

Supplementary material 1

Supplementary file 1 (how to obtain Pfile from MRI host)

[Open Window 1]

1. cd researchdata/ [Enter]
2. ls [Enter]
3. cd (current folder, for example: 20190628/) [Enter]
4. mkdir (create file name, for example: abc) [Enter]
5. cd (current file name, for example: abc) [Enter]
6. pwd [Enter]

Then we get the created folder path

[Open Window 2]

1. ftp 10.0.12.110 [Enter] (Note: This is an IP address which corresponds to different hosts)
2. sdc [Enter]
3. adw 2.0 [Enter] (Note: This is a password which corresponds to different hosts)
4. bin [Enter]
5. lcd (copy the folder path from Window 1) [Enter]
6. cd /usr/g/mrrow [Enter]
7. ls -tl [Enter] (Note: It shows recent pfiles which we can select and copy these according to experiment time)
8. get P04068.7 (pfile name) [Enter] (Note: Then the pfile was successfully obtained. We repeat the command "get" to obtain the pfile)
9. exit [Enter]

Then we exit window 2 and return to window 1.

Copy pfile to U disk from usb path

1. cd ..
2. su root [Enter]
3. operator [Enter]
4. fdisk -l [Enter] (Then we obtain the path of usb, for example: /del/usb)
5. mount -t vfat (for example: /del/usb) /mnt/usbdisk [Enter]
6. cp -r (The name of the folder created in window 1, for example: abc) /mnt/usbdisk [Enter] Note: You must confirm that window 1 has entered the abc folder before executing this command.
7. cd /mnt/usbdisk [Enter]
8. ls [Enter]
9. cd / [Enter]
10. umount /mnt/usbdisk [Enter]
11. exit [Enter]

Then we copy the pfile to U disk successfully.

Supplementary material 2

Supplementary file 2 (the method to analysis MRS)

The analysis step for single-voxel magnetic resonance spectroscopy using LCModel

1. Turn on the lab computer, enter the username and password
2. Open the virtual machine on the desktop (VMware), right-click on the username (for example: administrator) in the current mode, enter password (for example: dell123)
3. Double-click the terminal, then input `cd. lcmodel` [enter]
4. Click `./lcmgui` [enter]
5. Click the selected profile
6. Select a pfile from the path `/usr/g/spectro/data/abc/20190806`
7. Advanced settings: change control default file: select 3T, click OK
8. Click change basis, then select `press_te35_3t_gsh_v3.basis`
9. Click LCModel, then start to analysis results

The analysis step for multi-voxel magnetic resonance spectroscopy using LCModel and SAGE software

1. Turn on the lab computer, enter the username and password;
2. Open the virtual machine on the desktop (VMware), right-click on the username (for example: administrator) in the current mode, enter password (for example: dell123)
3. Double-click the terminal, then input `cd. lcmodel` [enter]
4. Click terminal: SAGE
5. Import Pfile, select the path `mnt/bgfs/libui /usr/g/spectro/data/abc/20190806`
6. Open a Pfile and reconstruct it: `recons CSI reconstruct`
7. Analysis by LCModel
8. Select the path of LCModel: `SDDEGE:999; SDDEGP:1`
9. Enter `PPMST:4.0, PPMEND:0.2`
10. Select calibration factor: First, the NAA value from single-voxel proton magnetic resonance spectroscopy (NAA_{svs}) is processed obtain in a selected volume of interest. The correction factor of multi-voxel proton magnetic resonance spectroscopy filled with 1, and then NAA value from multi-voxel proton magnetic resonance spectroscopy (NAA_{mvs}) is obtained. The ratio NAA_{mvs}/NAA_{svs} corresponding the same volume of interest was acquired, and it is the final calibration factor.
11. Then run Basis set: `press-te35-3t-gsh-v3-GE.basis`
12. Start to analysis results

Supplementary material 3

Supplementary file 3 (the basic process flow of the RF algorithm)

```
setwd('G:\\WORK-MBY\\data classification management\\revision manuscript')
library(randomForest)
library(readxl)
library(sqldf)
library(stringr)
library(pROC)
library(ROCR)
library(caret)
library(dplyr)
str2num <- function(data){as.numeric(as.character(data))}
df1 <- read_excel('data.xlsx', sheet = 1)
df2 <- read_excel('data.xlsx', sheet = 2)
##model1, ratio1=====
## use sex, age, education, variable NAA/mI
df <- df1
df$grp <- ifelse(df$grp == 'ECs', 0, 1)
df$gender <- factor(df$gender)
df[, str_detect(names(df), '_n$_l_im$_l_m$')] <- NULL
set.seed(22)## Set random seeds
## 65% for train
## 35% for test
train_index <- sample(1:nrow(df), 0.65 * nrow(df), replace = F)
train_df <- df[train_index, ]
test_df <- df[-train_index, ]
rf_fit <- randomForest(grp ~ ., data = train_df, ntree = 35, type = 'regression', mtry = 4, importance = T)
imp <- importance(rf_fit)
imp <- as.data.frame(imp)
imp$vname <- row.names(imp)
names(imp) <- c('inci_mse', 'inc_node_purity', 'vname')
imp <- imp[, c('vname', 'inci_mse')]
imp <- sqldf('select * from imp order by inci_mse desc')
Imp ## use test test
pred <- predict(rf_fit, test_df, type = 'response')
train_pred <- predict(rf_fit, train_df, type = 'response')
test_pred <- pred
pred <- prediction(pred, as.factor(test_df$grp))
perf <- performance(pred, 'tpr', 'fpr')
test_auc <- unlist(slot(performance(pred, 'auc'), 'y.values'))
test_auc ## the optimal segmentation point was found according to the model fitting results
train_cut_off <- roc(response = train_df$grp, predictor = train_pred)
train_cut_df <- cbind(train_cut_off$thresholds, train_cut_off$sensitivities + train_cut_off$specificities)
best_cut_off <- subset(e, e[, 2] == max(e[, 2]))[, 1]
## predictive classification was obtained according to segmentation points
pred_class <- ifelse(pred2 >= best_cut_off, 1, 0)
xtab <- table(pred_class, test_df$grp)
```

```

confusionMatrix(xtab, positive = '1')
cut_off <- roc(response = test_df$grp, predictor = test_pred)
par(mgp = c(2, 0.8, 0), oma = c(0, 0, 0, 0))
plot(1 - cut_off$specificities, cut_off$sensitivities, type = 'l', xlab = '1 - Specificity', ylab = 'Sensitivity', col = 'black', lwd = 2,
     bty = 'l', font.lab = 2, main = 'ROC Curve', cex.lab = 1.6, font = 2)
lines(c(0, 1), c(0, 0), lty = 2)
lines(c(1, 1), c(0, 1), lty = 2)
lines(c(0, 1), c(0, 1), lty = 2)
text(0.5, 0.6, paste('AUC = ', round(test_auc, 2), sep = ''), font = 2, cex = 1.2)
imp$vname
imp <- filter(imp, str_detect(vname, '_nm$') %in% c(T))
imp$x <- nrow(imp):1imp
imp$label <- c('L-Hip(NAA/ml)', 'R-Hip(NAA/ml)', 'L-Pr(NAA/ml)', 'L-PCC(NAA/ml)', 'L-OLWM(NAA/ml)')
par(oma = c(0, 4, 0, 0))
plot(0, NA, xlim = c(min(imp$inci_mse) - 0.5, max(imp$inci_mse) + 0.5), ylim = c(0.5, max(imp$x) + 0.5), type = 'l', xaxs = 'i',
     yaxs = 'i', bty = 'l', xlab = 'Importance measure (%IncMSE)', ylab = '', yaxt = 'n', font.lab = 2)
for (r in 1:nrow(imp)){lines(c(min(imp$inci_mse) - 0.5, imp$inci_mse[r]), rep(imp$x[r], 2), lwd = 1, lty = 2, xaxt = 'none')
points(imp$inci_mse[r], imp$x[r], pch = 19, col = 'red', cex = 1.6)}
axis(side = 1, tck = -0.01, font = 2)
axis(side = 2, at = imp$x, labels = imp$label, las = 2, font = 2)
##900* 630

##model, concentration1=====
## use sex, age, education, variable NAA/ml
df <- df1
df$grp <- ifelse(df$grp == 'ECs', 0, 1)
df$gender <- factor(df$gender)
df[, str_detect(names(df), '_nm$')] <- NULL
set.seed(27)## Set random seeds
##65% for train
##35% for test
train_index <- sample(1:nrow(df), 0.65 * nrow(df), replace = F)
train_df <- df[train_index, ]
test_df <- df[-train_index, ]
rf_fit <- randomForest(grp ~ ., data = train_df, ntree = 35, type = 'regression', # mtry = 3, importance = T)
imp <- importance(rf_fit)
imp <- as.data.frame(imp)
imp$vname <- row.names(imp)
names(imp) <- c('inci_mse', 'inc_node_purity', 'vname')
imp <- imp[, c('vname', 'inci_mse')]
imp <- sqldf('select * from imp order by inci_mse desc')
imp ## use test test
pred <- predict(rf_fit, test_df, type = 'response')
train_pred <- predict(rf_fit, train_df, type = 'response')
test_pred <- pred
pred <- prediction(pred, as.factor(test_df$grp))
perf <- performance(pred, 'tpr', 'fpr')
test_auc <- unlist(slot(performance(pred, 'auc'), 'y.values'))
test_auc ##the optimal segmentation point was found according to the model fitting results

```

```

train_cut_off <- roc(response = train_df$grp, predictor = train_pred)
train_cut_df <- cbind(train_cut_off$thresholds, train_cut_off$sensitivities + train_cut_off$specificities)
best_cut_off <- subset(e, e[, 2] == max(e[, 2]))[, 1]
##predictive classification was obtained according to segmentation points
pred_class <- ifelse(pred2 >= best_cut_off, 1, 0)
xtab <- table(pred_class, test_df$grp)
confusionMatrix(xtab, positive = '1')
cut_off <- roc(response = test_df$grp, predictor = test_pred)
par(mgp = c(2, 0.8, 0), oma = c(0, 0, 0, 0))
plot(1 - cut_off$specificities, cut_off$sensitivities, type = 'l', xlab = '1 - Specificity', ylab = 'Sensitivity', col = 'black', lwd = 2,
bty = 'l', font.lab = 2, main = 'ROC Curve', cex.lab = 1.6, font = 2)
lines(c(0, 1), c(0, 0), lty = 2)
lines(c(1, 1), c(0, 1), lty = 2)
lines(c(0, 1), c(0, 1), lty = 2)
text(0.5, 0.6, paste('AUC = ', round(test_auc, 2), sep = ''), font = 2, cex = 1.2)
imp$vname
imp <- filter(imp, str_detect(vname, '_n$|_m$') %in% c(T))
imp$x <- nrow(imp):1
imp
imp$label <- c('L-Hip(NAA)', 'L-CNC(NAA)', 'L-PCC(NAA)', 'R-Hip(ml)', 'L-Pr(NAA)')
par(oma = c(0, 4, 0, 0))
plot(0, NA, xlim = c(min(imp$inci_mse) - 0.5, max(imp$inci_mse) + 0.5), ylim = c(0.5, max(imp$x) + 0.5), type = 'l', xaxs = 'i',
yaxs = 'i', bty = 'l',
xlab = 'Importance measure (%IncMSE)', ylab = '', yaxt = 'n', font.lab = 2)
for (r in 1:nrow(imp)){lines(c(min(imp$inci_mse) - 0.5, imp$inci_mse[r]), rep(imp$x[r], 2), lwd = 1, lty = 2, xaxt = 'none')
points(imp$inci_mse[r], imp$x[r], pch = 19, col = 'red', cex = 1.6)}
axis(side = 1, tck = -0.01, font = 2)
axis(side = 2, at = imp$x, labels = imp$label, las = 2, font = 2)

##model2, ratio2=====
## use sex, age, education, variable NAA/ml
df <- df2
df$grp <- ifelse(df$grp == 'SCDplus', 0, 1)
df$gender <- factor(df$gender)
df[, str_detect(names(df), '_n$|_im$|_m$')] <- NULL
set.seed(22)##Set random seeds to facilitate duplication of results
##65% for train
##35% for test
train_index <- sample(1:nrow(df), 0.65 * nrow(df), replace = F)
train_df <- df[train_index, ]
test_df <- df[-train_index, ]
rf_fit <- randomForest(grp ~ ., data = train_df, ntree = 35, type = 'regression', mtry = 4, importance = T)
imp <- importance(rf_fit)
imp <- as.data.frame(imp)
imp$vname <- row.names(imp)
names(imp) <- c('inci_mse', 'inc_node_purity', 'vname')
imp <- imp[, c('vname', 'inci_mse')]
imp <- sqldf('select * from imp order by inci_mse desc')
imp

```

```

##use test test
pred <- predict(rf_fit, test_df, type = 'response')
train_pred <- predict(rf_fit, train_df, type = 'response')
test_pred <- pred
pred <- prediction(pred, as.factor(test_df$grp))
perf <- performance(pred, 'tpr', 'fpr')
test_auc <- unlist(slot(performance(pred, 'auc'), 'y.values'))
test_auc
##the optimal segmentation point was found according to the model fitting results
train_cut_off <- roc(response = train_df$grp, predictor = train_pred)
train_cut_df <- cbind(train_cut_off$thresholds, train_cut_off$sensitivities + train_cut_off$specificities)
best_cut_off <- subset(e, e[, 2] == max(e[, 2]))[, 1]
##predictive classification was obtained according to segmentation points
pred_class <- ifelse(pred2 >= best_cut_off, 1, 0)
xtab <- table(pred_class, test_df$grp)
confusionMatrix(xtab, positive = '1')
cut_off <- roc(response = test_df$grp, predictor = test_pred)
par(mgp = c(2, 0.8, 0), oma = c(0, 0, 0, 0))
plot(1 - cut_off$specificities, cut_off$sensitivities, type = 'l', xlab = '1 - Specificity', ylab = 'Sensitivity', col = 'black', lwd = 2,
     bty = 'l', font.lab = 2, main = 'ROC Curve', cex.lab = 1.6, font = 2)
lines(c(0, 1), c(0, 0), lty = 2)
lines(c(1, 1), c(0, 1), lty = 2)
lines(c(0, 1), c(0, 1), lty = 2)
text(0.5, 0.6, paste('AUC = ', round(test_auc, 2), sep = ''), font = 2, cex = 1.2)
imp$vname
imp <- filter(imp, str_detect(vname, '_nm$') %in% c(T))
imp$x <- nrow(imp):1
imp
imp$label <- c('R-PCC(NAA/ml)', 'R-Hip(NAA/ml)', 'L-Pr(NAA/ml)')
par(oma = c(0, 4, 0, 0))
plot(0, NA, xlim = c(min(imp$inci_mse) - 0.5, max(imp$inci_mse) + 0.5),
     ylim = c(0.5, max(imp$x) + 0.5), type = 'l', xaxs = 'i', yaxs = 'i', bty = 'l',
     xlab = 'Importance measure (%IncMSE)', ylab = '', yaxt = 'n', font.lab = 2)
for (r in 1:nrow(imp)){lines(c(min(imp$inci_mse) - 0.5, imp$inci_mse[r]), rep(imp$x[r], 2), lwd = 1, lty = 2, xaxt = 'none')}
points(imp$inci_mse[r], imp$x[r], pch = 19, col = 'red', cex = 1.6)
axis(side = 1, tck = -0.01, font = 2)
axis(side = 2, at = imp$x, labels = imp$label, las = 2, font = 2)
##900* 630

##mode1, concentration2=====
##use sex, age, education, concentration NAA, ml
df <- df2
df$grp <- ifelse(df$grp == 'SCDplus', 0, 1)
df$gender <- factor(df$gender)
df[, str_detect(names(df), '_nm$')] <- NULL
set.seed(67)##Set random seeds to facilitate duplication of results
##65% for train
##35% for test
train_index <- sample(1:nrow(df), 0.65 * nrow(df), replace = F)

```

```

train_df <- df[train_index, ]
test_df <- df[-train_index, ]
rf_fit <- randomForest(grp ~ ., data = train_df, ntree = 35, type = 'regression', # mtry = 4, importance = T)
imp <- importance(rf_fit)
imp <- as.data.frame(imp)
imp$vname <- row.names(imp)
names(imp) <- c('inci_mse', 'inc_node_purity', 'vname')
imp <- imp[, c('vname', 'inci_mse')]
imp <- sqldf('select * from imp order by inci_mse desc')
imp
### use test test
pred <- predict(rf_fit, test_df, type = 'response')
train_pred <- predict(rf_fit, train_df, type = 'response')
test_pred <- pred
pred <- prediction(pred, as.factor(test_df$grp))
perf <- performance(pred, 'tpr', 'fpr')
test_auc <- unlist(slot(performance(pred, 'auc'), 'y.values'))
test_auc
###the optimal segmentation point was found according to the model fitting results
train_cut_off <- roc(response = train_df$grp, predictor = train_pred)
train_cut_df <- cbind(train_cut_off$thresholds, train_cut_off$sensitivities + train_cut_off$specificities)
best_cut_off <- subset(e, e[, 2] == max(e[, 2]))[, 1]
###predictive classification was obtained according to segmentation points
pred_class <- ifelse(pred2 >= best_cut_off, 1, 0)
xtab <- table(pred_class, test_df$grp)
confusionMatrix(xtab, positive = '1')
cut_off <- roc(response = test_df$grp, predictor = test_pred)
par(mgp = c(2, 0.8, 0), oma = c(0, 0, 0, 0))
plot(1 - cut_off$specificities, cut_off$sensitivities, type = 'l', xlab = '1 - Specificity', ylab = 'Sensitivity', col = 'black', lwd = 2,
bty = 'l', font.lab = 2, main = 'ROC Curve', cex.lab = 1.6, font = 2)
lines(c(0, 1), c(0, 0), lty = 2)
lines(c(1, 1), c(0, 1), lty = 2)
lines(c(0, 1), c(0, 1), lty = 2)
text(0.5, 0.6, paste('AUC = ', round(test_auc, 2), sep = ''), font = 2, cex = 1.2)
imp$vname
imp <- filter(imp, str_detect(vname, '_n$l_m$') %in% c(T))
imp$x <- nrow(imp):1
imp
imp$label <- c('R-Hip(NAA)', 'R-PCC(ml)', 'R-PCC(NAA)', 'L-Hip(ml)', 'L-Pr(NAA)')
par(oma = c(0, 4, 0, 0))
plot(0, NA, xlim = c(min(imp$inci_mse) - 0.5, max(imp$inci_mse) + 0.5),
ylim = c(0.5, max(imp$x) + 0.5), type = 'l', xaxs = 'i', yaxs = 'i', bty = 'l',
xlab = 'Importance measure (%IncMSE)', ylab = '', yaxt = 'n', font.lab = 2)
for (r in 1:nrow(imp)){lines(c(min(imp$inci_mse) - 0.5, imp$inci_mse[r]),
rep(imp$x[r], 2), lwd = 1, lty = 2, xaxt = 'none') points(imp$inci_mse[r], imp$x[r], pch = 19, col = 'red', cex = 1.6)}
axis(side = 1, tck = -0.01, font = 2)
axis(side = 2, at = imp$x, labels = imp$label, las = 2, font = 2)

```


Supplementary material 4

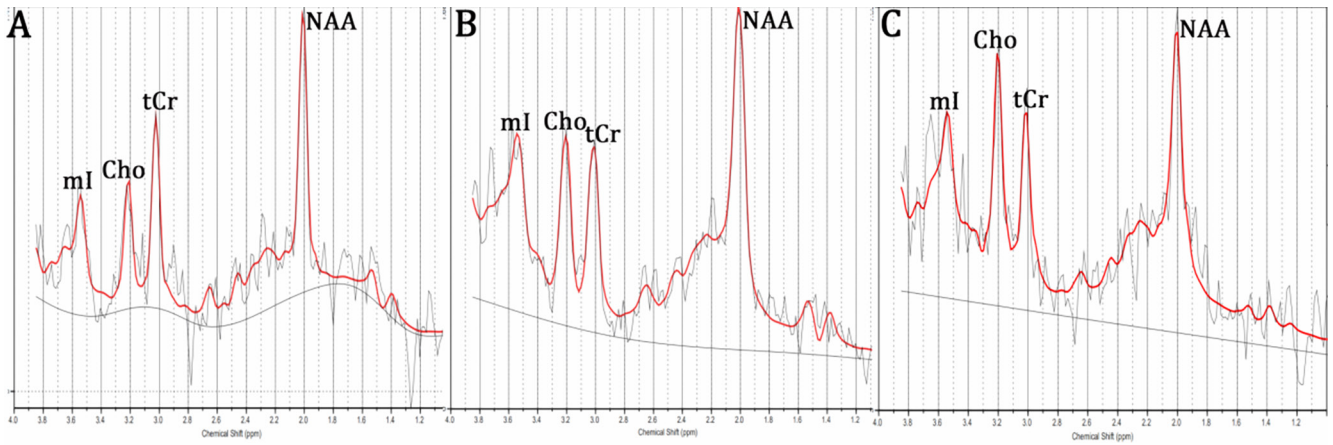


Figure S1 Examples of attained spectra from the left hippocampus. A = elderly controls, B = subjective cognitive decline plus, C = amnesic mild cognitive impairment.

Supplementary material 5

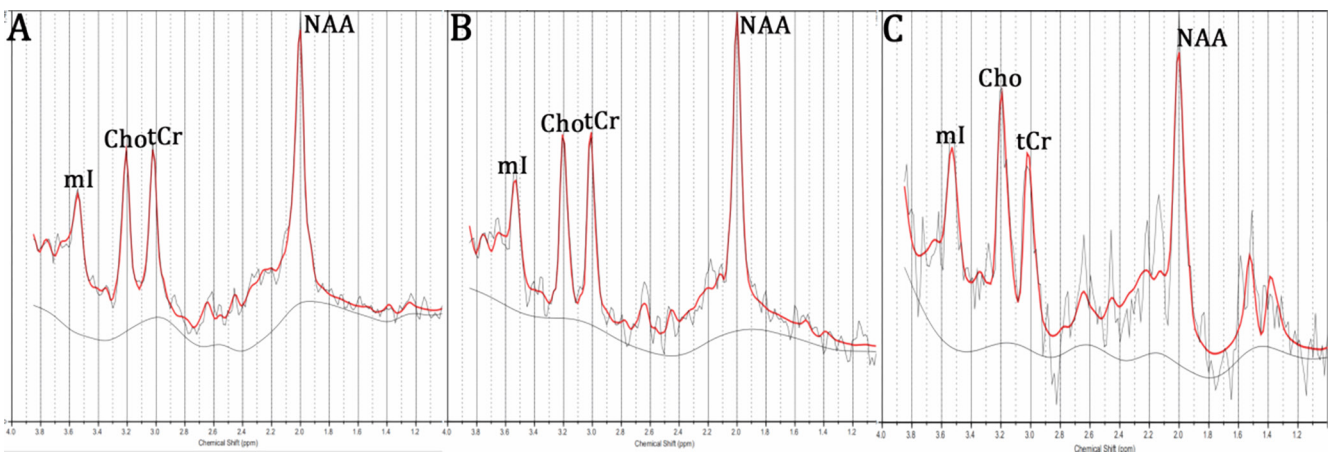


Figure S2 Examples of attained spectra from the right hippocampus. A = elderly controls, B = subjective cognitive decline plus, C = amnesic mild cognitive impairment.

Supplementary material 6

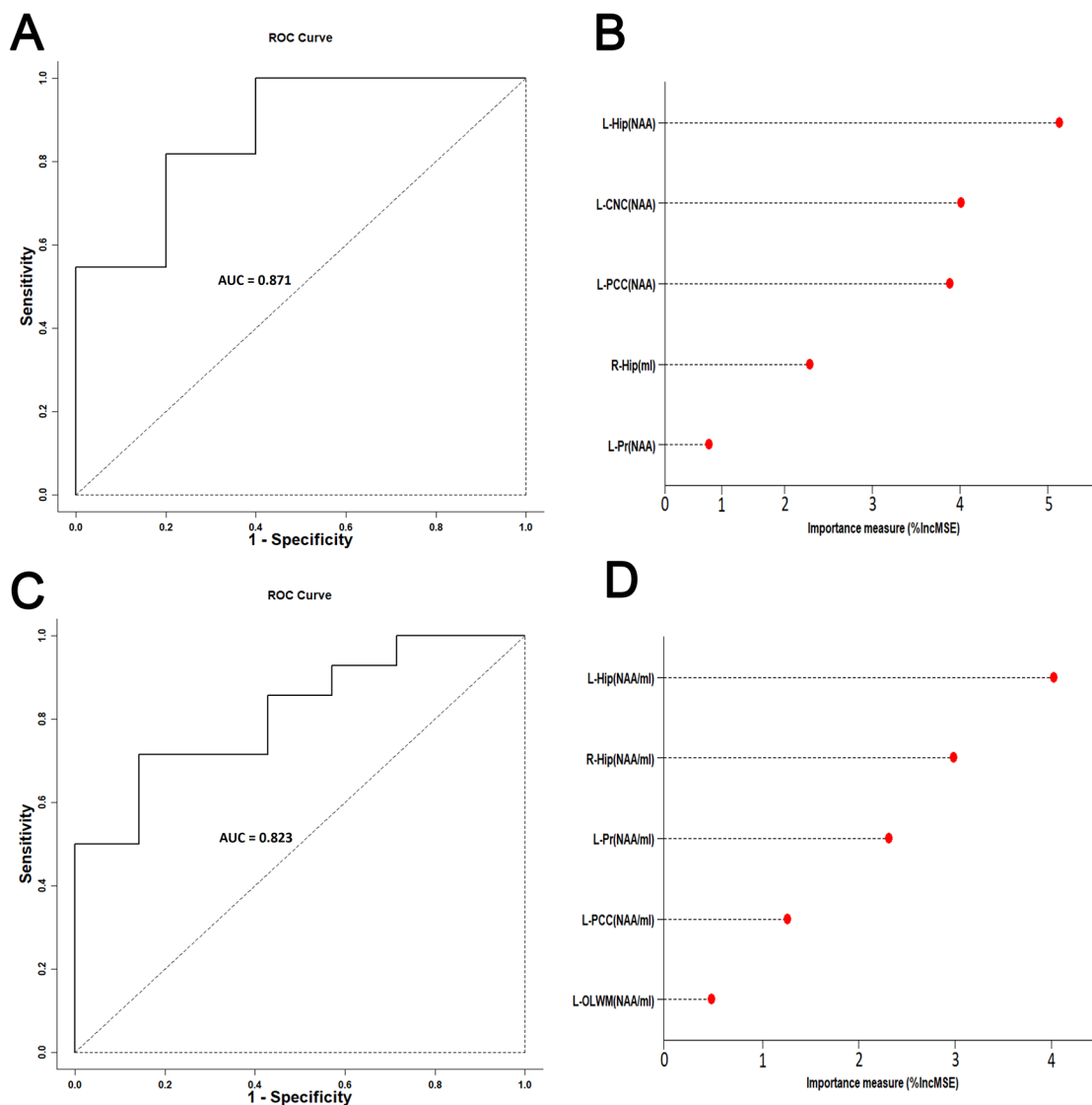


Figure S3 The AUC area and importance measurements from the random forest (RF) based on the testing dataset in discriminating participants with SCD plus from ECs. The horizontal and vertical coordinates represent specificity and sensitivity, respectively. A and B are from the RF in discriminating SCD plus from ECs using NAA and mI values. After variable selection, RF predicted with 0.871 accuracy; C and D are from the RF in discriminating SCD plus from ECs using NAA/mI ratio. After variable selection, RF predicted with 0.823 accuracy. Abbreviations: L-Hip = left hippocampus; R-Hip = right hippocampus; L-PCC = left posterior cingulate cortex; L-Pr = left precuneus; L-OLWM = left white matter of occipital lobe; L-CNC = left caput nuclei caudati; NAA = N-acetylaspartate; mI = myoinositol.

Supplementary material 7

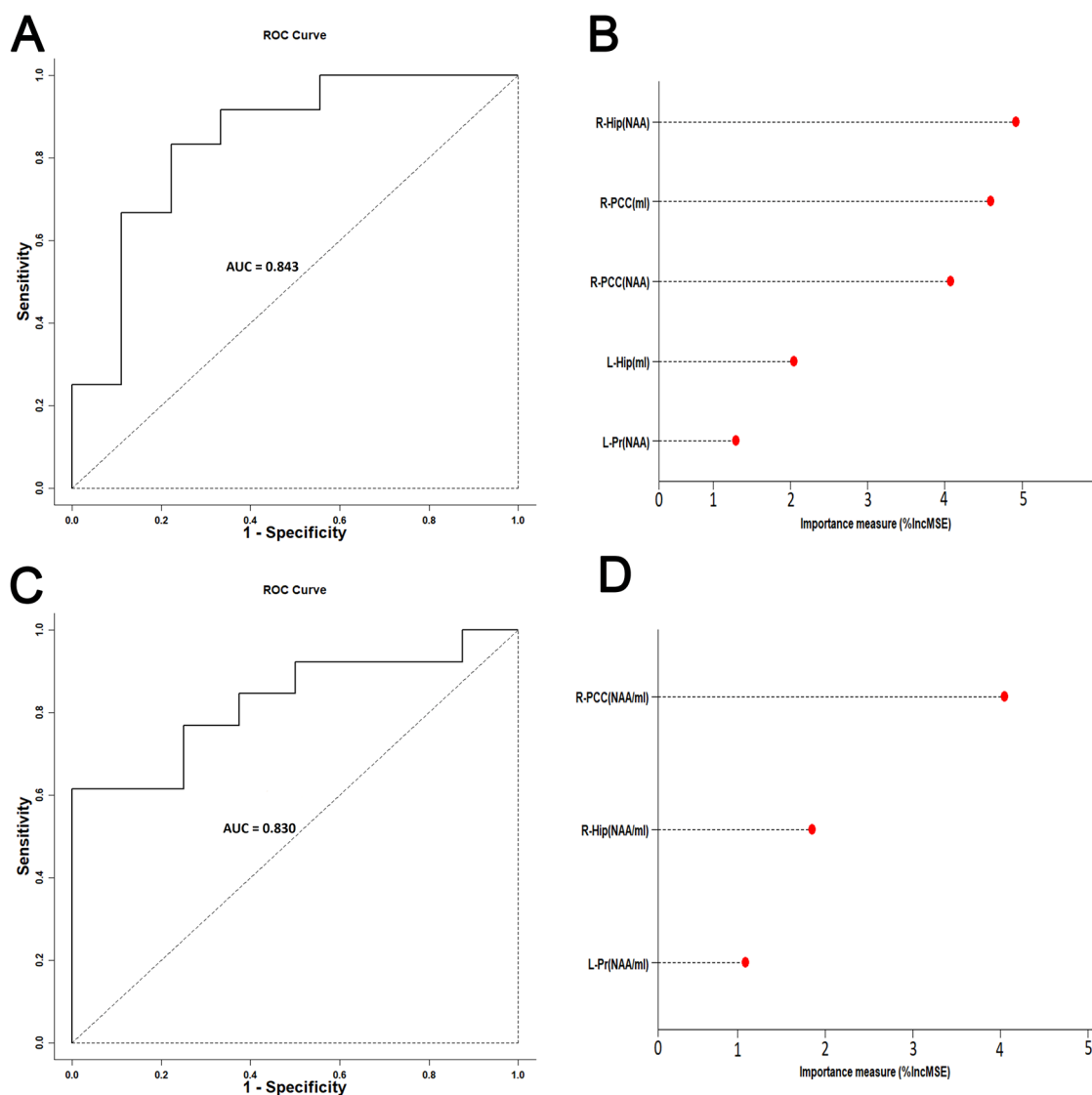


Figure S4 The AUC area and importance measurements from the random forest (RF) based on the testing dataset in discriminating participants with SCD plus from aMCI. The horizontal and vertical coordinates represent specificity and sensitivity, respectively. A and B are from the RF in discriminating SCD plus from aMCI using NAA and mI values. After variable selection, RF predicted with 0.843 accuracy; C and D are from the RF in discriminating SCD plus from aMCI using NAA/mI ratio. After variable selection, RF predicted with 0.830 accuracy. Abbreviations: L-Hip = left hippocampus; R-Hip = right hippocampus; L-Pr = left precuneus; R-PCC = right posterior cingulate; NAA = N-acetylaspartate; mI = myoinositol.

Supplementary material 8

Table S1 Group differences in VOI sizes and S/N in the left brain hemisphere of ECs, SCD plus, and aMCI participants

VOIs	Parameters	EC (n=33) (means ± SD)	SCD plus (n=27) (means ± SD)	aMCI (n=31) (means ± SD)	F	P values
L-Hip	VOI size (cm ³)	3.202 ± 0.468	3.035 ± 0.475	2.997 ± 0.385	1.123	0.386
	S/N	7.112 ± 1.108	6.962 ± 1.209	6.826 ± 1.369	1.822	0.125
L-PCC	VOI size (cm ³)	1.205 ± 0.076	1.186 ± 0.080	1.179 ± 0.091	1.231	0.322
	S/N	4.786 ± 1.087	4.658 ± 1.169	4.364 ± 1.027	0.664	0.738
L-Pr	VOI size (cm ³)	1.208 ± 0.083	1.188 ± 0.079	1.185 ± 0.072	1.085	0.327
	S/N	4.845 ± 1.100	4.591 ± 1.225	4.532 ± 1.132	1.763	0.141
L-OLWM	VOI size (cm ³)	2.392 ± 0.261	2.386 ± 0.237	2.375 ± 0.227	1.598	0.265
	S/N	5.852 ± 1.209	5.745 ± 1.313	5.585 ± 1.367	0.826	0.525
L-DT	VOI size (cm ³)	3.576 ± 0.531	3.493 ± 0.436	3.489 ± 0.391	1.233	0.297
	S/N	6.206 ± 1.218	6.126 ± 1.152	6.057 ± 1.230	0.743	0.658
L-FLWM	VOI size (cm ³)	3.586 ± 0.516	3.568 ± 0.478	3.576 ± 0.483	1.084	0.405
	S/N	4.602 ± 1.219	4.570 ± 1.120	4.346 ± 1.109	1.819	0.097
L-LN	VOI size (cm ³)	1.199 ± 0.096	1.199 ± 0.092	1.194 ± 0.081	1.272	0.295
	S/N	4.349 ± 1.551	4.272 ± 1.556	4.016 ± 1.655	1.432	0.252
L-CNC	VOI size (cm ³)	1.199 ± 0.079	1.183 ± 0.077	1.185 ± 0.107	1.316	0.300
	S/N	4.506 ± 1.206	4.307 ± 1.462	4.279 ± 1.547	1.462	0.281

Abbreviations: ECs = elderly controls; SCD plus = subjective cognitive decline plus; aMCI = amnesic mild cognitive impairment; L-Hip = left hippocampus; L-PCC = left posterior cingulate cortex; L-Pr = left precuneus; L-DT = left dorsal thalamus; L-OLWM = left white matter of the occipital lobe; L-FLWM = left white matter of the frontal lobe; L-LN = left lenticular nucleus; L-CNC = left caput nuclei caudati; VOIs = volume of interests; S/N = signal to noise. Values are presented as the mean ± SD.

Supplementary material 9

Table S2 Group differences in VOI size (cm³) and S/N in the right brain hemisphere of ECs, SCD plus, and aMCI participants.

VOIs	Parameters	EC (n=33) (means ± SD)	SCD plus (n=27) (means ± SD)	aMCI (n=31) (means ± SD)	F	p values
R-Hip	VOI size (cm ³)	3.458 ± 0.713	3.426 ± 0.475	3.186 ± 1.146	1.166	0.367
	S/N	5.706 ± 1.215	5.962 ± 1.209	5.526 ± 1.317	1.452	0.263
R-PCC	VOI size (cm ³)	1.188 ± 0.084	1.193 ± 0.078	1.785 ± 0.103	1.031	0.525
	S/N	4.686 ± 1.047	4.278 ± 1.038	4.658 ± 1.122	0.554	0.685
R-Pr	VOI size (cm ³)	1.211 ± 0.123	1.195 ± 0.108	1.186 ± 0.087	1.485	0.231
	S/N	4.872 ± 1.134	4.682 ± 1.036	4.553 ± 1.152	1.363	0.257
R-OLWM	VOI size (cm ³)	2.383 ± 0.261	2.396 ± 0.251	2.378 ± 0.239	1.598	0.246
	S/N	5.852 ± 1.321	5.745 ± 1.402	5.292 ± 1.567	0.626	0.586
R-DT	VOI size (cm ³)	2.593 ± 0.527	3.577 ± 0.459	3.563 ± 0.472	1.233	0.297
	S/N	6.312 ± 1.334	6.363 ± 1.139	6.238 ± 1.415	0.643	0.756
R-FLWM	VOI size (cm ³)	3.591 ± 0.508	3.573 ± 0.423	3.565 ± 0.512	1.684	0.206
	S/N	4.711 ± 1.308	4.660 ± 1.412	4.523 ± 1.314	2.019	0.159
R-LN	VOI size (cm ³)	1.205 ± 0.101	1.192 ± 0.090	1.184 ± 0.107	1.172	0.195
	S/N	4.398 ± 1.651	4.372 ± 1.527	4.335 ± 1.625	2.332	0.089
R-CNC	VOI size (cm ³)	1.196 ± 0.094	1.182 ± 0.103	1.881 ± 0.095	2.027	0.165
	S/N	4.676 ± 1.512	4.707 ± 1.561	4.856 ± 1.397	1.465	0.268

Abbreviations: ECs = elderly controls; SCD plus = subjective cognitive decline plus; aMCI = amnesic mild cognitive impairment; R-Hip = right hippocampus; R-PCC = right posterior cingulate cortex; R-Pr = right precuneus; R-DT = right dorsal thalamus; R-OLWM = right white matter of the occipital lobe; R-FLWM = right white matter of the frontal lobe; R-LN = right lenticular nucleus; R-CNC = right caput nuclei caudati; VOI = volume of interest; S/N = signal to noise. Values are presented as the mean ± SD.