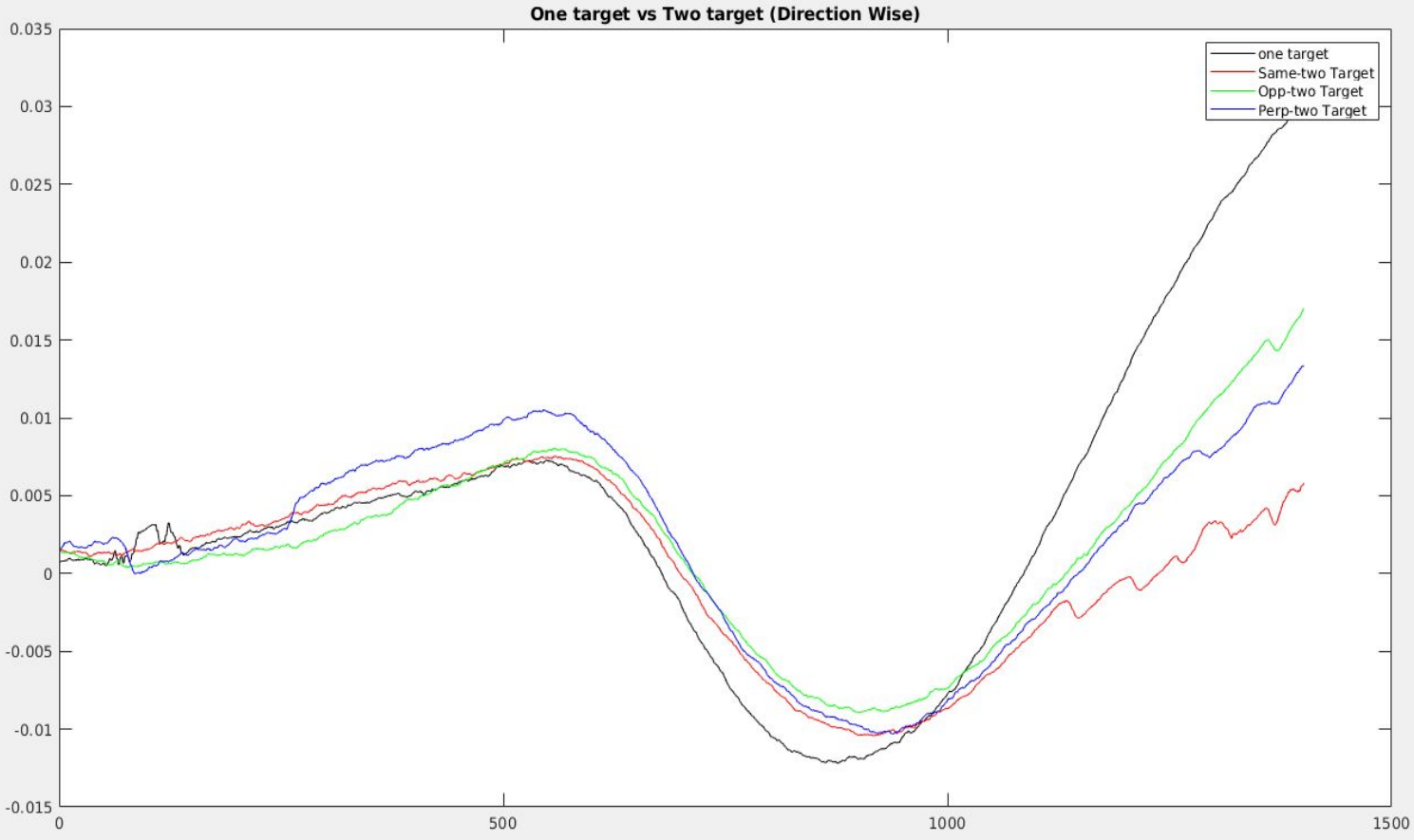


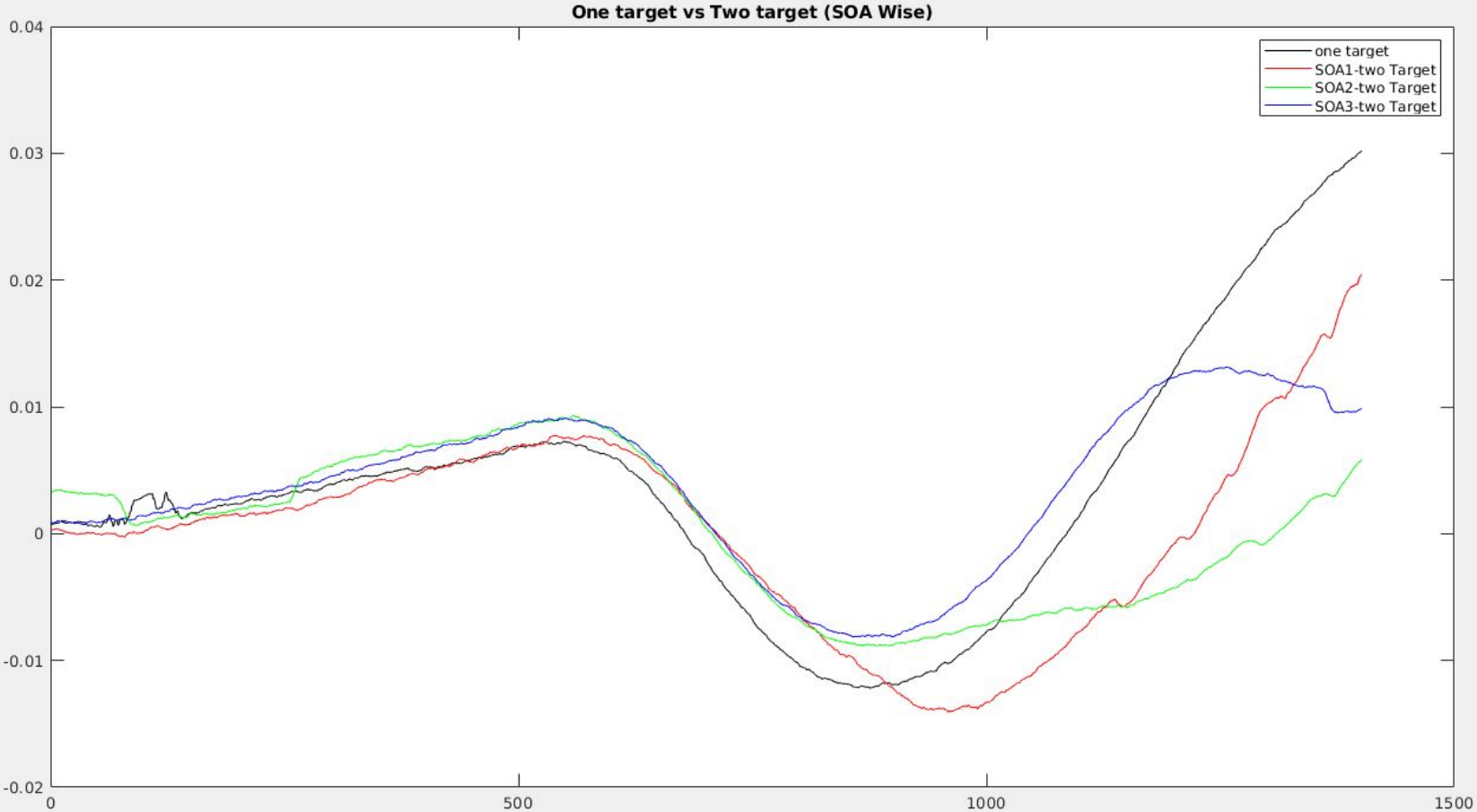
Dilate Latency Graph

On y axis point where h==1
On x axis plotted 3 points (1,2,3) showing SOA




One target vs Two target
one target
two Target
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
0
500
1000
1500


One target vs Two target (Direction Wise)
one target
Same-two Target
Opp-two Target
Perp-two Target
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
-0.005
-0.01
-0.015
0
500
1000
1500


One target vs Two target (SOA Wise)
0.04
0.03
0.02
0.01
0
-0.01
-0.02
0
500
1000
1500
one target
SOA1-two Target
SOA2-two Target
SOA3-two Target