

# {NEUROCONDUCTOR CHEAT SHEET}

JOHN MUSCHELLI  
(JOHNS HOPKINS UNIVERSITY)

## IMAGE OBJECTS:

<b>- Types:</b>		
	<ul style="list-style-type: none"><li>• <code>nifti</code> (<i>oro.nifti</i>) - 3D array with header information, data in memory</li><li>• <code>antsImage</code> (<i>ANTsR</i>) - C++ pointer, not in memory</li><li>• <code>niftiImage</code> (<i>RNifti</i>) - C++ pointer, not in memory</li></ul>	
<b>- Manipulation:</b>		

nifti objects

Comparison Operators	>, >=, <, <=, ==, !=	Logical image
Arithmetic	+, -, *, /	Numeric image
In operator	%in%	Logical vector

- Conversion:		
---------------	--	--

How to convert **to** nifti objects from:

Type	function	Description
antsImage	extrantsr::ants2oro	Writes out image, reads in as a nifti
antsImage	extrantsr::ants2oro(aimg, reference = img)	Uses the img nifti object as header, faster
niftiImage	oro.nifti::nii2oro(aimg)	Extracts aimg array, then copies header to nifti object
.....		

How to convert **from** nifti objects to:

Type	function	Description
antsImage	extrantsr::oro2ants	Writes out image, reads in as a antsImage
antsImage	extrantsr::oro2ants(aimg, reference = img)	Uses the img antsImage object as header, faster
niftiImage	oro.nifti::oro2nii(aimg)	Writes out image, reads in as a niftiImage

## IMAGE PLOTTING:

- Plotting:	
Function	Output
oro.nifti::orthographic	3-planar view of axial, sagittal, and coronal brain
oro.nifti::image	Prints slices of an image, make sure plot.type = "single" for only one slice
neurobase::ortho2	Similar to oro.nifti::orthographic, but different defaults, cross-hairs, and terms of location When ortho2(x, y) is called, zeros in y are set to NA, assuming y is an overlay
neurobase::double_ortho(x, y)	Two side-by-side orthographic views
neurobase::multi_overlay	Takes in a <b>list</b> ; plots one slice side-by-side from each image in the list Single slice of brain with red and blue overlaid on it

## LISTAS NO PYTHON

- Listas no Python	
Listas são compostas por elementos de qualquer tipo (podem ser alteradas)	

### Manipulação de Listas no Python

Criação	
uma_lista = [5,3,'p',9,'e']	cria: [5,3,'p',9,'e']
Acessando	
uma_lista[0]	retorna: 5
Fatiando	
uma_lista[1:3]	retorna: [3,'p']
Comprimento	
len(uma_lista)	retorna: 5
count( item)	
Retorna quantas vezes o item foi encontrado na lista.	
cont(uma_lista('p'))	retorna: 1
Pode ser usado juntamente com a função while para 'andar' pelo comprimento da lista:	
while x < len(uma_lista):	retorna: [3,'p']
Ordenar - sort()	
uma_lista.sort()	retorna: [3,5,9,'e','p']
Ordenar sem alterar a lista	
print(sorted(uma_lista))	retorna: [3,5,9,'e','p']
Adicionar - append(item)	
uma_lista.append(37)	retorna: [5,3,'p',9,'e',37]
Inserir - insert(position,item)	
insert(uma_lista.append(3),200)	retorna: [5,3,200,'p',9,'e']
Retornar e remover - pop(position)	
uma_lista.pop()	retorna: 'e' e a lista fica [5,3,'p',9]
uma_lista.pop(1)	retorna: 3 e a lista fica [5,'p',9,'e']
Remover - remove(item)	
uma_lista.remove('p')	retorna: [5,3,9,'e']
Inserir	
uma_lista.insert(2,'z')	retorna: [5,'z',3,'p',9,'e']
Inverter - reverse()	
reverse(uma_lista)	retorna: ['e',9,'p',3,5]
Concatenar	
uma_lista+[0]	retorna: [5,3,'p',9,'e',0]
uma_lista+uma_lista	retorna: [5,3,'p',9,'e',5,3,'p',9,'e']
Encontrar	
9 in uma_lista	retorna: True
for x in uma_lista	retorna toda a lista, um elemento por linha
.....print(x)	

# {PYTHON CHEAT SHEET}

MICHELLE CRISTINA DE SOUSA BALTAZAR  
(UNIVERSIDADE FEDERAL DO TRIÂNGULO MINEIRO)