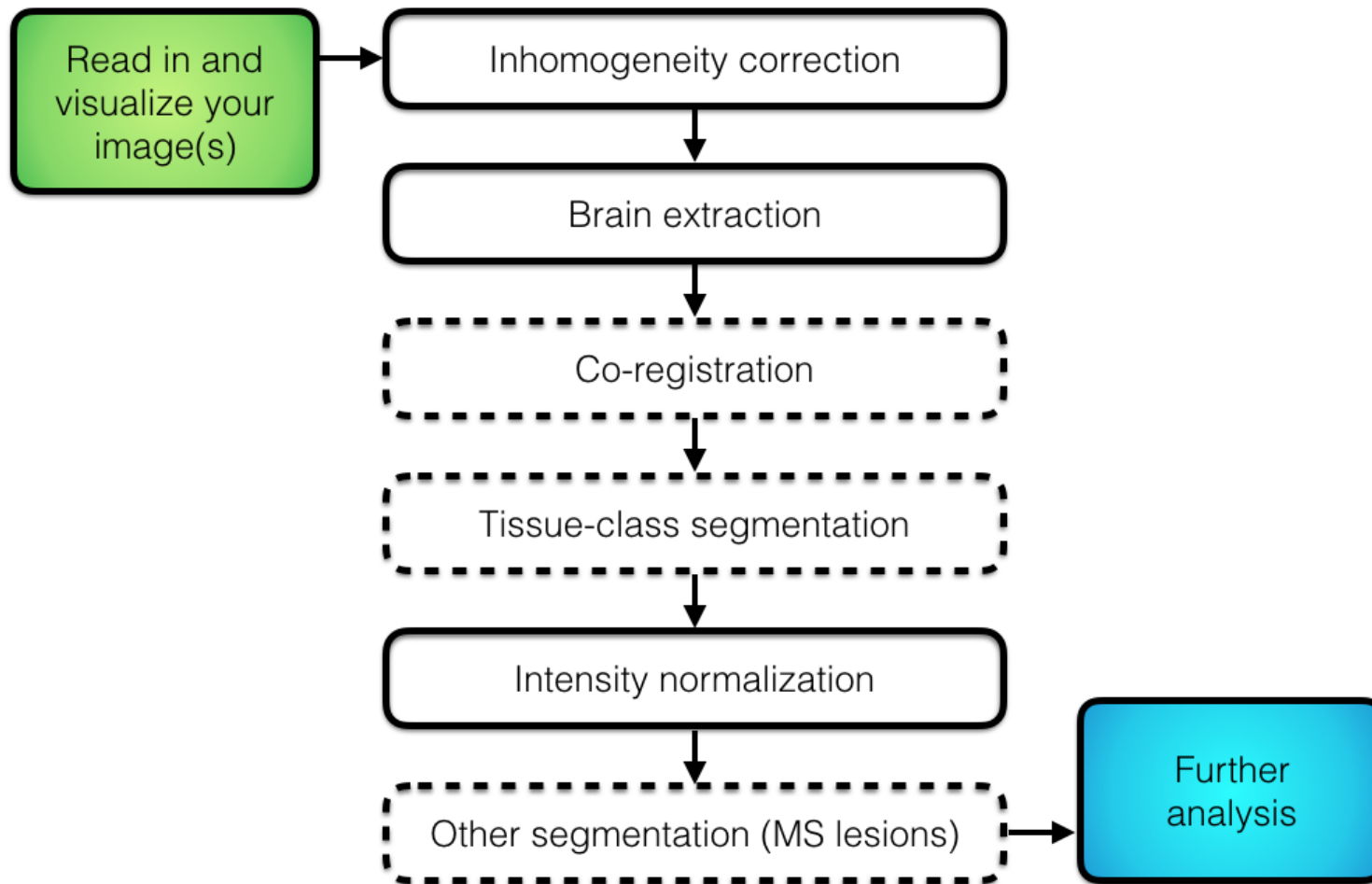# Inhomogeneity Correction

MS Lesion

Let's reset and read in the T1 image from a MS lesion data set:

```
t1 = neurobase::readnii("training01_01_t1.nii.gz")
t1[ t1 < 0 ] = 0
```

Processing math: 100%

Overall Pipeline

Processing math: 100%

Inhomogeneity correction

- Scans can have nonuniform intensities throughout the brain

- Usually low frequency - smooth over the brain (assumed)
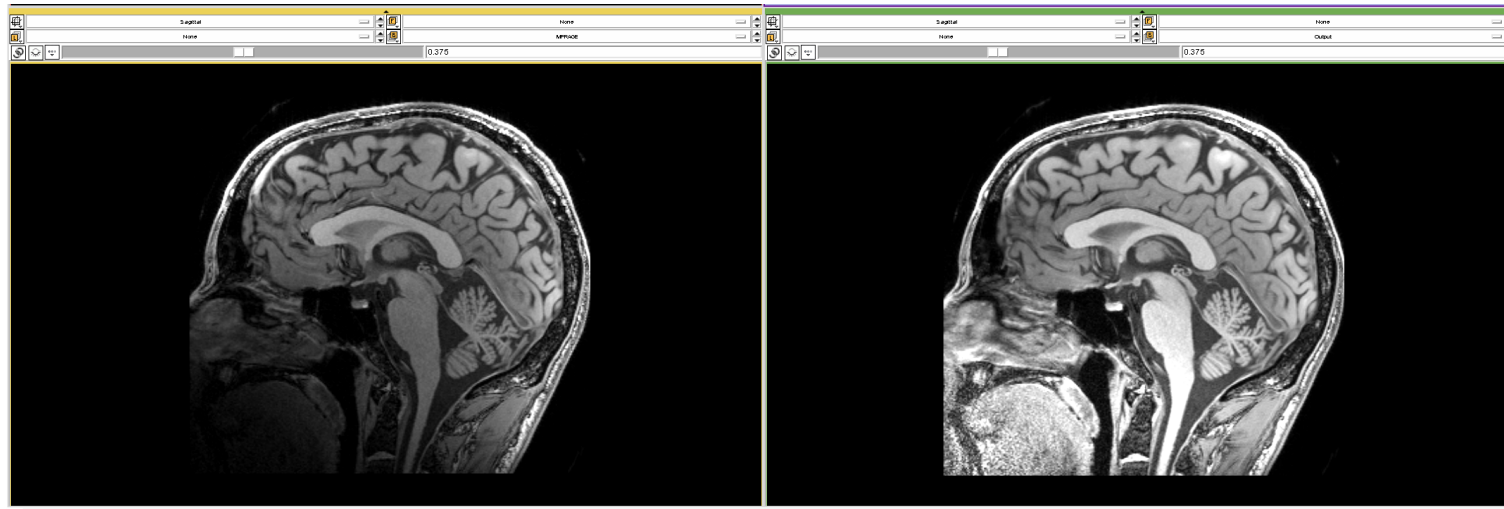
- Referred to as bias, bias field, or inhomogeneity

Image From
https://www.slicer.org/w/images/7/77/MRI_Bias_Field_Correction_Slicer3_close_up.png

Processing math: 100%

Image Data

It's hard to see subtler bias fields, but sometimes they can be seen visually.

```
ortho2(robust_window(t1))
```

Processing math: 100%
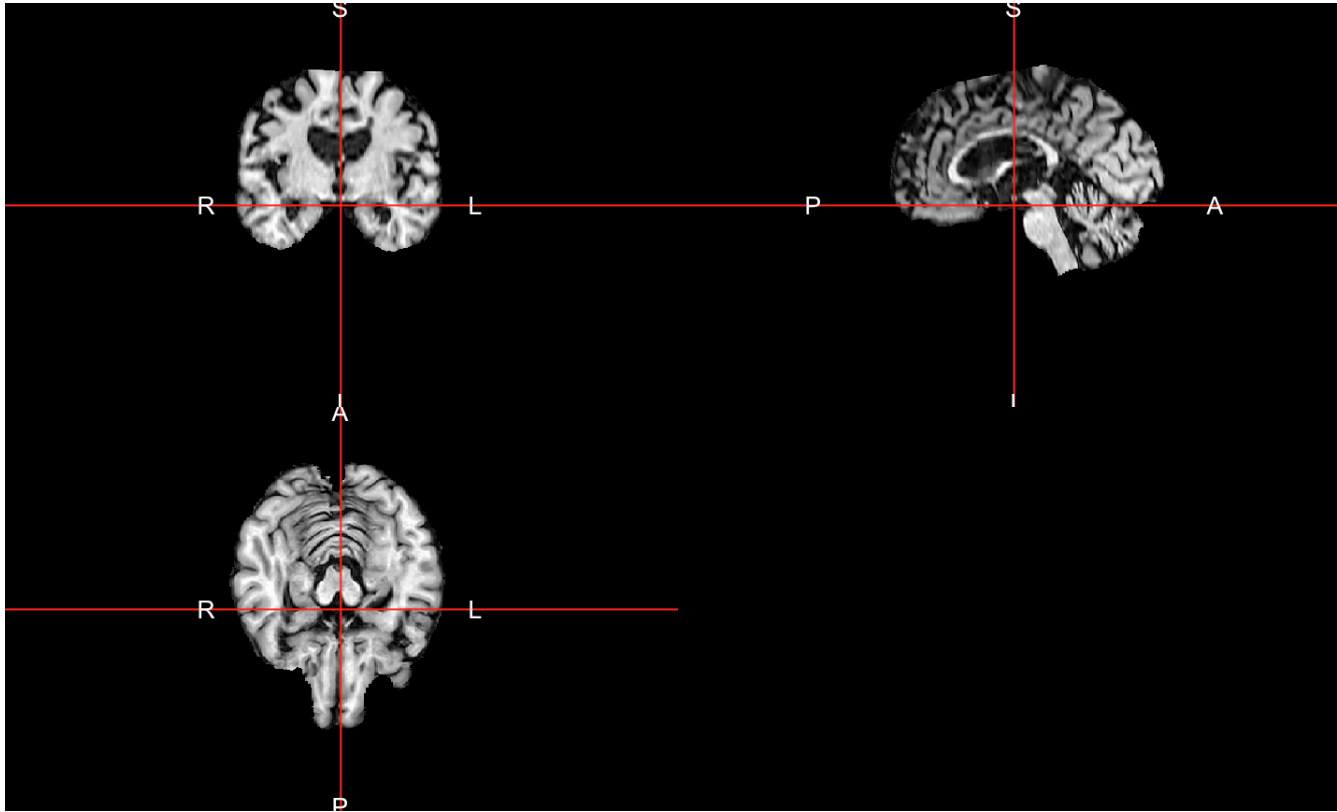
Image Data

```
flair = neurobase::readnii("training01_01_flair.nii.gz")
ortho2(robust_window(flair))
```

Processing math: 100%
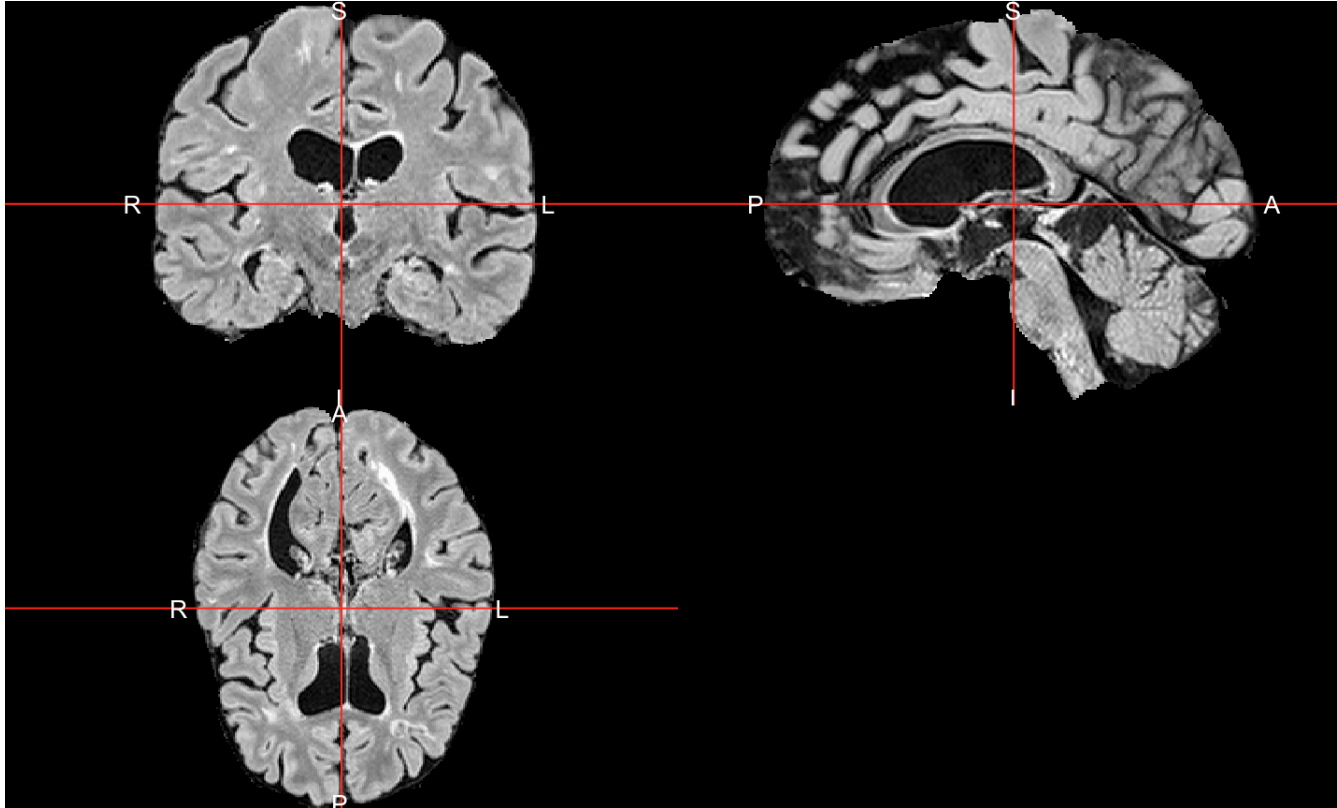
```r
image(robust_window(t1), useRaster = TRUE)
```

Processing math: 100%
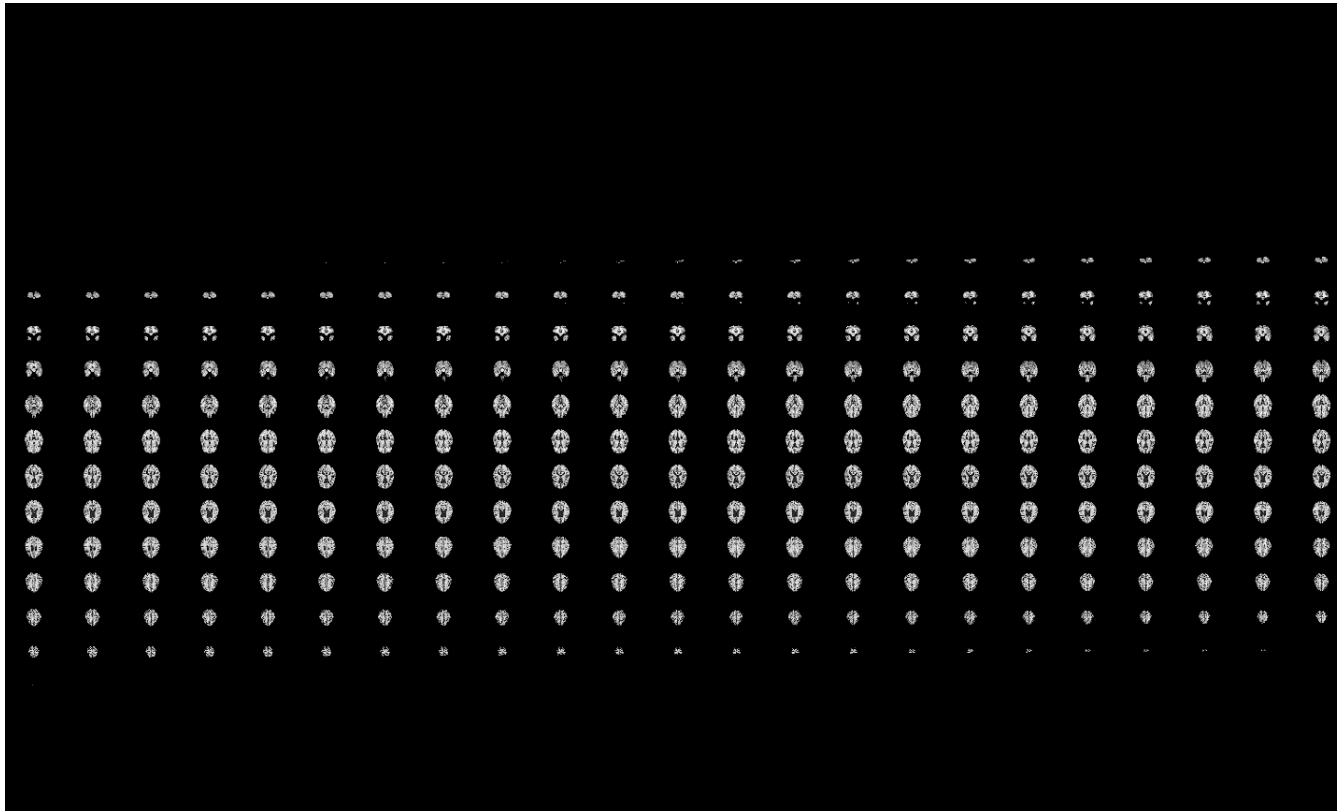
N4 Inhomogeneity Correction

We will use N4: Improved N3 Bias Correction (Tustison et al. 2010).

The model assumed in the N4 is: $v(x) = u(x)f(x) + n(x)$

- v is the given image
- u is the uncorrupted image
- f is the bias field
- n is the noise (assumed to be independent and Gaussian)
- x is a location in the image

Processing math: 100%

N4 Inhomogeneity Correction

The data is log-transformed and assuming a noise-free scenario, we have:

$\log(v(x)) = \log(u(x)) + \log(f(x))$

- N4 uses a B-spline approximation of the bias field
- It iterates until a convergence criteria is met
  - when the updated bias field is the same as the last iteration
- It outputs the data back in the original units (not log-transformed)

Processing math: 100%

Bias Field Correction

Here we will use the `bias_correct` function in `extrantsr`, which calls `n4BiasFieldCorrection` from `ANTsR`.

You can pass in the image:

```
library(extrantsr)
bc_t1 = bias_correct(file = t1, correction = "N4")
```
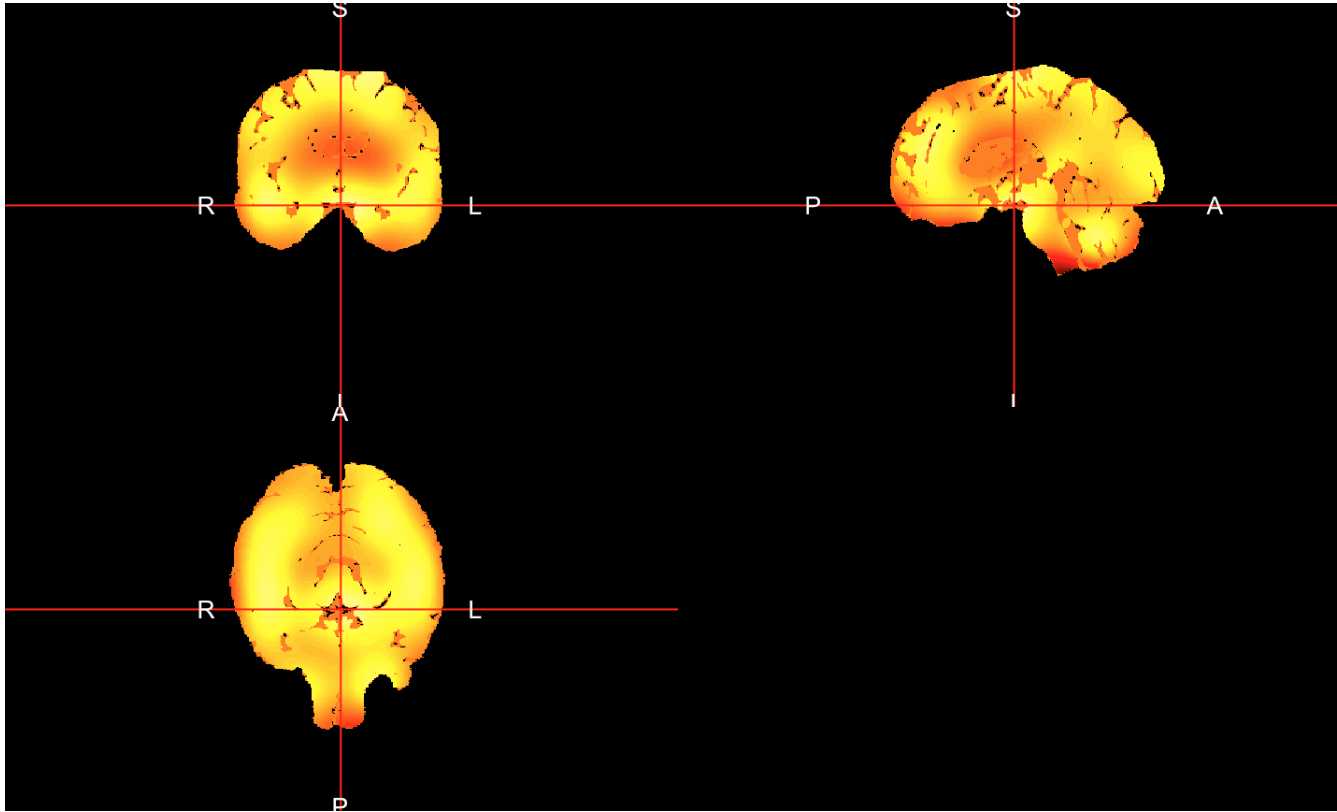
or the filename (but negatives are in there):

```
bc_t1 = bias_correct(file = "training01_01_t1.nii.gz", correction = "N4")
```

Processing math: 100%

Visualizing Bias Field Correction

Here we take the ratio of the images and overlay it on the original image:

```
ratio = t1 / bc_t1; ortho2(t1, ratio)
```

Processing math: 100%

Visualizing Bias Field Correction

We are breaking the ratio into quantiles:

```
ortho2(t1, ratio, col.y = colors, ybreaks = quantiles, ycolorbar = TRUE)
```

Processing math: 100%
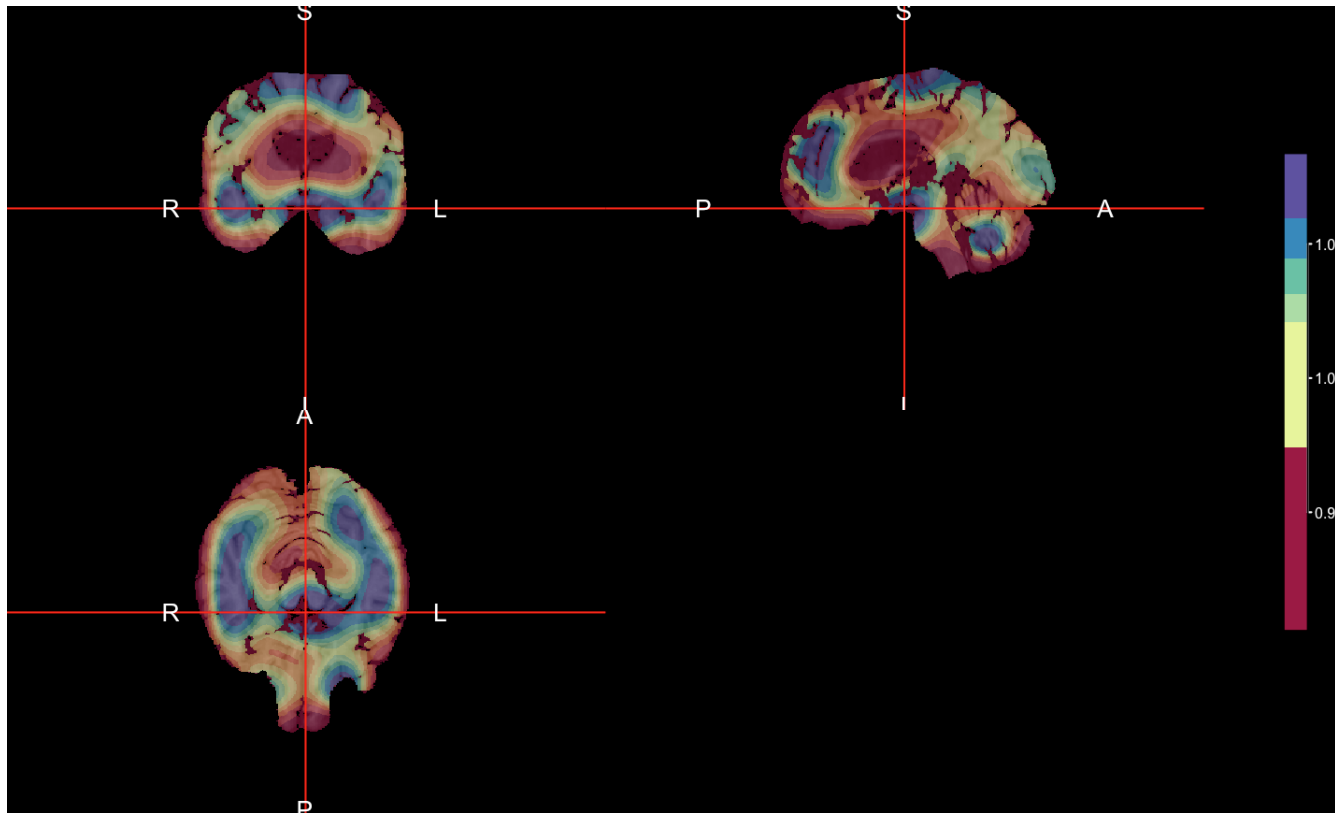
Conclusions

- Inhomogeneity correction is one of the first steps of most structural MRI pipelines
- Inhomogeneity can cause problems for other methods/segmentation
- Corrections try to make tissues of the same class to have similar intensities
- Use the `extrantsr bias_correct` function
    - There is also `fsl_biascorrect` from `fslr` (not as effective in our experience)
- You may also want to run corrections after skull stripping on the brain only
    - this is possible with the result after the brain extraction lecture
    - correction before skull-stripping may be necessary and can improve after correction

Processing math: 100%

Website

http://johnmuschelli.com/imaging_in_r

Processing math: 100%

References

Tustison, Nicholas J., Brian B. Avants, Philip A. Cook, Yuanjie Zheng, Alexander Egan, Paul A. Yushkevich, and James C. Gee. 2010. "N4ITK: Improved N3 Bias Correction." 29 (6):1310–20. https://doi.org/10.1109/TMI.2010.2046908.

Processing math: 100%