

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(ggplot2)
library(gridExtra)

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

library(caret)

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

library(knitr)
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]]

gm_files[1:3]

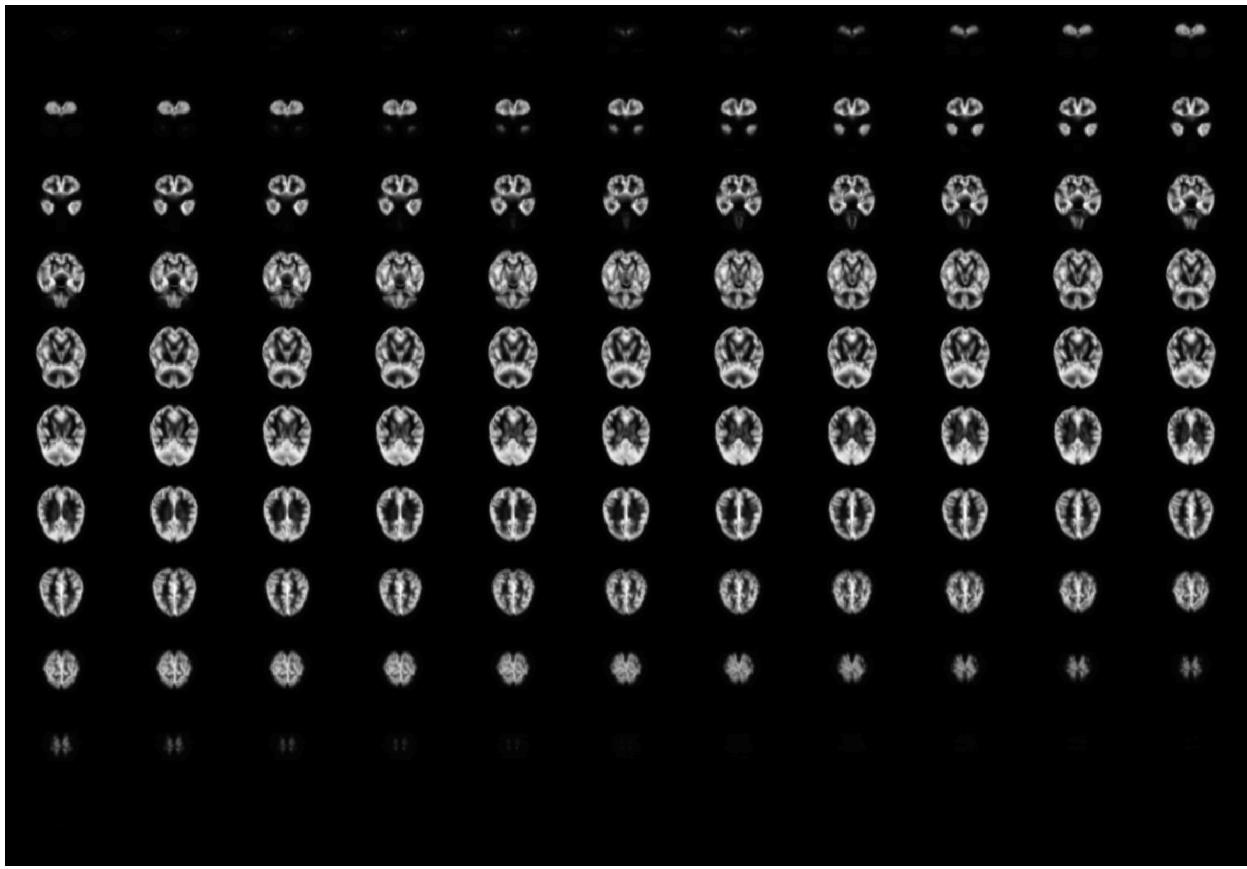
## [1] "preprocessed/imgsss/smt-002_S_0413-I118675-gm.nii.gz"
## [2] "preprocessed/imgsss/smt-002_S_0413-I120746-gm.nii.gz"
## [3] "preprocessed/imgsss/smt-002_S_0413-I128346-gm.nii.gz"
```

Visualizing the MRI images

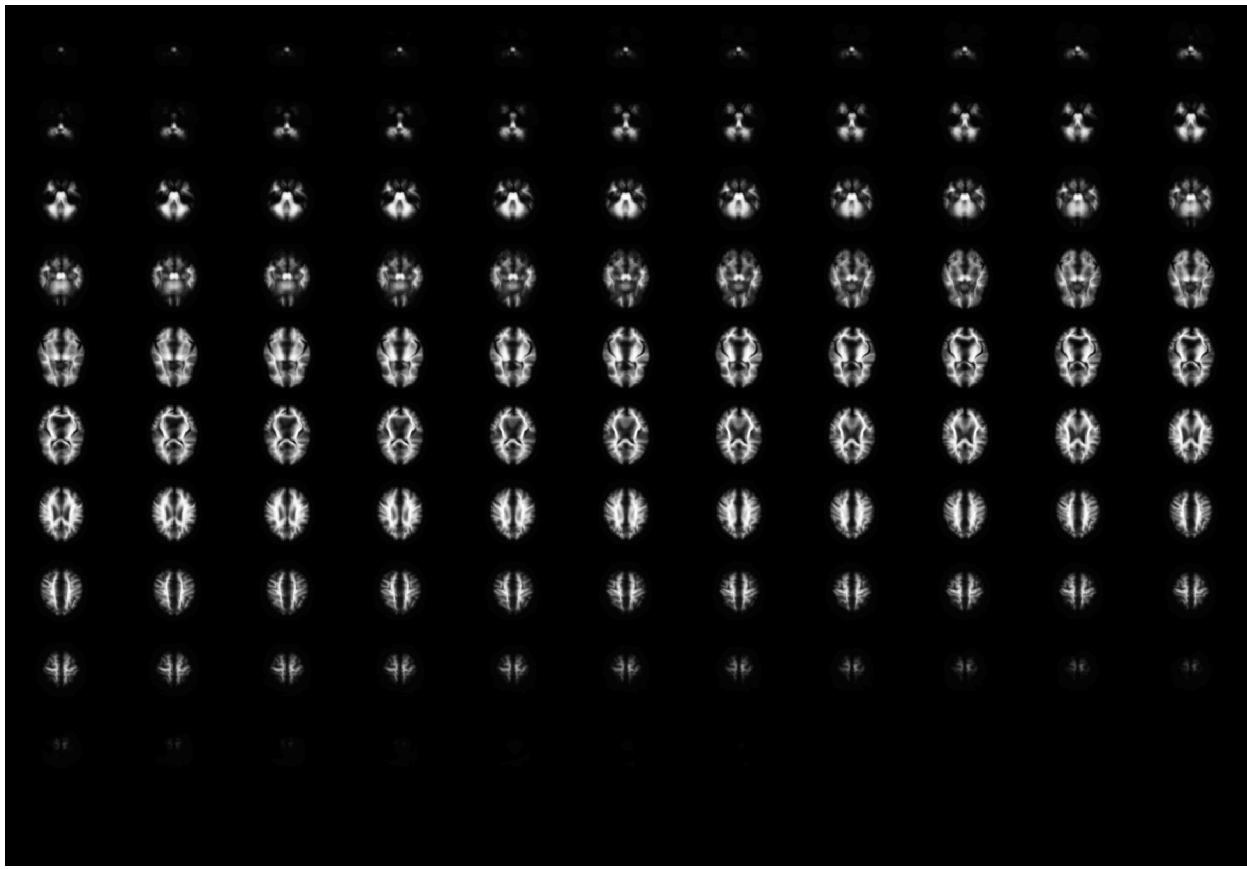
```
# load all Nifti object images
nifti_images_gm <- lapply(gm_files[1:2], readNIIfTI)

nifti_images_wm <- lapply(wm_files[1:2], readNIIfTI)

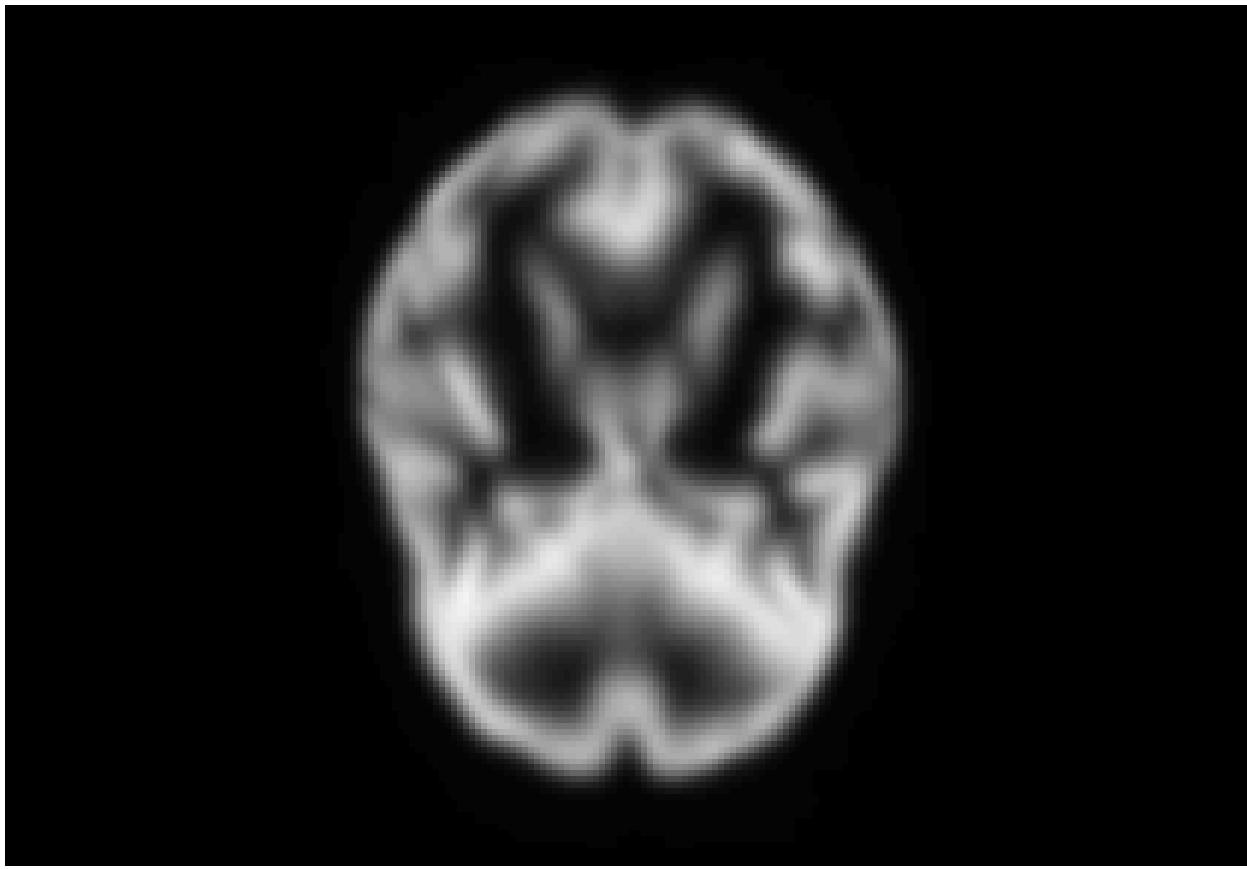
# all slices from the axial plane of 1 image
oro.nifti::image(x=nifti_images_gm[[1]], plane=c('axial'), plot.type=c("multiple"))
```



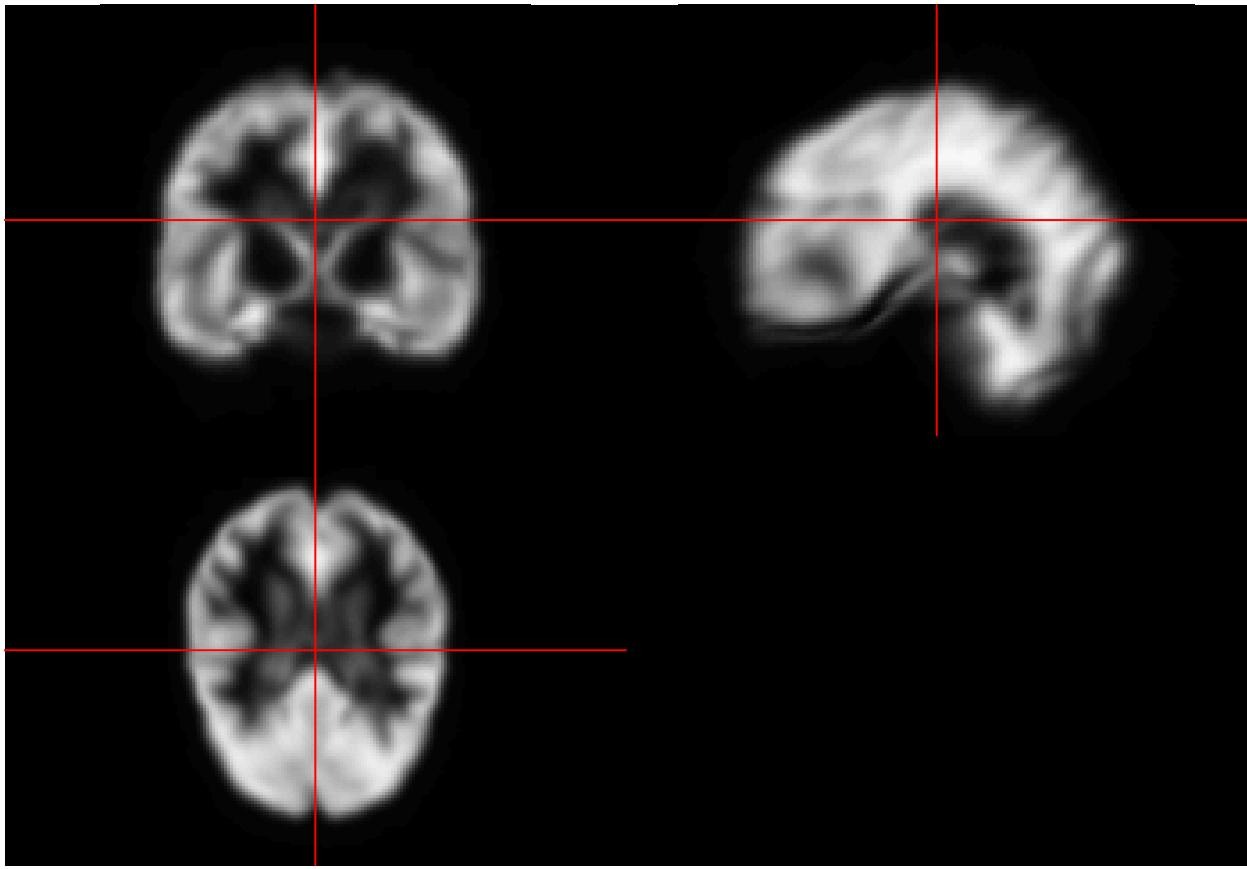
```
oro.nifti::image(x=nifti_images_wm[[1]], plane=c('axial'), plot.type=c("multiple"))
```



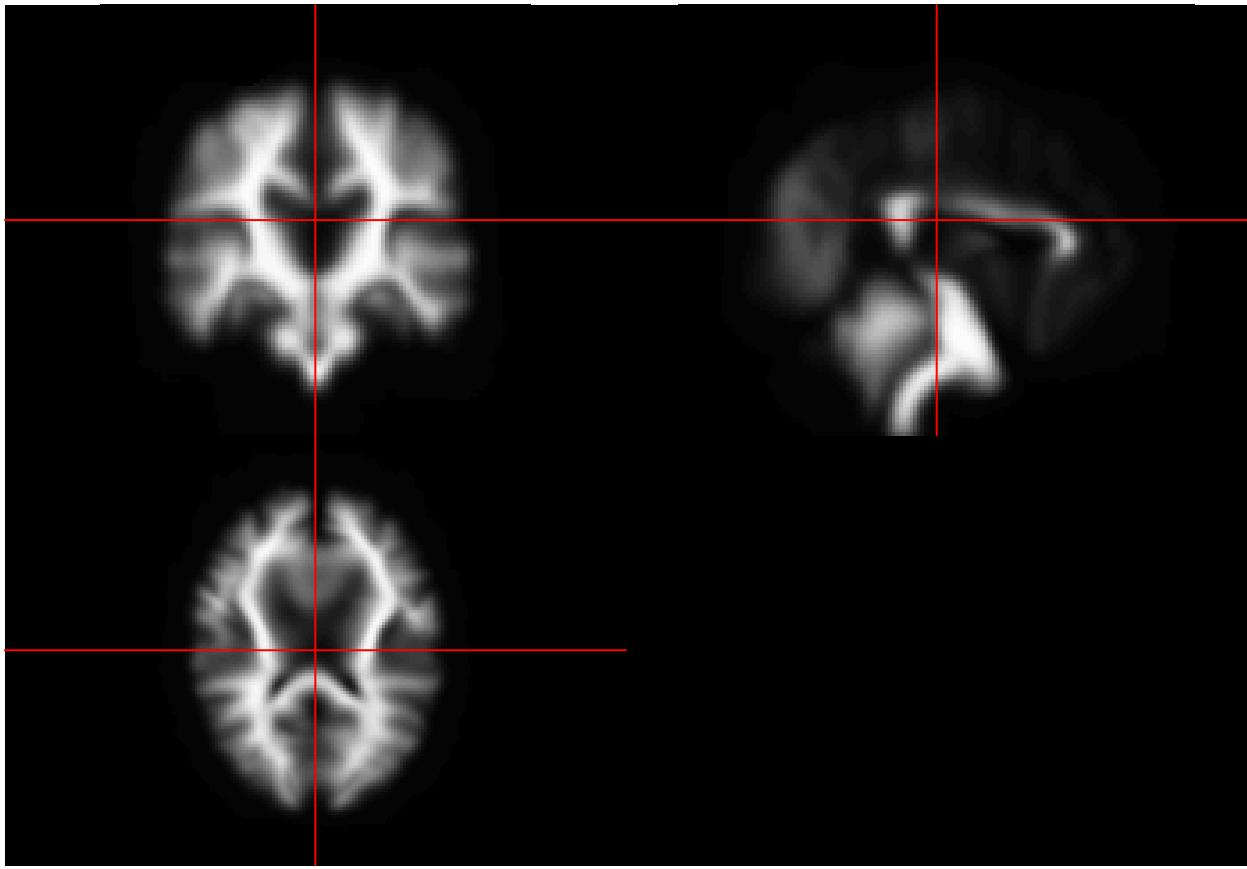
```
# slice 52 of axial plane
oro.nifti::image(x=nifti_images_gm[[2]], z=52, plane=c('axial'), plot.type=c("single"))
```



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



```
ortho2(nifti_images_wm[[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)
save(X_wm, file = file.path(root_dir, "R_data", "X_wm.Rda"))

X <- X_wm$X
y <- X_wm$y

dim(X)

## [1] 434 1015697
kable(X[3:8,500000:500005], format = "markdown")

```

| | V1041269 | V1041270 | V1041271 | V1041272 | V1041273 | V1041274 |
|---|-----------|-----------|-----------|-----------|-----------|-----------|
| 3 | 0.5695567 | 0.5730507 | 0.5626814 | 0.5230017 | 0.4553069 | 0.3679161 |
| 4 | 0.0349222 | 0.0405091 | 0.0708260 | 0.1122733 | 0.1653108 | 0.2148363 |
| 5 | 0.1361559 | 0.0784277 | 0.0504006 | 0.0439253 | 0.0470014 | 0.0584395 |
| 6 | 0.1909431 | 0.0871770 | 0.0356524 | 0.0136465 | 0.0095101 | 0.0141475 |
| 7 | 0.5022411 | 0.4080574 | 0.3077039 | 0.2258514 | 0.1618806 | 0.1226162 |
| 8 | 0.0901738 | 0.0845516 | 0.0961416 | 0.1099358 | 0.1290341 | 0.1471924 |

```

head(as.vector(unique(X[3:8,500000:500005])))

## $V1041269
## [1] 0.56955665 0.03492218 0.13615592 0.19094306 0.50224113 0.09017382
##
## $V1041270
## [1] 0.57305074 0.04050909 0.07842772 0.08717702 0.40805742 0.08455156
##
## $V1041271
## [1] 0.56268138 0.07082600 0.05040057 0.03565240 0.30770391 0.09614156
##
## $V1041272
## [1] 0.52300167 0.11227329 0.04392528 0.01364649 0.22585142 0.10993576
##
## $V1041273
## [1] 0.455306888 0.165310815 0.047001388 0.009510069 0.161880553 0.129034102
##
## $V1041274
## [1] 0.36791611 0.21483633 0.05843950 0.01414747 0.12261619 0.14719239

summary(X[3:8,500000:500005])

```

```

##      V1041269          V1041270          V1041271          V1041272
##  Min.   :0.03492   Min.   :0.04051   Min.   :0.03565   Min.   :0.01365
##  1st Qu.:0.10167  1st Qu.:0.07996  1st Qu.:0.05551  1st Qu.:0.06043
##  Median :0.16355  Median :0.08586  Median :0.08348  Median :0.11110
##  Mean   :0.25400  Mean   :0.21196  Mean   :0.18723  Mean   :0.17144
##  3rd Qu.:0.42442  3rd Qu.:0.32784  3rd Qu.:0.25481  3rd Qu.:0.19746

```

```
##   Max.    :0.56956   Max.    :0.57305   Max.    :0.56268   Max.    :0.52300
##   V1041273          V1041274
##   Min.    :0.00951   Min.    :0.01415
##   1st Qu.:0.06751   1st Qu.:0.07448
##   Median  :0.14546   Median  :0.13490
##   Mean    :0.16134   Mean    :0.15419
##   3rd Qu.:0.16445   3rd Qu.:0.19793
##   Max.    :0.45531   Max.    :0.36792

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, gm_files, comb=TRUE)
save(X_cb, file = file.path(root_dir, "R_data", "X_cb.Rda"))
```