

Averaged correlations and correlated averages

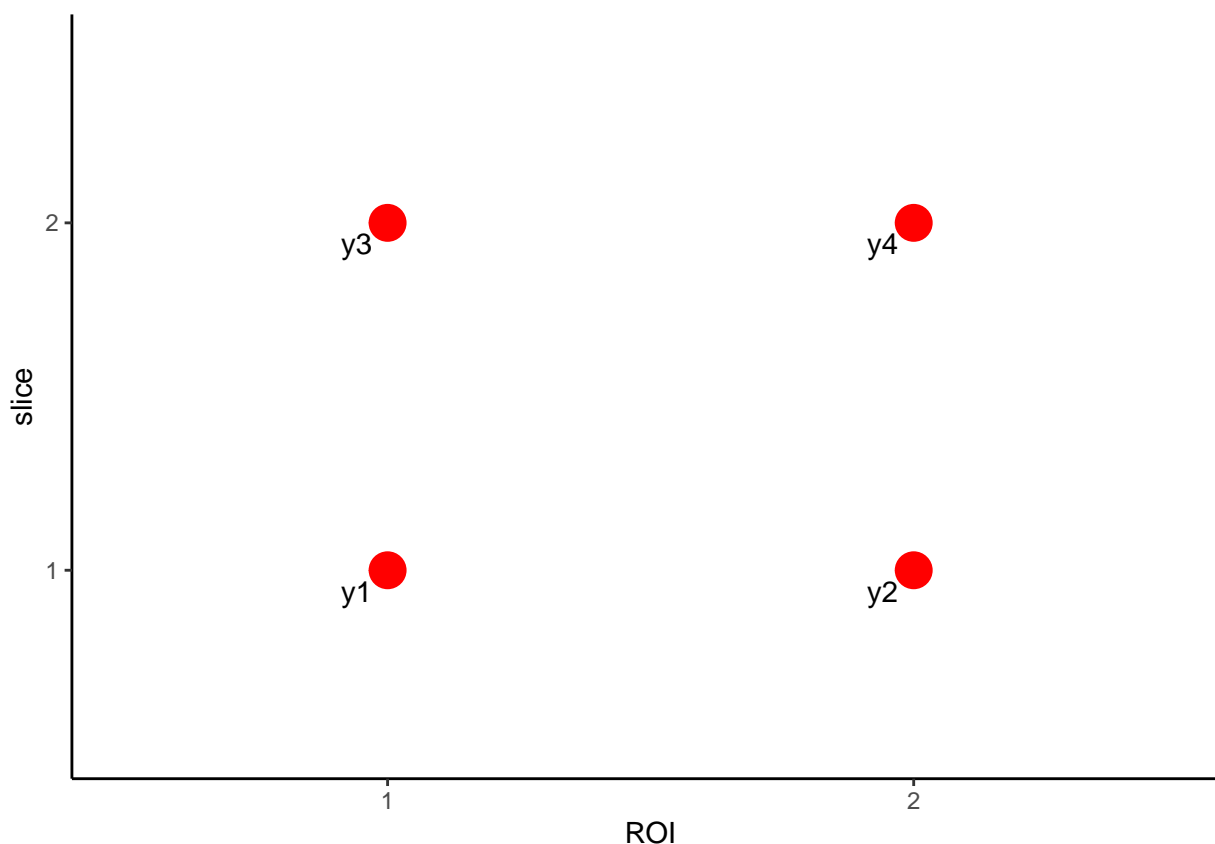
Ulrike Horn

11 4 2023

In this document we explore the difference between averaging correlations of resting-state fMRI time series and correlating average resting-state fMRI time series. To run the R-code in this notebook or explore further options on your own, you need to install the packages “MASS” and “tidyverse”.

Toy example

Let us assume that we have acquired two axially-oriented EPI slices from the spinal cord. In both of these, we separately obtain the time series of two ROIs (e.g. left dorsal horn and right dorsal horn). We would now like to assess the correlation between these two ROIs and have two options for this: we could either obtain the correlation between the two ROI time series within each slice separately and then average these correlations over slices, or we could first average the time series within each ROI over the two slices and only then determine their correlation.

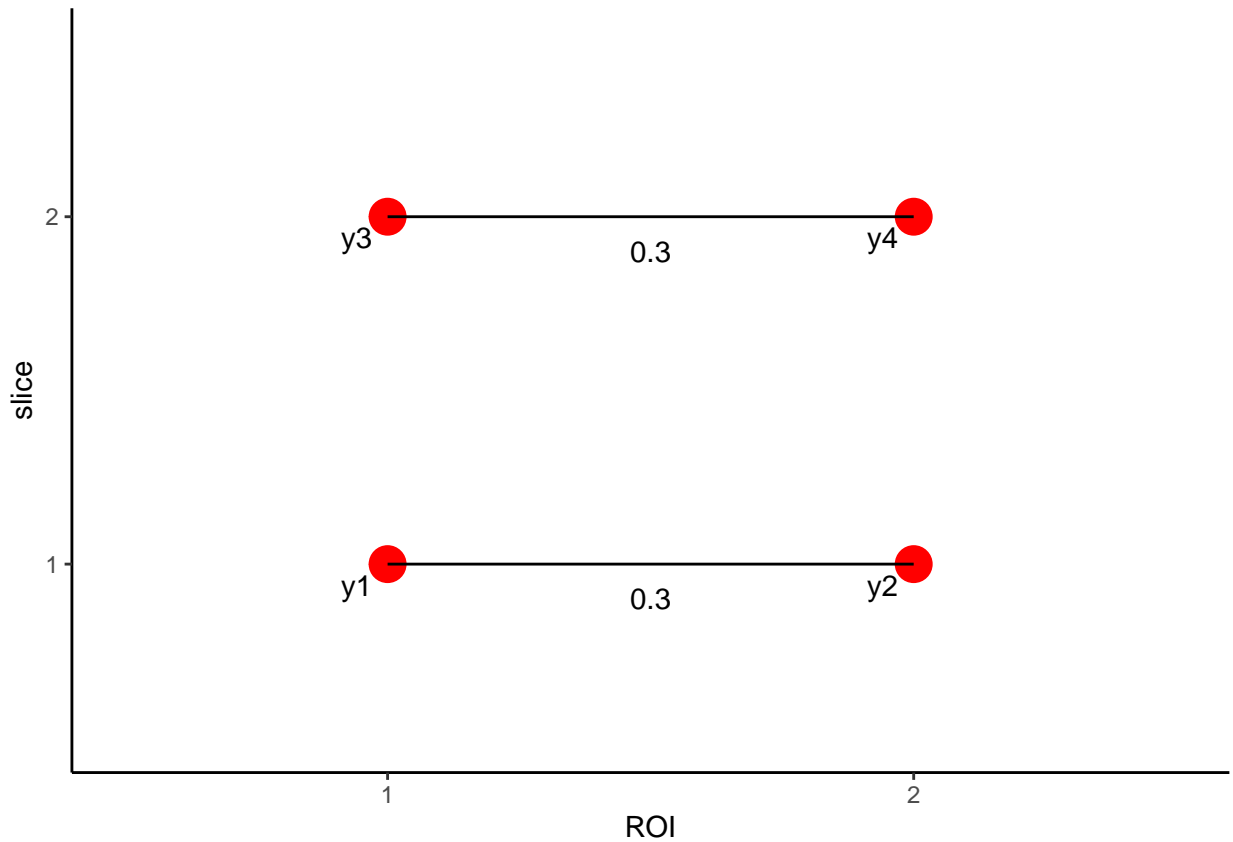


In any case, to construct random time series that have a specific correlation structure, we will use the following function:

```
cor2cov <- function(R, S) {  
  sweep(sweep(R, 1, S, "*"), 2, S, "*")  
}
```

1. No between slice correlation

For our first example, let us assume that the time series correlations between the two regions are 0.3 within each slice (so that a slice-wise average of these would also be 0.3).



```
# within slice correlation
wsc <- 0.3
# between slice correlation
bsc <- 0.0
# standard deviation of each time series
my_sd <- 0.2

# correlation matrix
r <- matrix(c(1, wsc, bsc, bsc,
              wsc, 1, bsc, bsc,
              bsc, bsc, 1, wsc,
              bsc, bsc, wsc, 1),
            4, 4)
# vector of standard deviations for all time series
s <- rep(my_sd, 4)

# convert to covariance matrix
my_cov <- cor2cov(r, s)

# create random normal variables y1 to y4 with length 50
# with mean 0 based on that covariance matrix
```

```
Y <- mvrnorm(n = 50, rep(0,4), my_cov, empirical = TRUE)
df <- tibble(y1 = Y[,1], y2 = Y[,2], y3 = Y[,3],
            y4 = Y[,4], x = c(1:50))
```

The correlation matrix looks as expected, with correlations of 0.3 between the two regions within slices and correlations of 0 everywhere else:

```
# check correlation
cor_matrix <- df %>%
  select(-x) %>%
  cor() %>%
  as.matrix()

(round(cor_matrix,digits=1))
```

```
##      y1 y2 y3 y4
## y1 1.0 0.3 0.0 0.0
## y2 0.3 1.0 0.0 0.0
## y3 0.0 0.0 1.0 0.3
## y4 0.0 0.0 0.0 1.0
```

But what happens if we take the alternative approach and first average the time series across slices and only then calculate the correlation? We obtain the same result:

```
first_roi_mean <- rowMeans(df[, c('y1', 'y3')])
second_roi_mean <- rowMeans(df[, c('y2', 'y4')])

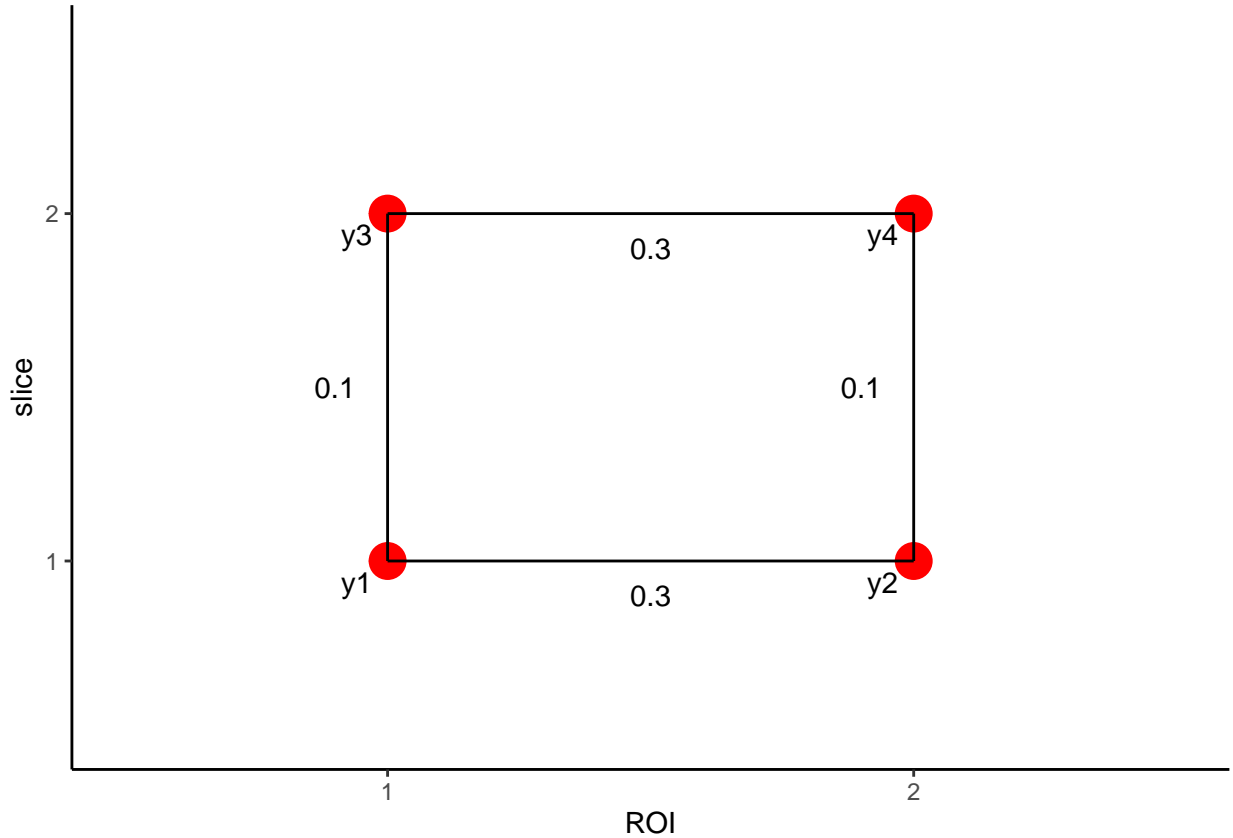
cor_avg_without_betw_slice <- cor(first_roi_mean, second_roi_mean)
cor_avg_without_betw_slice
```

```
## [1] 0.3
```

So, in this very simple case the average of correlations and the correlation of averages is the same.

2. Between slice correlation within same region

Let us now add time series correlations within the same region, but across different slices, for example having ROI1 and ROI2 time series correlated with a strength of 0.1 (meaning y1 is correlated with y3 and y2 is correlated with y4).



```
# within slice correlation
wsc <- 0.3
# between slice within ROI correlation
bscROI <- 0.1
# standard deviation of each time series
my_sd <- 0.2

# correlation matrix
r <- matrix(c(1, wsc, bscROI, 0,
              wsc, 1, 0, bscROI,
              bscROI, 0, 1, wsc,
              0, bscROI, wsc, 1),
            4, 4)
# vector of standard deviations for all time series
s <- rep(my_sd, 4)

# convert to covariance matrix
my_cov <- cor2cov(r, s)

# create random normal variables y1 to y4 with length 50
```

```
# with mean 0 based on that covariance matrix
Y <- mvrnorm(n = 50, rep(0,4), my_cov, empirical = TRUE)
df <- tibble(y1 = Y[,1], y2 = Y[,2], y3 = Y[,3],
             y4 = Y[,4], x = c(1:50))
```

The correlation matrix looks as expected, with correlations of 0.3 between the two regions within slices, correlations within regions between slices of 0.1 and correlations of 0 everywhere else.

```
# check correlation
cor_matrix <- df %>%
  select(-x) %>%
  cor() %>%
  as.matrix()
(round(cor_matrix,digits=1))
```

```
##      y1 y2 y3 y4
## y1 1.0 0.3 0.1 0.0
## y2 0.3 1.0 0.0 0.1
## y3 0.1 0.0 1.0 0.3
## y4 0.0 0.1 0.3 1.0
```

But what happens if we take the alternative approach and first average the time series across slices and only then calculate the correlation? The obtained correlation is smaller than our average of correlations in this case:

```
first_roi_mean <- rowMeans(df[, c('y1', 'y3')])
second_roi_mean <- rowMeans(df[, c('y2', 'y4')])

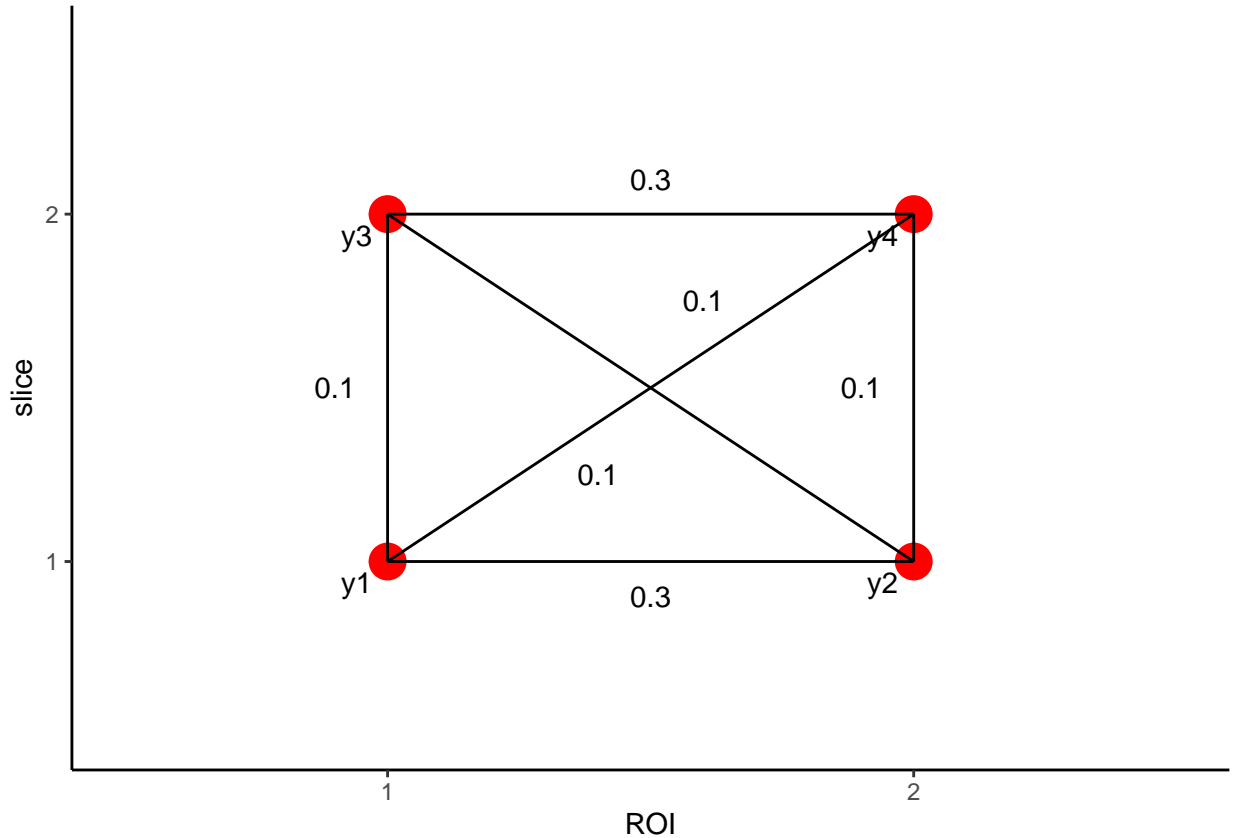
cor_avg_with_betw_slice <- cor(first_roi_mean, second_roi_mean)
cor_avg_with_betw_slice
```

```
## [1] 0.2727273
```

We will follow up on this in the next sections and provide an explanation for this behavior.

3. Between slice correlation across different regions

Considering neurobiological complexity, it might actually be valid to assume that there is also a correlation between different regions in different slices, so for example from left dorsal horn in slice 1 to right dorsal horn in slice 2. Let us simulate this assuming a correlation strength of 0.1.



```
# within slice correlation
wsc <- 0.3
# between slice within ROI correlation
bscROI <- 0.1
# between slice across ROIs correlation
bsc <- 0.1
# standard deviation of each time series
my_sd <- 0.2

# correlation matrix
r <- matrix(c(1, wsc, bscROI, bsc,
              wsc, 1, bsc, bscROI,
              bscROI, bsc, 1, wsc,
              bsc, bscROI, wsc, 1),
            4, 4)
# vector of standard deviations for all time series
s <- rep(my_sd, 4)

# convert to covariance matrix
my_cov <- cor2cov(r, s)
```

```
# create random normal variables y1 to y4 with length 50
# with mean 0 based on that covariance matrix
Y <- mvrnorm(n = 50, rep(0,4), my_cov, empirical = TRUE)
df <- tibble(y1 = Y[,1], y2 = Y[,2], y3 = Y[,3],
             y4 = Y[,4], x = c(1:50))
```

The correlation matrix again looks as expected:

```
# check correlation
cor_matrix <- df %>%
  select(-x) %>%
  cor() %>%
  as.matrix()
(round(cor_matrix,digits=1))
```

```
##      y1 y2 y3 y4
## y1 1.0 0.3 0.1 0.1
## y2 0.3 1.0 0.1 0.1
## y3 0.1 0.1 1.0 0.3
## y4 0.1 0.1 0.3 1.0
```

But what happens in this case if we take the alternative approach and first average the time series across slices and only then calculate the correlation? The obtained correlation is larger than our average of correlations in this case:

```
first_roi_mean <- rowMeans(df[, c('y1', 'y3')])
second_roi_mean <- rowMeans(df[, c('y2', 'y4')])

cor_avg_with_betw_slice <- cor(first_roi_mean, second_roi_mean)
cor_avg_with_betw_slice

## [1] 0.3636364
```

So, depending on the overall correlational structure of the network, the correlation of average time series can deviate quite substantially from the average of correlations.

Background

This phenomenon has already been investigated extensively in social and economic studies [1]. For example, Nickerson et al., 1995 [2], discussed an example from another paper in which respondents had to rate two different properties X and Y for different instances. The original authors then correlated the averaged responses for X and Y made by different respondents and claimed that they have shown that a high response in X is associated with a high response in Y. Nickerson et al. point out that this claim is wrong, as the original hypothesis is a within-respondent hypothesis. Here the average of all within-respondent correlations would have been more appropriate to answer this question.

In our case, the situation is not as clear-cut, since our hypotheses aim at answering questions such as whether the left dorsal horn is functionally connected with the right dorsal horn - and one might thus ask: what exactly encompasses the dorsal horn? One could argue that the signals in two neighboring slices are so similar that we mean both of them if we speak of the left dorsal horn for example. But one also has to acknowledge that the signals in the left dorsal horn of segment C5 might be quite different from those of the left dorsal horn of segment C7, for example.

In any case, to determine the exact relationship between the correlation of averages and the average of correlations one can use the following formula (given in Nickerson et al., 1995):

$$r_{\overline{XY}} = \frac{\frac{1}{N} \cdot \overline{r_{XY}} + \frac{(N-1)}{N} \cdot \overline{r'_{XY}}}{\sqrt{\frac{1}{N} + \frac{(N-1)}{N} \cdot \overline{r'_{XX}}}} \cdot \sqrt{\frac{1}{N} + \frac{(N-1)}{N} \cdot \overline{r'_{YY}}}}$$

While it does look complicated, it is actually pretty simple. When we want to determine the association between X and Y, i.e. in our case the time series of the left dorsal horn and the time series of the right dorsal horn (with N being the number of slices):

$r_{\overline{XY}}$ is the correlation of averages (meaning that the time series of different slices have been averaged first and only afterwards a correlation was calculated).

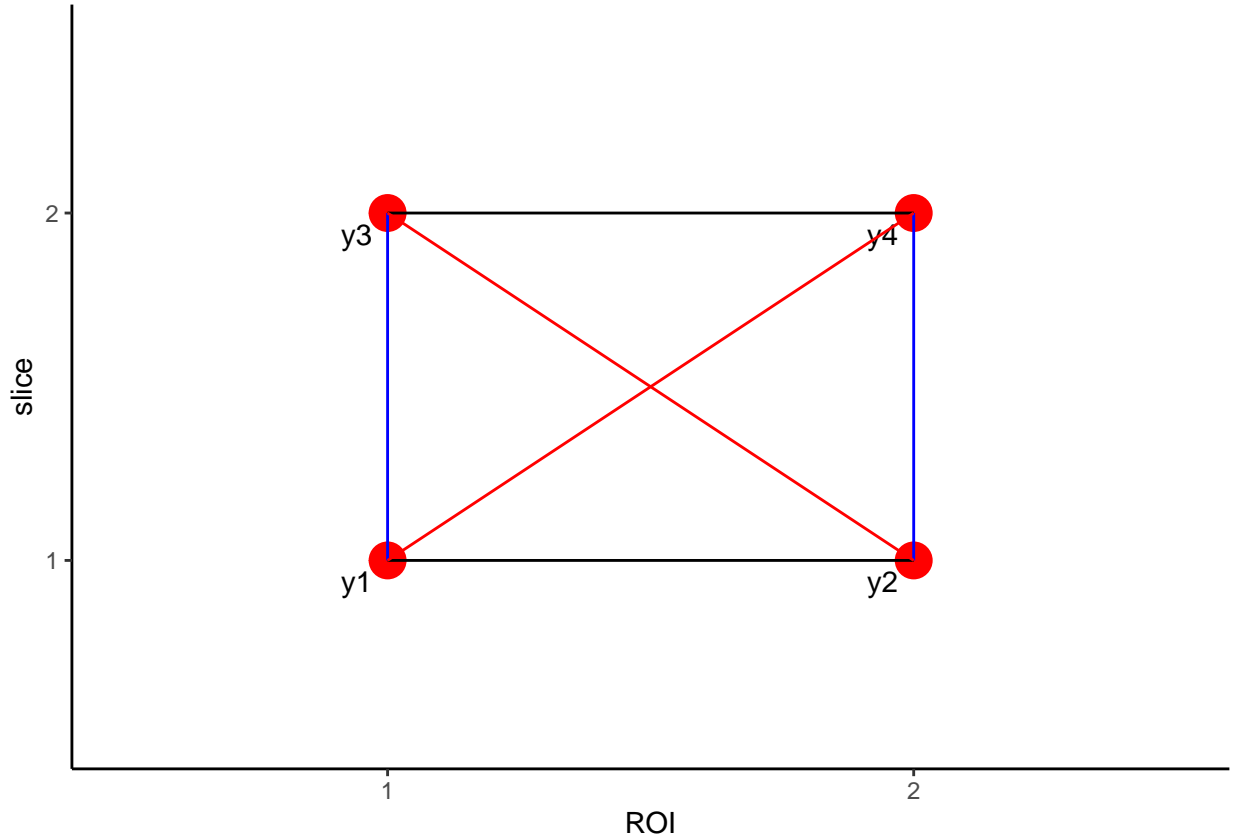
$\overline{r_{XY}}$ is the average of correlations (meaning the slice-wise approach has been applied where correlations are obtained first and then averaged).

$\overline{r'_{XY}}$ is the average of the correlations between every slice's left dorsal horn time series (X) and every other slice's right dorsal horn time series (Y).

$\overline{r'_{XX}}$ is the average of the correlations between every slice's left dorsal horn time series (X) and every other slice's left dorsal horn time series (X).

$\overline{r'_{YY}}$ is the average of the correlations between every slice's right dorsal horn time series (Y) and every other slice's right dorsal horn time series (Y).

This formula provides a clear explanation how the different variables influenced our results in the above simulations: the larger the red connections, the larger is the correlation of averaged time series, and the larger the blue connections, the smaller is the correlation of averaged time series (as they are in the denominator). The exact relation will obviously also depend on the number of slices considered.



In our case, we deemed the slicewise approach to be the most reasonable one, as we consider signals from different segments of the spinal cord to belong to different functional entities and we also did not want to inflate our correlations simply because signals within a segment might be highly interconnected.

References

- [1] *Monin & Oppenheimer, Correlated averages vs. averaged correlations: demonstrating the warm glow heuristic beyond aggregation, Social Cognition 23 (3), 257-278 (2005)*
- [2] *Nickerson, Does Willingness to pay reflect the purchase of moral satisfaction? A reconsideration of Kahneman and Knetsch, J. Environ. Econ. Manag. 28, 16-133 (1995)*
- [3] *Paumonen & Gardner, Biases resulting from the use of aggregated variables in psychology, Psychol. Bull. 109, 520-523 (1991)*