

Neuroimaging-Project-230

2023-04-19

Load libraries

```
library(oro.nifti)

## oro.nifti 0.11.4
library(neurobase)
library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.1     v readr     2.1.4
## vforcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.1     v tibble    3.2.1
## v lubridate 1.9.2     v tidyrr    1.3.0
## v purrr     1.0.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
## x dplyr::slice() masks oro.nifti::slice()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(caret)

## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
## 
##     lift

library(ggplot2)
library(gridExtra)

##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
## 
##     combine

library(caret)
source("~/scripts/230/utils.R")
```

Creating list of white matter and gray matter paths

```
# image directory
img_dir <- 'preprocessed/imgsss'

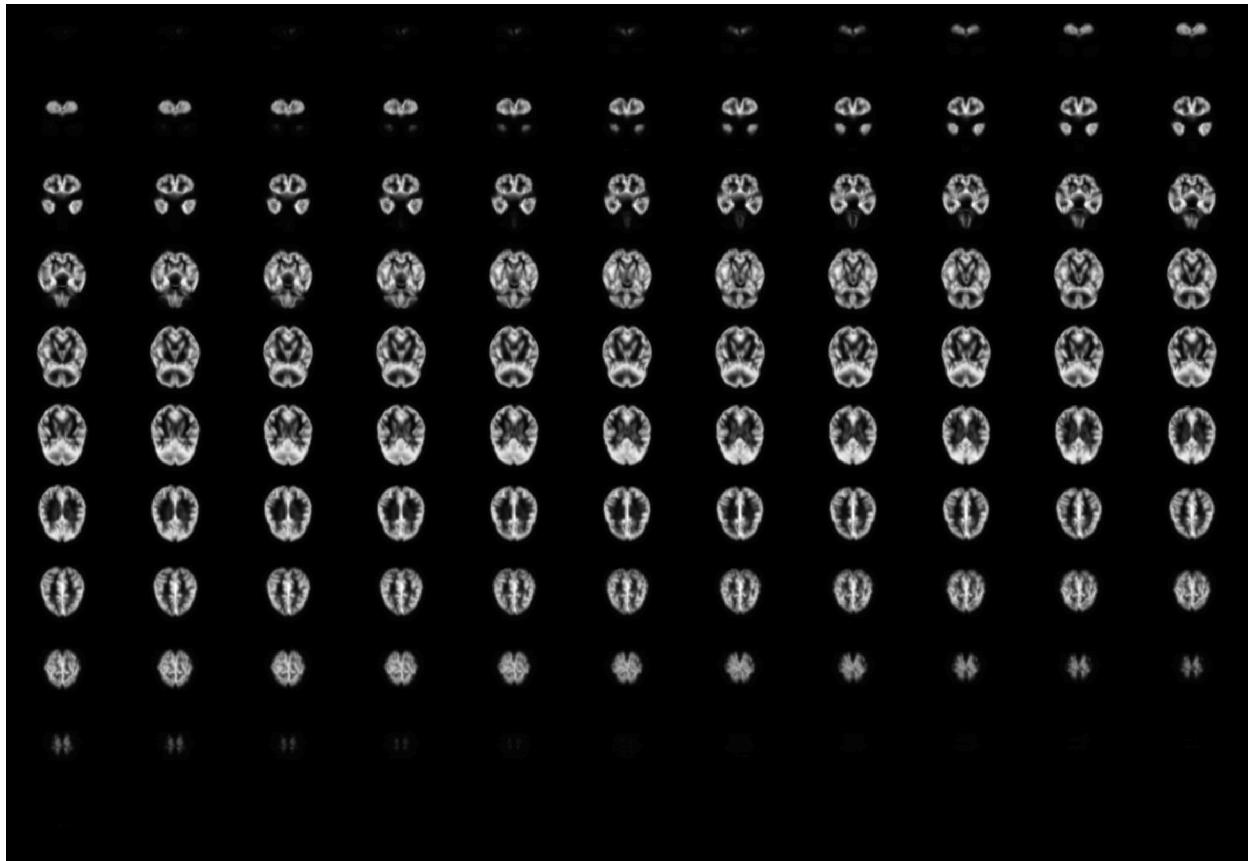
# creating full list of gm and wm files
all_MRI_files <- get_ordered_files(img_dir, 'smt')

# splitting into wm and gm
gm_files <- all_MRI_files[[1]]
wm_files <- all_MRI_files[[2]]
```

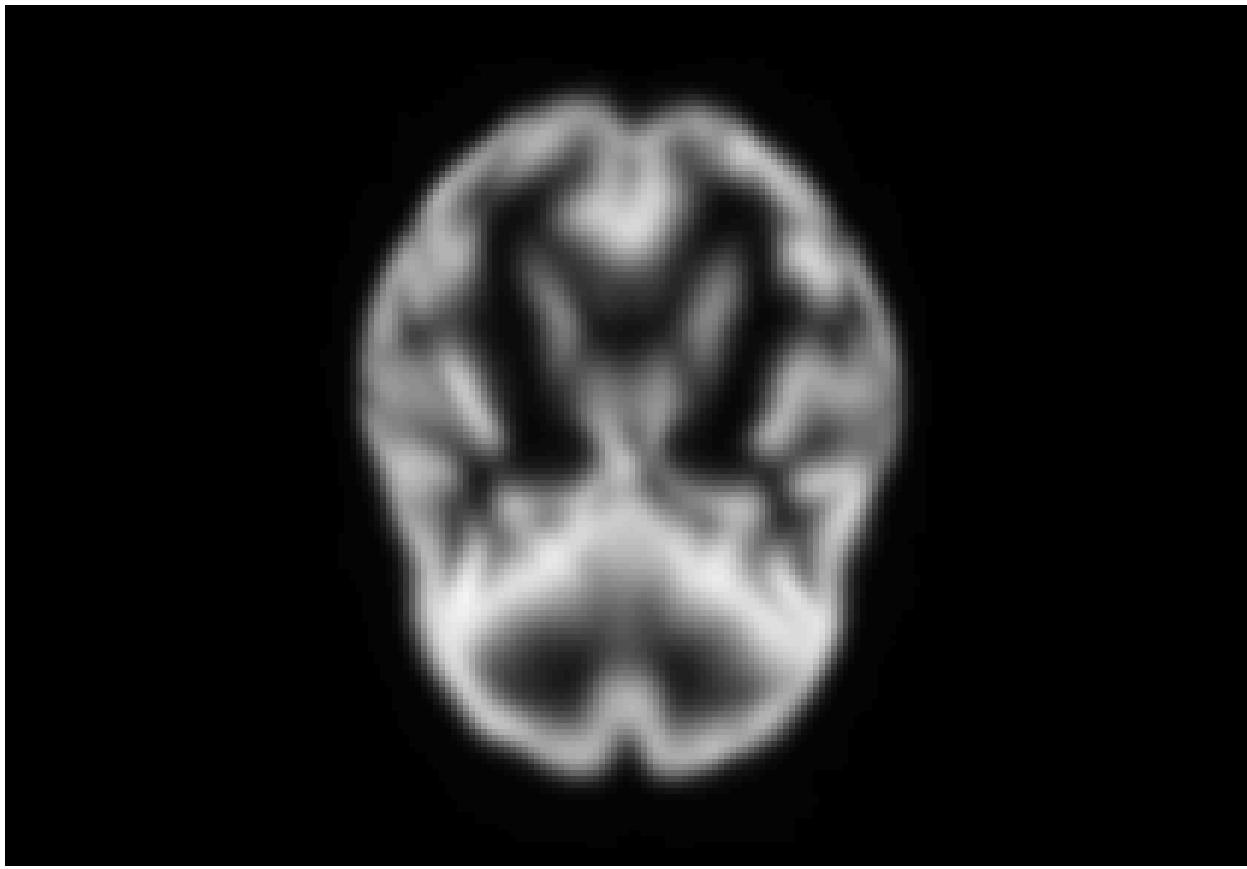
Visualizing the MRI images

```
# load all Nifti object images
nifti_images <- lapply(gm_files[1:2], readNIIfTI)

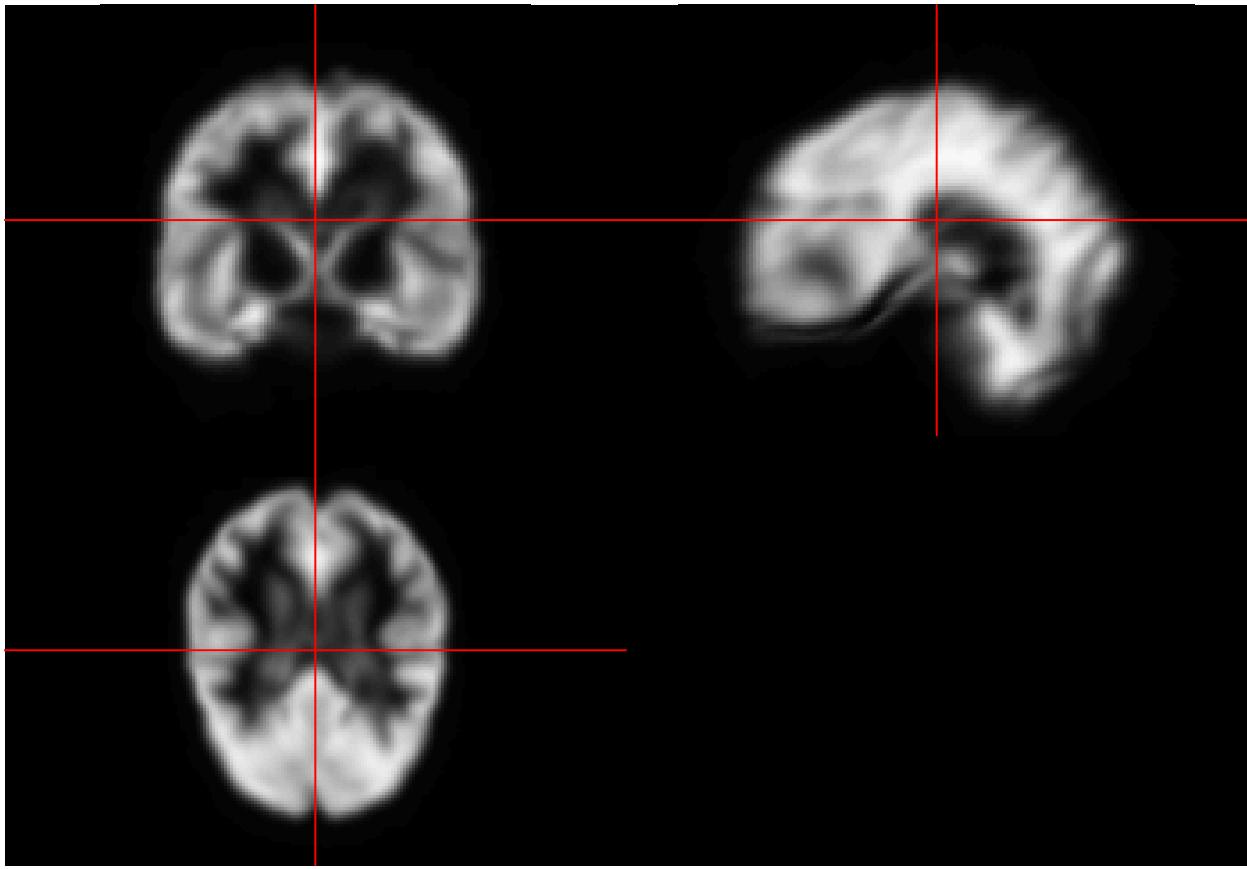
# all slices from the axial plane of 1 image
oro.nifti::image(x=nifti_images[[1]], plane=c('axial'), plot.type=c("multiple"))
```



```
# slice 52 of axial plane
oro.nifti::image(x=nifti_images[[2]], z=52, plane=c('axial'), plot.type=c("single"))
```



```
# all 3 planes of the MRI image spatially aligned  
ortho2(nifti_images[[1]])
```



Reading in and cleaning metadata

```
# read in metadata
metadata_fname <- "ADNI1_Complete_2Yr_3T_4_18_2023.csv"
md <- read_csv(metadata_fname, show_col_types = FALSE)

# select usable features and make into dummie vars
md <- md %>%
  select(`Image Data ID`, Subject, Group, Sex, Age) %>%
  rename(Image_ID = `Image Data ID`) %>%
  mutate(Sex = ifelse(Sex=='M', 0, 1),
        Group = case_when(
          Group == 'CN' ~ 0,
          Group == 'MCI' ~ 1,
          Group == 'AD' ~ 2,
          TRUE ~ NA_integer_
        ))
head(md)

## # A tibble: 6 x 5
##   Image_ID Subject    Group  Sex  Age
##   <chr>     <chr>     <dbl> <dbl> <dbl>
## 1 I205567  136_S_1227     1     1    66
## 2 I66824   136_S_1227     1     1    65
## 3 I79080   136_S_1227     1     1    65
```

```

## 4 I143856 136_S_1227      1      1    67
## 5 I99265  136_S_1227      1      1    66
## 6 I92097  136_S_0579      1      1    67

```

Creating the design matrix

```

X_wm <- create_design_mat(md, file_ls=wm_files[1:5])
save(X_wm, file = file.path(root_dir, "R_data", "X_wm.Rda"))

X_wm[1:5,1:20]

##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20
## 1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 3  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 4  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
## 5  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0

dim(X_wm)

## [1]      5 2122948

# X_gm <- create_design_mat(md, file_ls=gm_files)
# save(X_gm, file = file.path(root_dir, "R_data", "X_gm.Rda"))

# X_cb <- create_design_mat(md, wm_files, wm_files, gm_files, comb=TRUE)
# save(X_cb, file = file.path(root_dir, "R_data", "X_cb.Rda"))

```