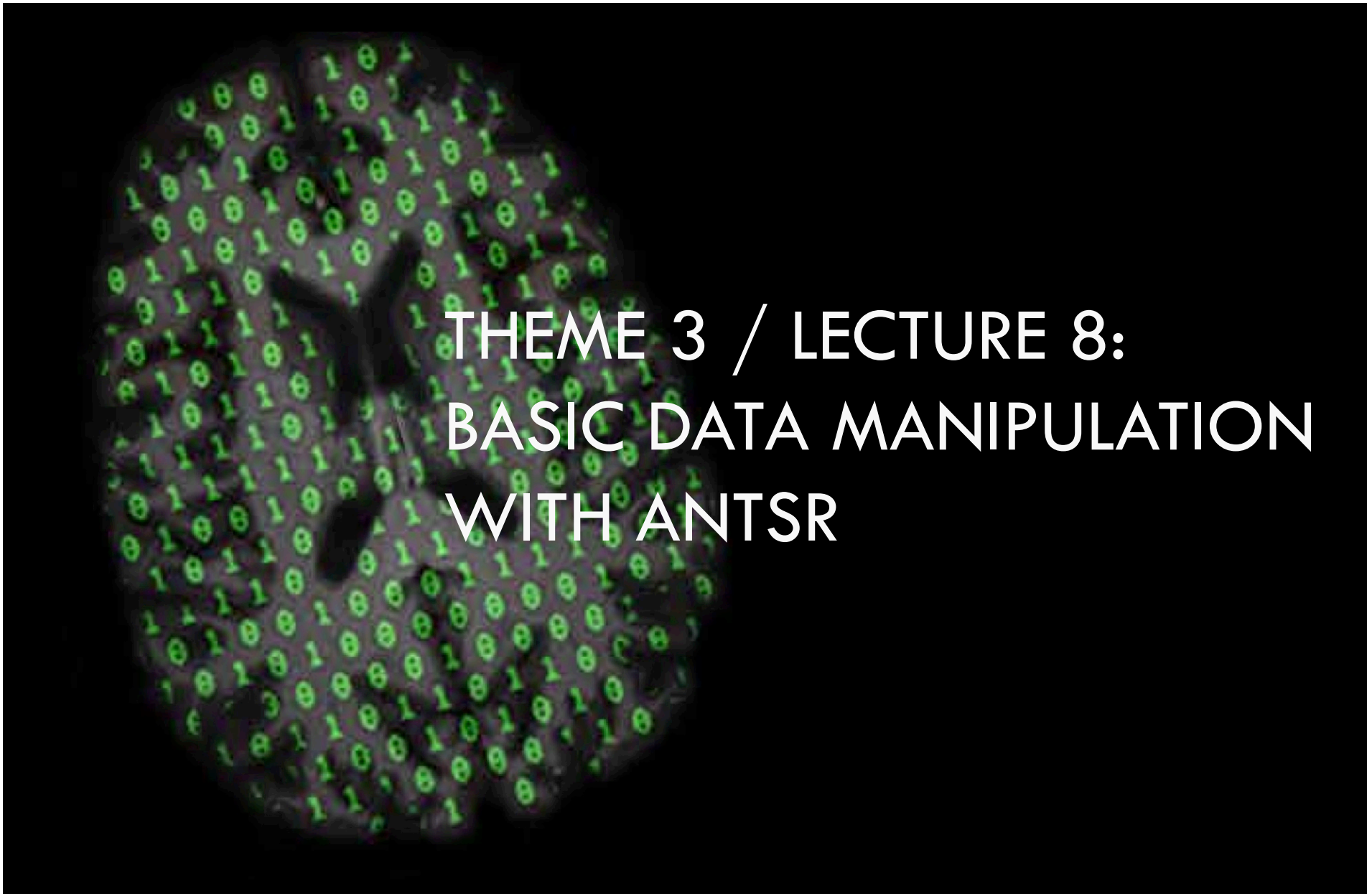
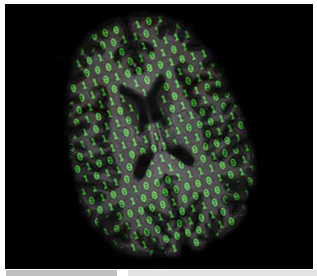
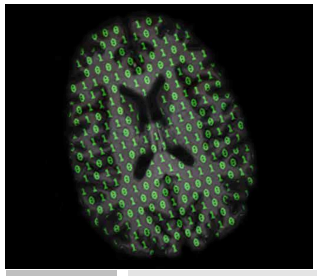


THEME 3 / LECTURE 8: BASIC DATA MANIPULATION WITH ANTSR



Basic Data Manipulation with ANTsR

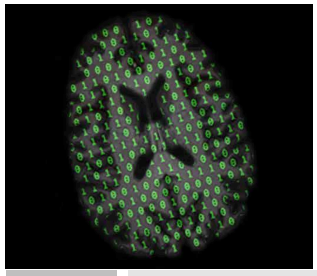
- Reading images
- ANTsR Images
- Basic Statistics
- ANTs image class



Reading Images Using ANTsR

- Requires 2 changes compared to readNIfTI from oro.nifti
  - The extension of the filename (e.g. .nii.gz) must be specified
  - The dimension of the image must be supplied (could be 2, 3, or 4)

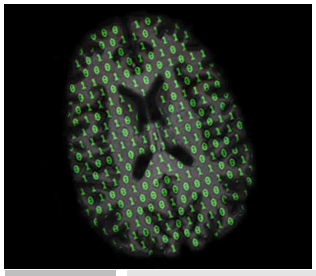
```
library(ANTsR)
aimg=antsImageRead("113-01-MPRAGE.nii.gz",dimension=3)
```



ANTsR Images

- The aimg object is an object of antsImage, which consists of:
  - pixeltype - how is the image stored (integers versus fractional numbers)
  - dimension - how many dimensions does the image have
  - pointer - where the data is stored

```
class(aimg)
[1] "antsImage"
attr(,"package")
[1] "ANTsR"
aimg
antsImage
Pixel Type : float
Pixel Size : 1
Dimensions : 512x512x22
Voxel Spacing: 0.46875x0.46875x5
Origin : 0 0 0
```



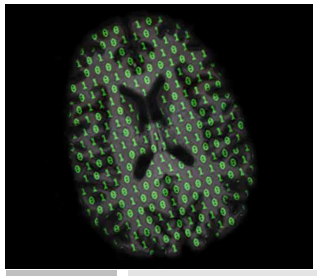
ANTsR Images: Statistics

- Basic data manipulations can be done on the `antsImage`

```
mean(aimg)
[1] 102.4701
mean(aimg[aimg!=0])
[1] 179.4116
```

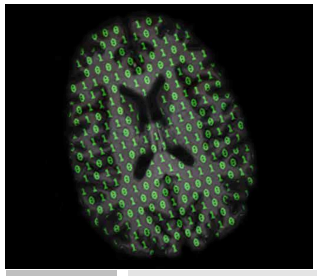
- The image can be obtained from `antsImage` using `as.array`

```
class(as.array(aimg))
[1] "array"
```



Why Discuss the `antsImage` Class?

- The class can be very fast at performing operations
- Some ANTsR functions return object of `antsImage` class
- Some ANTsR functions require an object of `antsImage` class as input



From `antsImage` to `nifti`

- The `extrantsr` (EXTRa ANTsR) package has helper functions to jump from ANTsR to the oro.nifti classes:
- Installing extrantsr:

```
devtools::install_github("muschellij2/extrantsr")
library(extrantsr)
class(nim <- ants2oro(aimg))
[1] "nifti"
attr(,"package")
[1] "oro.nifti"
```