

```
#####
#####
#
#          Error in the affine matrix          #
#
#####
#####

#####
# Description of the error          #
#####
```

It has come to our attention (thanks to Eleftherios Garyfallidis) that there is a mismatch between the voxel size as given by the pixel resolution and the voxel size if you extract it using the affine information for the diffusion weighted data (DWI) which has been made available on:

[https://www.nitrc.org/frs/?group\\_id=990](https://www.nitrc.org/frs/?group_id=990)

and is described in the following publication:

Boekel, W., Forstmann, B.U., Keuken, M.C., (2017). A test-retest reliability analysis of diffusion measures of white matter tracts relevant for cognitive control. *Psychophysiology* 54 (1): 24-33.

The DWI data was acquired with the following sequence:

[www.spinozacentre.nl/wiki/index.php/NeuroWiki:Sequences#DTI32\\_SENSE2](http://www.spinozacentre.nl/wiki/index.php/NeuroWiki:Sequences#DTI32_SENSE2)

and this sequence should have resulted in 2mm isotropic voxels. However there is a mismatch in the header of the shared files. More specifically, in python using nibabel:

```
import nibabel as nib
# loading the data as exported by V2.6:
img1 = nib.load('pp01_dwi_run01_A.nii.gz')

# Give me the voxel size based on the affine info
nib.affines.voxel_sizes(img1.affine)

# output:
#      array([ 2.00000002,  1.99999997,  3.73333342])

# Give me the voxel size based on the header info
img1.header.get_zooms()[:3]

# output:
#      (2.0, 2.0, 2.0)
```

Based on the header info the data is 2mm isotropic, whereas according to the affine info the voxel size in the z-dimension is 3.733mm which is incorrect.

The question is whether the data was actually acquired with a z voxel size of 3.7mm, and then resliced OR whether the tool that was used to export the data damaged the header info and that the affine matrix info is not correct?

```
#####  
# Where does the error come from ? #  
#####
```

I went back to the raw data which were backed up by the first author. The raw data that was exported from the scanner were par/rec files. This is in Philips proprietary format and can be used instead of the DICOM format.

# Par/rec stage?

The par/rec file of pp01\_dwi\_run01\_A has the following header info (as read by MIPAV):

#### Image information

```
Dimension 0:    112  
Dimension 1:    112  
Dimension 2:     60  
Dimension 3:     34  
Type:          Float  
Min:           0.0  
Max:           1072259.625  
Modality:      Magnetic Resonance  
Slice origin upper left corner of image - right hand rule  
Origin X (left to right) :    -0.0  
Origin Y (top to bottom) :    -0.0  
Origin Z:(into the screen):    58.0  
Origin T:(time):              0.0  
Orientation:    Axial  
X axis orientation: right to left  
Y axis orientation: anterior to posterior  
Z axis orientation: inferior to superior  
Pixel resolution 0: 2.0 Millimeters  
Pixel resolution 1: 2.0 Millimeters  
Pixel resolution 2: 2.0 Millimeters  
Pixel resolution 3: 7542.096 Milliseconds  
Slice thickness: 0.0  
Endianness: Little Endian  
Matrix:  
1.0000 0.0000 0.0000 0.0000  
0.0000 1.0000 0.0000 0.0000  
0.0000 0.0000 1.0000 0.0000  
0.0000 0.0000 0.0000 1.0000
```

```

Other information
Description = Magnetic Resonance
Voxel Offset = 352.0
Intent code = No intention
X,Y,Z Coordinate system = Scanner-based anatomical
Source type = FLOAT
Slope scale = 1.0
Added offset = 0.0
Frequency encoding direction = none
Phase encoding direction = none
Slice acquisition direction = none
Axis: x-orientation = Right to Left
Axis: y-orientation = Anterior to Posterior
Axis: z-orientation = Inferior to Superior
X Origin: -0.0
Y Origin: -0.0
Z Origin: 58.0
cal_min = 0.0
cal_max = 0.0
Bits per Pixel = 32
Name or meaning of data =
No extended header is present
Qform Matrix =
2.0000 -0.0000 -0.0000 -0.0000
-0.0000 2.0000 -0.0000 -0.0000
0.0000 0.0000 2.0000 -0.0000
0.0000 0.0000 0.0000 1.0000

```

Based on this information, the par/rec has the expected dimensions (2mm isotropic). So the error does not come from the data acquisition side of things.

# Par/rec -> nifti stage?

The par/rec files were initially exported to Nifti's using an open source tool called r2agui (v 2.6. <https://sourceforge.net/projects/r2agui/> ). When checking the voxel size using nibabel I get the exact same output as when I would use the images that were uploaded to nitrc.org.

So it seems that the r2agui software did not interpret the DWI par/rec data correctly and damaged the affine info.

```

#####
# Solution #
#####

```

In the 5 years that passed between acquiring and exporting the par/rec data to nifti there

has been a new release of the r2agui software.

Using r2agui (v 2.7. which addressed issues with exporting DWI data) I exported the par/rec file again. Now in python I get the following:

```
# loading the data as exported by V2.7:
img2 = nib.load('pp01_dwi_run01_A_v27.nii.gz')

# Give me the voxel size based on the affine info
nib.affines.voxel_sizes(img2.affine)

# output:
#      array([ 2.00000002,  1.99999997,  1.99999995])

# Give me the voxel size based on the header info
img2.header.get_zooms()[:3]

# output:
#      (2.0, 2.0, 2.0)
```

So using the updated version of r2agui (V 2.7) the data is now exported in such a way that the affine voxel sizes correspond to the voxel sizes that were acquired at the scanner.

All structural scans have been re-exported, de-faced with the identical mask, and re-uploaded into a new data release (V2.0)