Quality control for ISC between group analysis

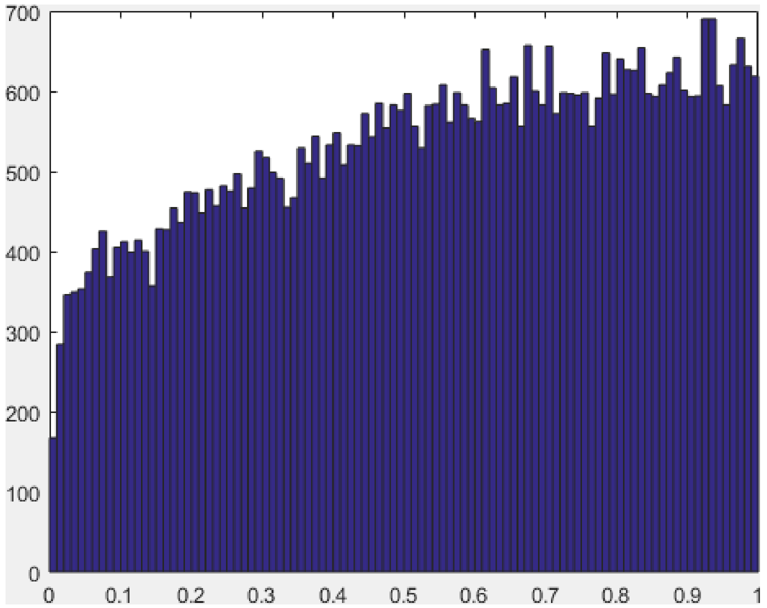
Jussi Tohka with input from Juha Salmitaival, Mostafa Metwaly, Frank E. Pollick, Greta Mohr, Dewy Nijhof

Motivation

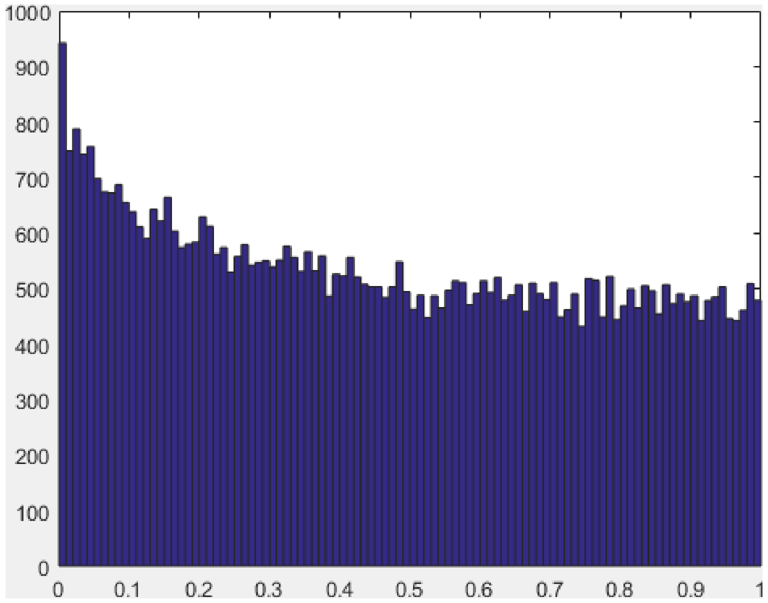
- Finding ISC differences between two groups is complicated by artifacts, which may cause certain subset of subjects to be very correlated just due to artefacts.
- These artefacts are reduced by proper pre-processing sometimes including ICA based artefact correction, global signal regression and/or regression of white matter and CSF signals.
- In our experience, these steps do not guarantee a good dataset for the analysis of group differences

How to detect that something goes wrong?

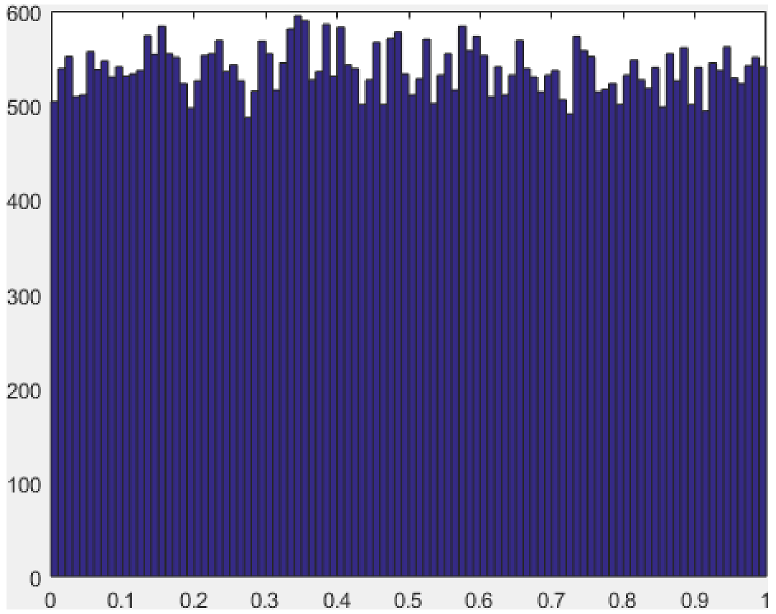
- Take a look to uncorrected p-value histograms across the brain mask:
  - `load('\data\ISCgroup\PFsession\band0pvalmaps3Dindvox1.mat')`
  - `hist(pVsindvox,100)`
- Histograms fall in roughly three classes:



Bad histogram. Something goes wrong.



Good histogram. An effect in part of the voxels
While others do not differ between the groups.



No effect histogram. There are no effects but
nothing curious happens.

Why bad histogram is bad?

- If there are no differences between the groups, one would expect the p-values across the brain to be uniformly distributed (no effect histogram)
- If there is a group difference in part of the voxels, then the histogram is a mixture of uniform distribution and unknown distribution of small p-values (good histogram)
- However, the bad histogram should not occur and indicates that some other labeling of data (into group1, group2) would bring more differences than the current labeling. This can be due to confounds, or much more likely, confounding artefacts to which ISC is very sensitive to (e.g. motion artefacts).
- Bad motion artefacts completely change the correlation between time-series

How to check the data?

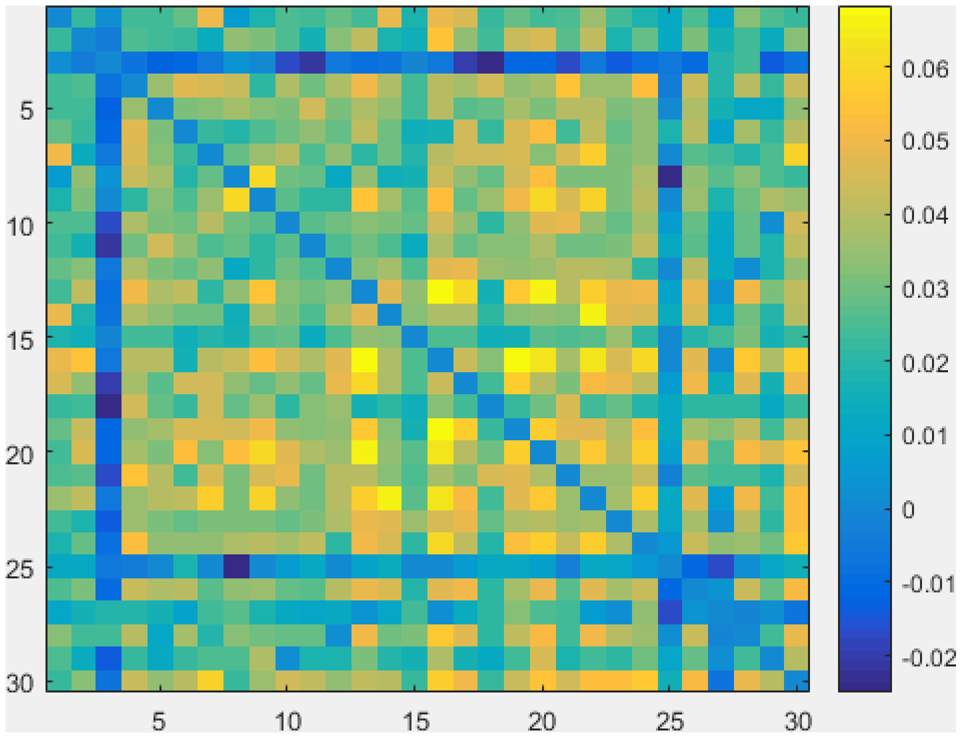
- ISC toolbox provides two tools or metrics for checking the data.
- These are available in a file : debugGroupComparison.m, which takes the params file as an input and saves the file GroupCompDebug.mat to the GroupCompDebug directory. **This only works with already run analysis, and expects that memory maps of the original data still exist**.
-
```
% load the parameter file
load('/data/ISCgroup/ISC_params')
% run the debugging function
[cz,c,t,tn] = debugGroupComparison(Params);
```

How to check the data?

- Then do:
- `load(’\Data\ISCgroup\GroupCompDebug\GroupCompDebug.mat')`
- `imagesc(corrmat + corrmat')`
- `colorbar`

This shows pairwise ISCs averaged over brain mask for two groups (see next page)

How to check the data? Average correlation



This matrix is from an application where there was 15 subjects per group (group1 and group2). The subjects 1 to 15 are group1 and 16 to 30 are group2. One can see clearly that subjects number 3 (group 1 no 3) and 25 (group2 no 10) are different (not correlated with others) and should probably be removed.

A quantitative metric for this is that if the average correlation between a subject and other subjects is very small (below zero), then the subject should be removed.

The indexes can found by saying:

```
find(mean(corrmat + corrmat') < 0)
```

However, the zero may not be the correct threshold, but watch for the subjects that are clearly different from others.

I have no idea why certain subjects show reduced correlations to other across the whole brain

How to check the data? Average time-series

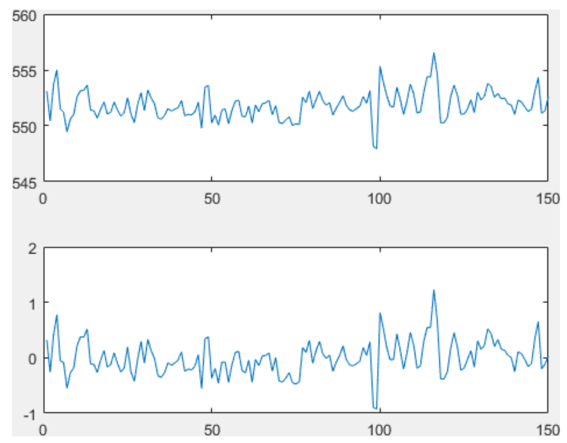
- It is a good idea to also look at the subject time-series averaged over whole brain mask to be able to detect jumps in time series.
- These are stored in variables `timeseries` and `timeseries_normalized`.
- `Timeseries` is just a raw average of voxel-wise timeseries and `timeseries_normalized` is an average of de-meaned and standardized (to unit variance) timeseries. The latter is more useful as it is analogous to the computation of correlation coefficient.
- We provide a script `plotGroupDebug.m` for going through these timeseries:

```
for g = 1:2
    for i = 1:size(timeseries,3)
        disp([g i])
        subplot(2,1,1)
        plot(timeseries(:,g,i))
        subplot(2,1,2)
        plot(timeseries_normalized(:,g,i))
        pause
        close all
    end
End
```

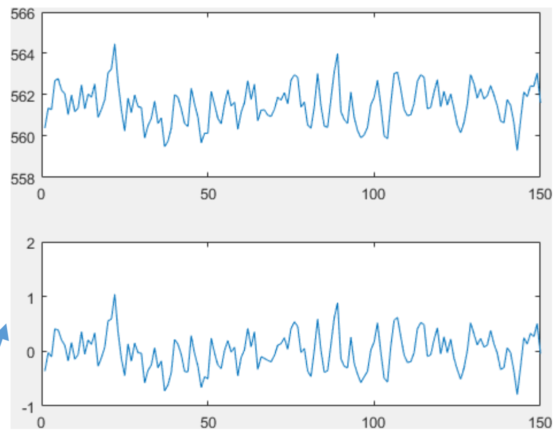
- It displays timeseries of one subject and then pauses until a button press.

How to check the data? Average time-series

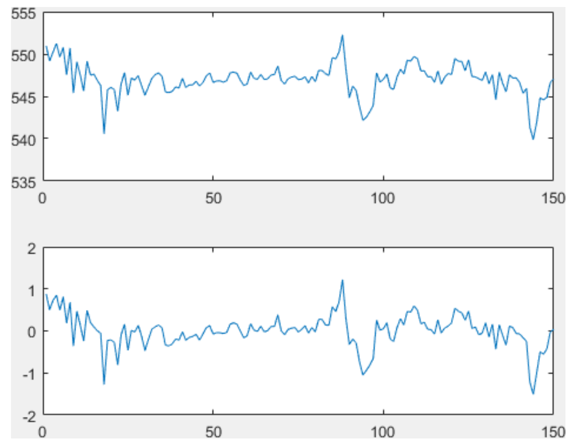
Ok



Ok

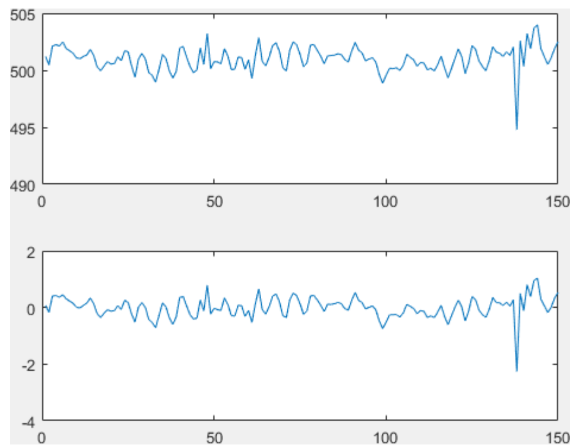


OKish



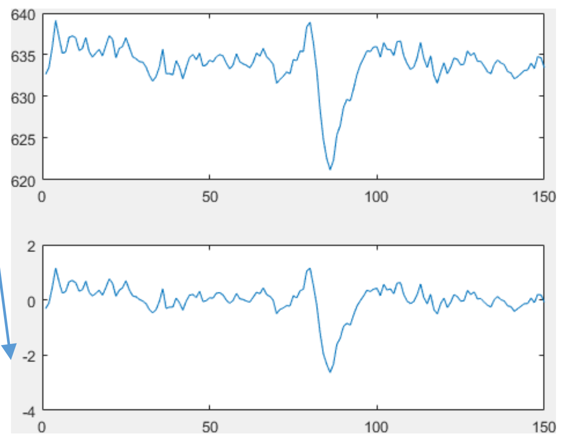
Important! Pay attention to the limits of the axes. If min or max is more than about 2.3 that is not good. Also, if the normalized scaling does not appear proper, that could be indicative of problems

Bad



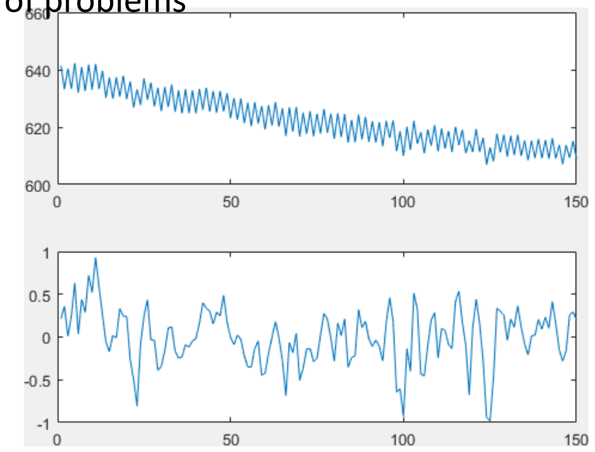
Large jump at 130

Bad



Jump at 90

Bad



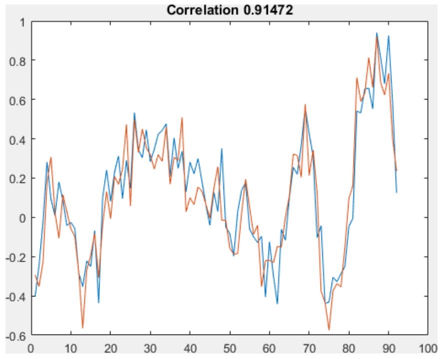
Un-normalized series indicates drift

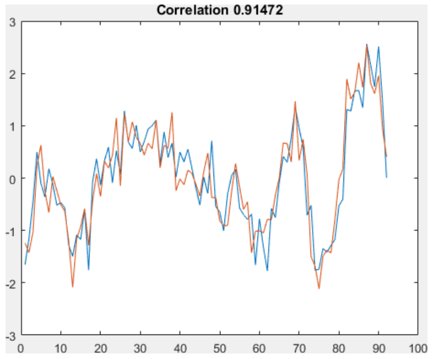
Why are jumps bad?

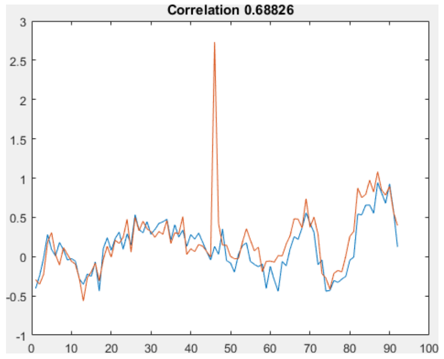
- First, note that we check for large jumps in the average time series across the brain mask
- These are not related to neural responses as changes in the neural responses are sufficiently localized and averaging over the whole brain should result only in moderate changes in neural response (I would argue that most of the arupt changes in average time series are due to motion artefacts, but I don’t have any hard evidence to support the argument).
- Large jumps often are followed by a minor change in the baseline level of the signal
- These effects alter the correlations as shown in the next slide!

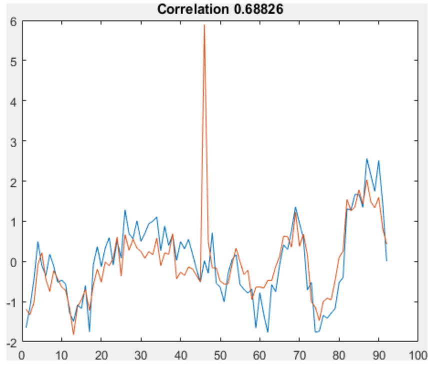
Why are jumps bad?

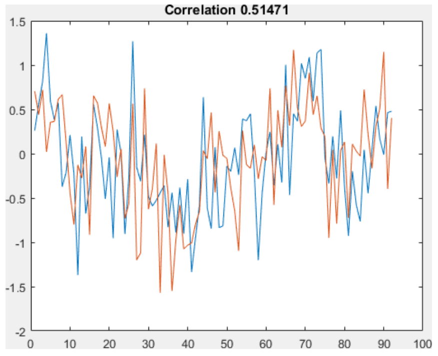
Examples on effects of noise peaks followed by a light (almost unnoticeable) baseline change on correlations. This on the level of individual time-courses

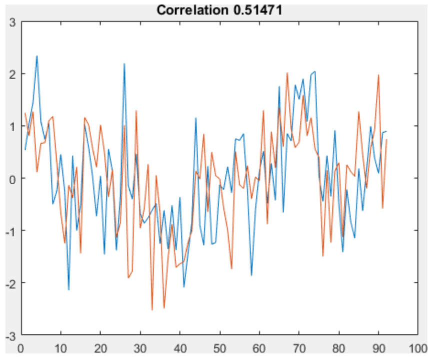


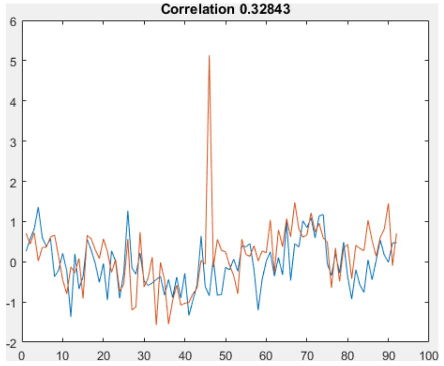


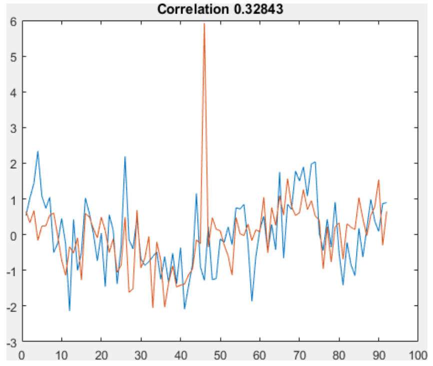


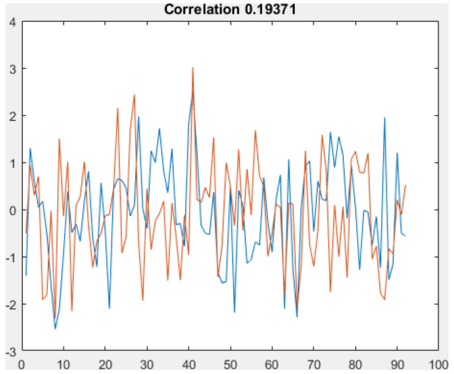


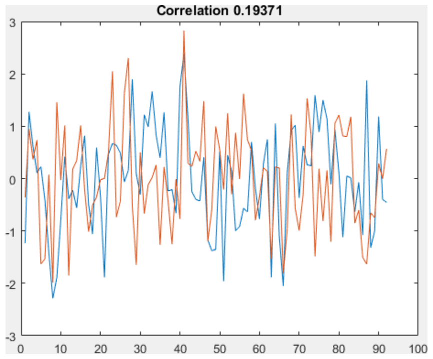


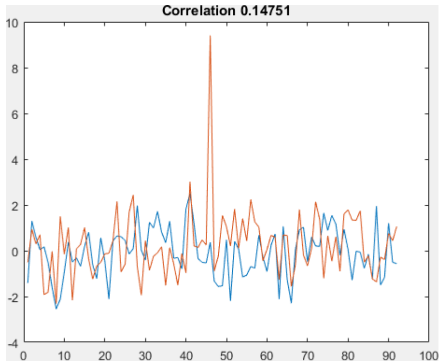


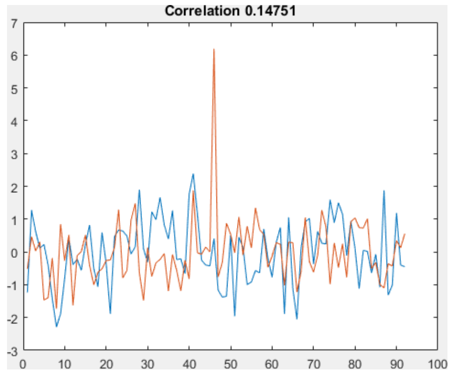








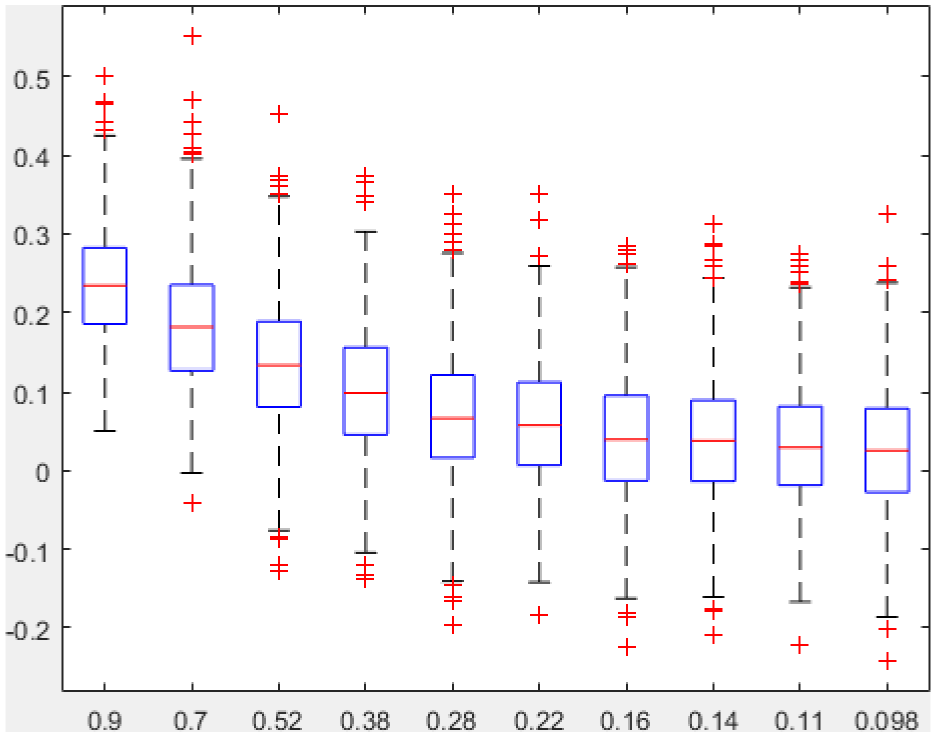




From left: 1st column: two time courses with different correlations; 2nd column: the same time-series normalized; 3rd column: the same time courses with noise peak followed by a baseline change in one of them; 4th column: the same as the column 3 but with normalized time-series

Why are jumps bad?

A more quantitative account of the previous example: the box plot shows the change in correlation coefficient as a result of adding the peak and the baseline change. The labels in axis are the average values of the original correlation.



The code used to generate the plot

```
tf =100;
fl = 9;
iter = 1000;
avec = [0.1:0.1:1];
for j = 1:length(avec)
  for i = 1:iter
    tf = 100;
    x = randn(tf,1);
    x = conv(x,(1/fl)*ones(fl,1),'valid');
    tf = length(x);
    a = avec(j);
    y = x + a*randn(length(x),2);
    c = corr(y(:,1),y(:,2));
    cc(i,j) = c;
    s(1) = std(y(:,1));
    s(2) = std(y(:,2));
    y(tf/2,2) = 8*s(2);
    y(tf/2:tf,2) = y(tf/2:tf,2) + 0.5*s(2);
    c2 = corr(y(:,1),y(:,2));
    co(i,j) = c - c2;
  end
end
boxplot(co)
for j = 1:length(avec)
  xl{j} = num2str(mean(cc(:,j)),2);
end
Xticklabels(xl)
```

Thanks!

- Frank E. Pollick, University of Glasgow, UK for the use of data to prepare these slides
- Juha Salmitaival, Mostafa Metwaly for additional examples that have helped to prepare the QC.
- Greta Mohr, Dewy Nijhof, and Frank E. Pollick for comments