Script

March 8, 2019

```
In [29]: %load_ext autoreload
         %autoreload 2
         %matplotlib inline
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```

```
In [30]: import xlrd
         import numpy as np
         import pandas as pd
         import os
         import re
         from progressbar import *
         import nibabel as nib
         import pdb
         import scipy.ndimage
         import matplotlib.pyplot as plt
         from my_tools import *
         from preprocess import *
         from PIL import Image
```

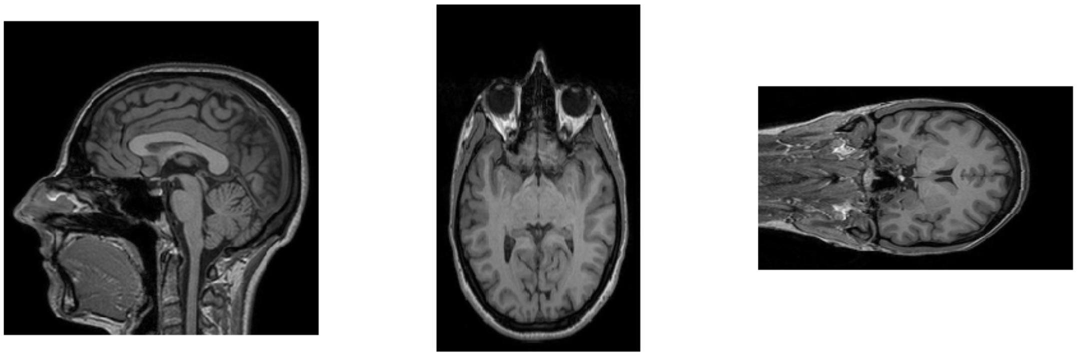
0.1 All images are resampled according to the pixel dimension information read from the image header files.

0.2 This ensures that all images will have the same resolution.

```
In [31]: nii_img = nib.load('/media/woody/Elements/age_data/IXI/IXI-T1/IXI002.nii.gz')
         header = nii_img.header
         print('header info from IXI002.nii.gz:')
         print_sep()
         print(header)
         print_sep()
         pixdim = header['pixdim'][1:4]
         print('\033[1;31mpixdim = {}\033[0m'.format(pixdim))
         npy_img = nii_img.get_data()
         print_sep()
         print('How does IXI002.nii.gz look like:')
         print2d(npy_img)
```

```
header info from IXI002.nii.gz:
---------------------------------------- - ----------------------------------------
<class 'nibabel.nifti1.Nifti1Header'> object, endian='<'
sizeof_hdr      : 348
data_type       : b''
db_name         : b''
extents         : 0
session_error   : 0
regular         : b'r'
dim_info        : 0
dim             : [  3 256 256 150   1   1   1   1]
intent_p1       : 0.0
intent_p2       : 0.0
intent_p3       : 0.0
intent_code     : none
datatype        : int16
bitpix          : 16
slice_start     : 0
pixdim          : [-1.        0.9375    0.9375    1.199997  0.        0.        0.
  0.      ]
vox_offset      : 0.0
scl_slope       : nan
scl_inter       : nan
slice_end       : 0
slice_code      : unknown
xyzt_units      : 10
cal_max         : 0.0
cal_min         : 0.0
slice_duration  : 0.0
toffset         : 0.0
glmax           : 0
glmin           : 0
descrip         : b'MR'
aux_file        : b''
qform_code      : scanner
sform_code      : scanner
quatern_b       : 0.468175
quatern_c       : -0.5299171
quatern_d       : -0.468175
qoffset_x       : -88.63989
qoffset_y       : 116.532005
qoffset_z       : -112.113556
srow_x          : [  0.         0.         1.199997 -88.63989 ]
srow_y          : [-9.30352330e-01  1.15545668e-01  0.00000000e+00  1.16532005e+02]
srow_z          : [ 1.15545668e-01  9.30352330e-01 -2.49799545e-16 -1.12113556e+02]
intent_name     : b''
magic           : b'n+1'
---------------------------------------- - ----------------------------------------
```

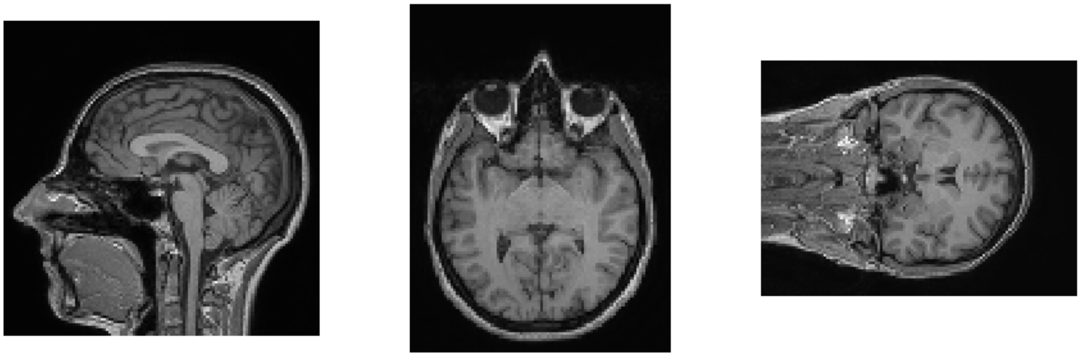
```
pixdim = [0.9375   0.9375   1.199997]
---------------------------------------- - ----------------------------------------
How does IXI002.nii.gz look like:
Dimension:  (256, 256, 150)
```



0.3 After the resampling

```
In [32]: resampled_img = resample(npy_img, pixdim, [2,2,2])
         print('How does the resampled image look like:')
         print2d(resampled_img)
```

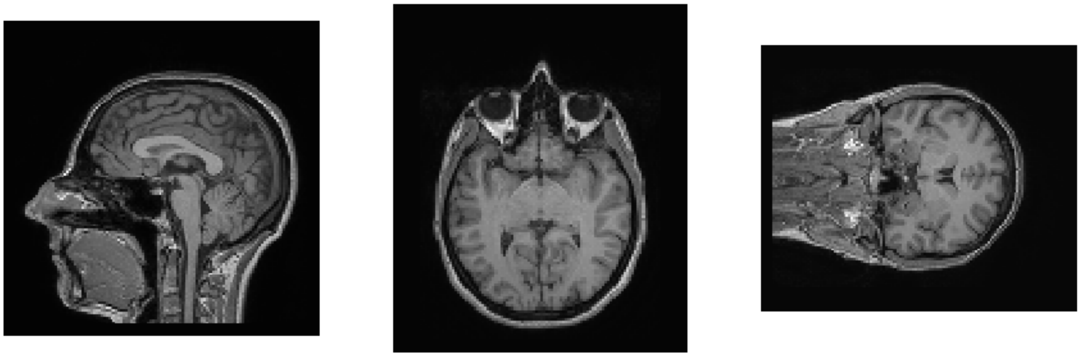
```
How does the resampled image look like:
Dimension:  (120, 120, 90)
```



0.4 The images are cropped and padded to the same shape (130,130,110)

```
In [33]: crop_padded_img = np.load('./IXI_npy/origin/IXI002.npy')
         print('How does the cropped image look like:')
         print2d(crop_padded_img)
```

```
How does the cropped image look like:
Dimension:  (130, 130, 110)
```



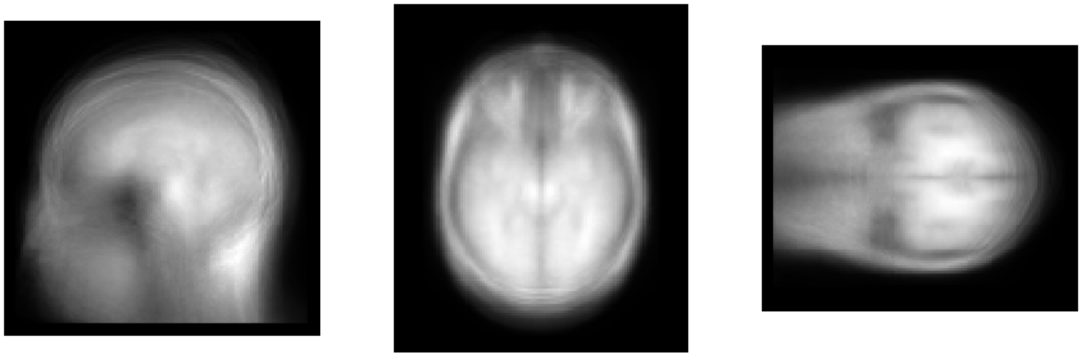
0.5 The mean image of all the training data is computed and is subtracted from all training and test data.

0.5.1 It is worth noting that the test data does not contribute to the mean image.

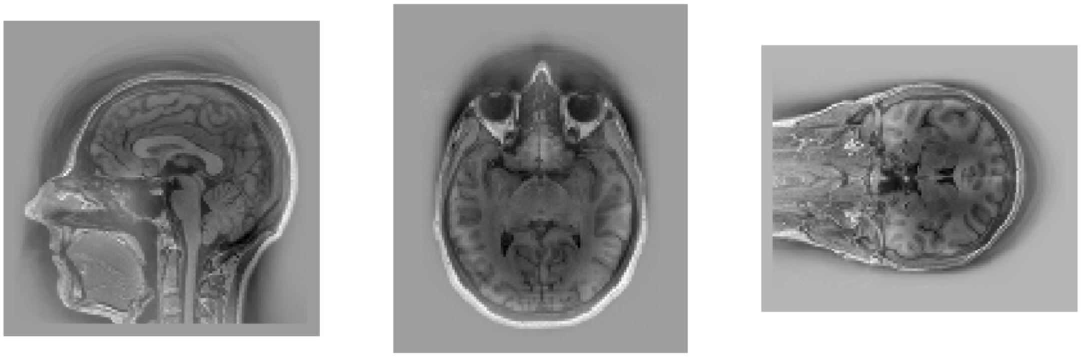
0.5.2 This is because the training data, and only training data, needs to have zero mean for better training performance.

```
In [34]: print('How does the mean values look like:')
         mean_npy = np.load('./IXI_npy/mean_npy.npy')
         print2d(mean_npy)
```

```
How does the mean values look like:
Dimension:  (130, 130, 110)
```

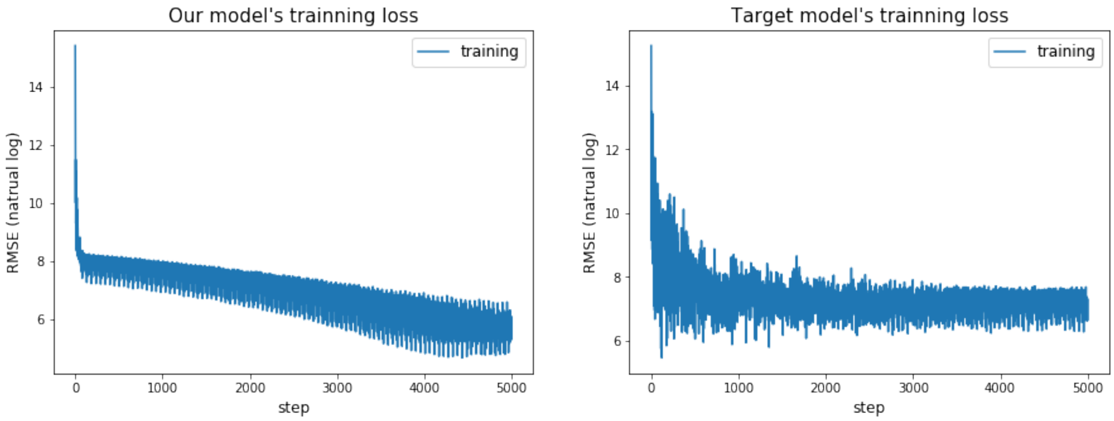


```
In [35]: print('How does the mean values subtracted image look like:')
         final_img = np.load('./IXI_npy/mean_subtracted/IXI002.npy')
         print2d(final_img)

How does the mean values subtracted image look like:
Dimension:  (130, 130, 110)
```



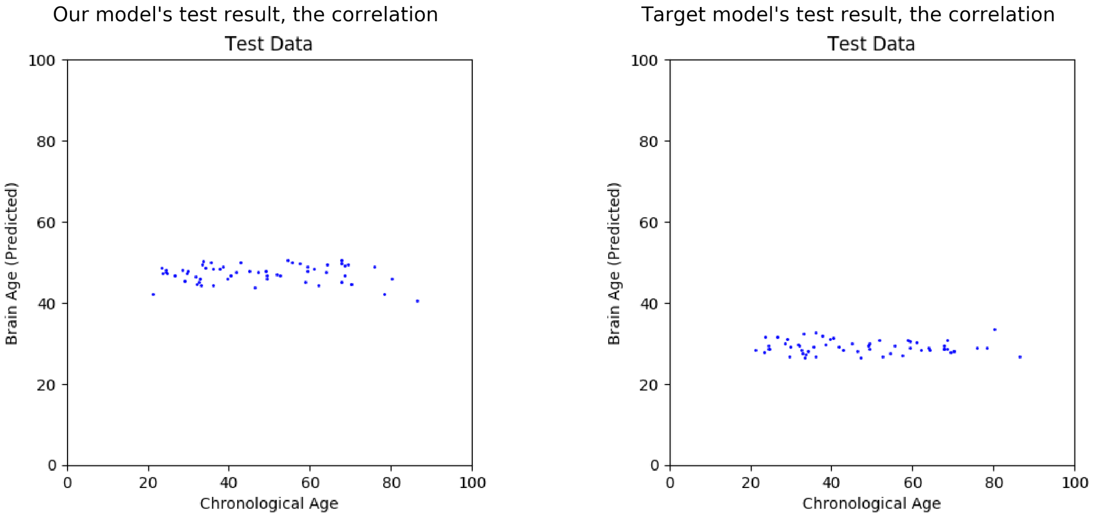
In Demo1.x we just used IXI dataset and split them into training and test sets. 10% for test, 90% for training. Unfortunately, it seems that IXI doesn't endure a strong correlation between the chronological and predicted age. But we have got the model ready for further use.

```
In [42]: def subdraw(ax,filename,max_step=10000):
         #     pdb.set_trace()
             arr = np.load(filename)
             steps = list(range(len(arr[0])))
             ax.plot(steps[:max_step], np.log(arr[0])[:max_step],label='training')
         #     steps2 = list(range(len(arr[1])))
         #     ax.plot(steps2[:max_step]*100, np.log(arr[1])[:max_step],label='validation')
             ax.legend(fontsize=12)
             ax.set_xlabel('step',fontsize=12)
             ax.set_ylabel('RMSE (natrual log)',fontsize=12)

         f, (ax1, ax2) = plt.subplots(1, 2, figsize=(15,5))
         subdraw(ax=ax1,filename='./img/demo_1_1_pltdata_2019.03.08.15:53:19.npy',max_step=5000)
         ax1.set_title('Our model\'s trainning loss',fontsize=15)
         subdraw(ax=ax2,filename='./img/demo_1_2_pltdata_2019.03.08.17:18:47.npy',max_step=5000)
         ax2.set_title('Target model\'s trainning loss',fontsize=15)

Out[42]: Text(0.5, 1.0, "Target model's trainning loss")
```



```
In [43]: f, (ax1, ax2) = plt.subplots(1, 2, figsize=(40,20))
         ax1.imshow(plt.imread('./img/demo_1_1_test.png'))
         ax1.axis('off')
         ax1.set_title('Our model\'s test result, the correlation',fontsize=40)
         ax2.imshow(plt.imread('./img/demo_1_2_test.png'))
         ax2.axis('off')
         ax2.set_title('Target model\'s test result, the correlation',fontsize=40)
```

```
Out[43]: Text(0.5, 1.0, "Target model's test result, the correlation")
```



```
In [44]: printimg('/home/woody/Pictures/d3.png')
         printimg('/home/woody/Pictures/datainfo.png')
         printimg('/home/woody/Pictures/dataset.png')
         printimg('/home/woody/Pictures/d4.png')
```

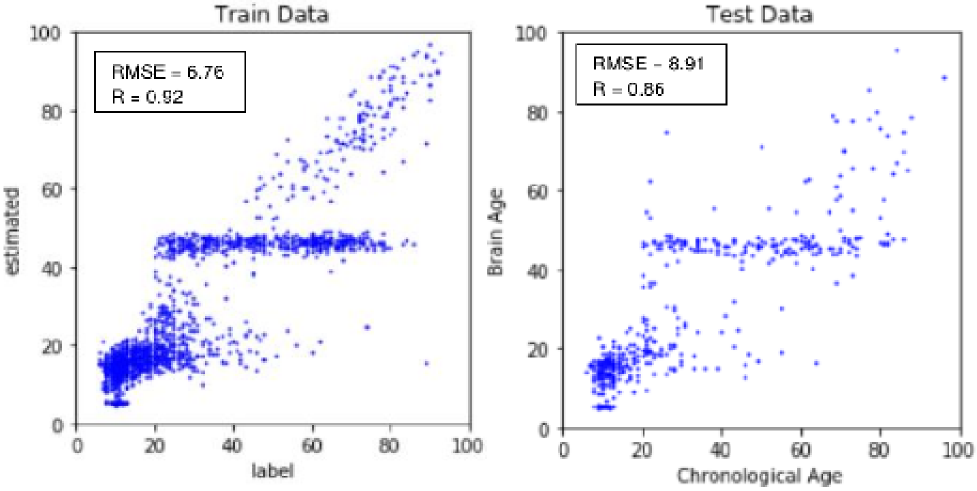
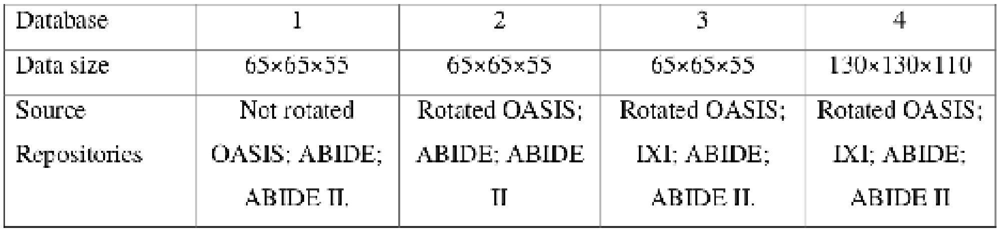


Figure 3.17 Performance of the model trained by dataset 3

Name	Size (N)	Age RANGE	Age Average
OASIS	316	18-96	52.4
ABIDE	574	6-56	17.0
IXI	590	20-86	49.4
ABIDEII	593	6-64	14.9
Total	2072	6-96	31.0

Table 2.13 Raw data information



Database	1	2	3	4
Data size	65×65×55	65×65×55	65×65×55	130×130×110
Source Repositories	Not rotated OASIS; ABIDE; ABIDE II.	Rotated OASIS; ABIDE; ABIDE II	Rotated OASIS; IXI; ABIDE; ABIDE II.	Rotated OASIS; IXI; ABIDE; ABIDE II

Table 3.13 A summary of databases

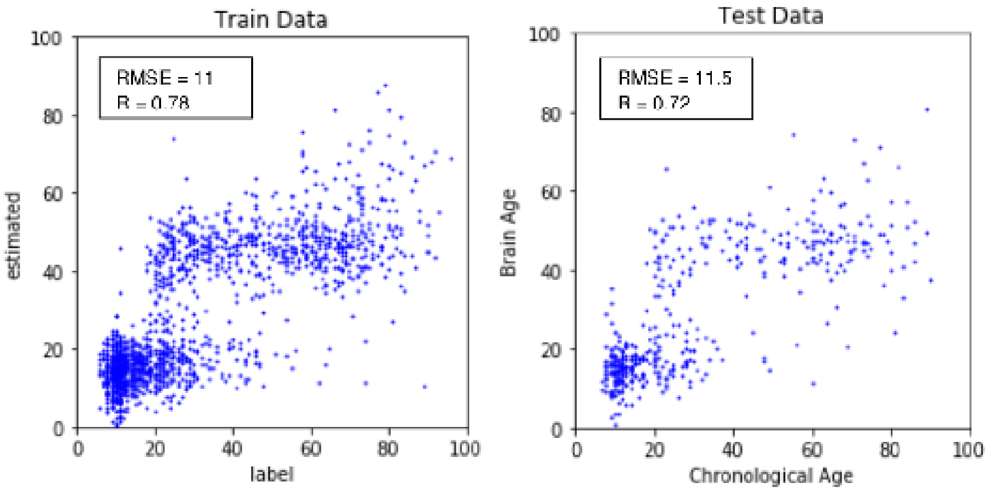


Figure 3.19 Performance of the model trained by dataset 4

```
In [ ]:
```