Cross-Validation of Optimized Composites for Preclinical Alzheimer's

Appendix A. Minimizing the minimum detectable δ

We provide sample Rcode that was used to determine weights that would minimize the minimum detectable δ as a proportion of the group difference between the target group (e.g. Aβ+) and a control group (e.g Aβ-). We demonstrate the numerical optimization method on a simulated dataset.

Appendix A.1. Load required Rpackages

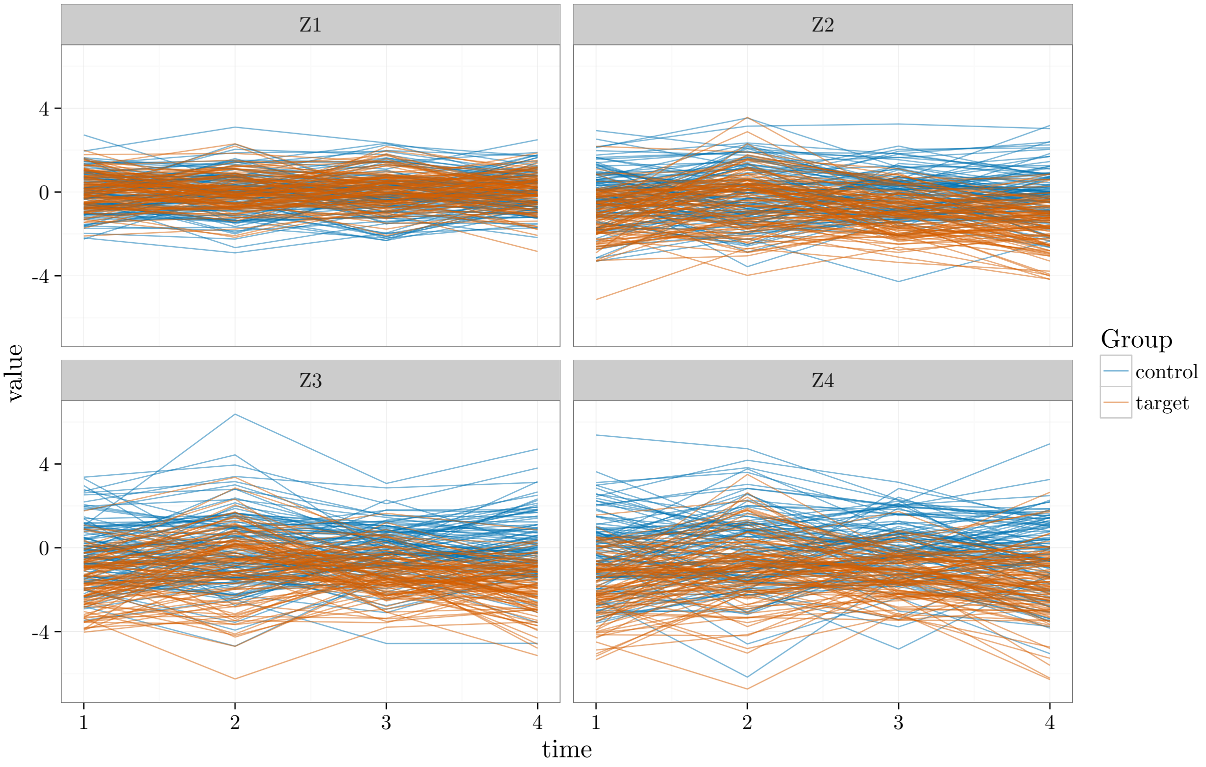
```
library(mvtnorm)   # simulating multivariate normal data
library(nlme)      # fitting the Mixed Model of Repeated Measures (MMRM)
library(contrast)
library(ggplot2)   # plotting
library(longpower) # power calculations for longitudinal data
library(tidyr)     # data manipulation
library(dplyr)
```

Appendix A.2. Simulation parameters

```
# control group mean is flat
control_mean <- rep(0, 16)
# target grou mean is generally decreasing
target_mean <- -c(
  0, 1.00, 1.50, 1.80,  # component 1
  0, 0.50, 0.75, 1.00,  # component 2
  0, 1.10, 1.20, 1.30,  # component 3
  0, 1.50, 1.75, 2.00   # component 4
)
# standard deviations of 4 components over 4 timepoints
component_sd <- c(
  1, 1.25, 1.50, 1.75, # component 1
  1, 1.50, 1.75, 2.00, # component 2
  1, 1.10, 1.20, 1.30, # component 3
  1, 1.25, 1.50, 1.75  # component 4
)
# compound symmetric correlation matrix
R <- matrix(0.5, nrow = 16, ncol = 16)
diag(R) <- 1
# total sample size
N <- 200
# retention rate across 4 timepoints
retention <- c(1, 0.9, 0.8, 0.7)
# variance-covariance matrix
VarCov <- diag(component_sd) %*% R %*% diag(component_sd)
```

Appendix A.3. Simulate data

```
set.seed(20161118)
dd_control_wide <- cbind(group=0, rmvnorm(N/2, mean=control_mean, sigma=VarCov))
dd_target_wide <- cbind(group=1, rmvnorm(N/2, mean=target_mean, sigma=VarCov))
dd_wide <- as.data.frame(rbind(dd_control_wide, dd_target_wide))
colnames(dd_wide)[-1] <- paste(paste0('Z', 1:4), rep(1:4, each=4), sep='_')
dd_wide$id <- 1:N
dd_long <- gather(dd_wide, component_time, value, -id, -group) %>%
  separate(component_time, c('component', 'time'), sep='_') %>%
  mutate(time=as.numeric(time), Time=as.factor(time),
  Group = factor(group, levels=0:1, labels=c('control', 'target')))
dd <- spread(dd_long, component, value)
head(dd)

  group id time Time   Group      Z1     Z2      Z3     Z4
1     0  1    1    1 control  0.0589 -0.985  0.8872  2.246
2     0  1    2    2 control  0.3227 -0.954 -0.9109  0.191
3     0  1    3    3 control  0.2882  0.912 -1.1351 -0.148
4     0  1    4    4 control  0.5209  0.445  0.0868 -0.849
5     0  2    1    1 control -0.2579 -1.993 -0.7859  0.691
6     0  2    2    2 control -0.2680 -1.269 -1.8823 -0.137

ggplot(dd_long, aes(x = time, y = value, color = Group, group=id)) +
  geom_line(alpha=0.5) + facet_wrap(~component)
```



Appendix A.4. The function to be optimized

The following function returns the minimum detectable effect for given vector of weights

```
delta_as_proportion_of_group_difference <- function(weights, data){
  # calculate composite, Y, as weighted sum of 4 components, given weights
  data$Y <- apply(data[, paste0('Z', 1:4)], 1,
    FUN = weighted.mean, w = weights, na.rm = TRUE)*4

  # fit the MMRM
  fit <- gls(Y ~ Time*Group, data = data,
    correlation = corCompSymm(form = ~ time | id),
          weights = varIdent(form = ~ 1 | time)
  )

  # obtain the treatment effect at last visit
  ctrast <- contrast(fit,
    list(Time=levels(data$Time)[4], Group = levels(data$Group)[2]),
    list(Time=levels(data$Time)[4], Group = levels(data$Group)[1]))

  # extract the variance-covariance parameters for power calculation
  phi <- coef(fit$modelStruct, unconstrained = FALSE)
  Ra <- matrix(phi['corStruct.Rho'], nrow = 4, ncol = 4)
  diag(Ra) <- 1
  v <- coef(fit$modelStruct$varStruct, uncons = FALSE, allCoef = TRUE)

  # determin minimum detectable delta at given power, alpha, retention raet, and sample size
  pow <- power.mmrm(Ra, ra=retention,
        sigmaa = fit$sigma*v[length(v)],
        power = 0.80, sig.level = 0.05, N=200)

  abs(pow$delta/ctrast$Contrast[1])
}
```

So, for example, with equal weights, we get:

```
delta_as_proportion_of_group_difference(rep(0.25,4), dd)

    1
0.345
```

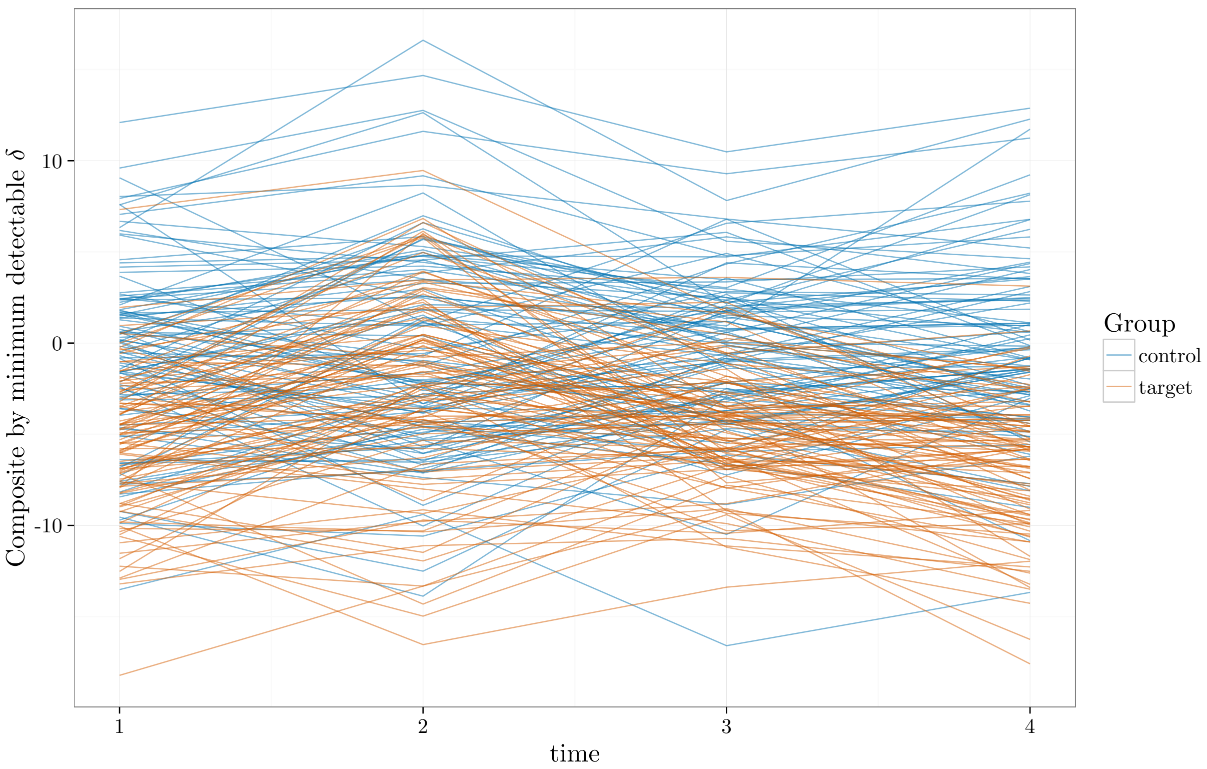
Appendix A.5. The optimization

To find the weights which minimize `delta_as_proportion_of_group_difference`:

```
opt_weights <- constrOptim(theta = c(0.25, 0.25, 0.25, 0.25),
  f=delta_as_proportion_of_group_difference,
  ui = diag(1, 4), ci = 0,
  method = "Nelder-Mead", data = dd)
```

And the optimized composite is:

```
dd$Y.min.delta <- apply(dd[, paste0('Z', 1:4)], 1,
  FUN = weighted.mean, w = opt_weights$par, na.rm = TRUE)*4
ggplot(dd, aes(x = time, y = Y.min.delta, color = Group, group=id)) +
  geom_line(alpha=0.5) +
  ylab('Composite by minimum detectable $\\delta$')
```


Composite by minimum detectable δ
10
0
-10
1
2
3
4
time
Group
control
target

```
fit.min.delta <- gls(Y.min.delta ~ Time*Group, data = dd,
  correlation = corCompSymm(form = ~ time | id),
  weights = varIdent(form = ~ 1 | time)
)
summary(fit.min.delta)

Generalized least squares fit by REML
  Model: Y.min.delta ~ Time * Group
  Data: dd
   AIC  BIC logLik
  4267 4328  -2120

Correlation Structure: Compound symmetry
 Formula: ~time | id
 Parameter estimate(s):
  Rho
0.667
Variance function:
 Structure: Different standard deviations per stratum
 Formula: ~1 | time
 Parameter estimates:
    1     2     3     4
1.000 1.284 0.871 0.988

Coefficients:
                  Value Std.Error t-value p-value
(Intercept)       -0.50     0.442   -1.12  0.2610
Time2              0.05     0.427    0.12  0.9009
Time3              0.26     0.341    0.78  0.4377
Time4              0.43     0.358    1.20  0.2324
Grouptarget       -5.12     0.625   -8.20  0.0000
Time2:Grouptarget  3.14     0.604    5.20  0.0000
Time3:Grouptarget  0.65     0.482    1.35  0.1788
Time4:Grouptarget -1.62     0.506   -3.20  0.0014

 Correlation:
                  (Intr) Time2  Time3  Time4  Grptrg Tm2:Gr Tm3:Gr
Time2             -0.148
Time3             -0.542  0.413
Time4             -0.420  0.421  0.533
Grouptarget       -0.707  0.104  0.383  0.297
Time2:Grouptarget  0.104 -0.707 -0.292 -0.298 -0.148
Time3:Grouptarget  0.383 -0.292 -0.707 -0.377 -0.542  0.413
Time4:Grouptarget  0.297 -0.298 -0.377 -0.707 -0.420  0.421  0.533

Standardized residuals:
    Min      Q1     Med      Q3     Max
-4.2520 -0.6174  0.0169  0.6440  3.0062

Residual standard error: 4.42
Degrees of freedom: 800 total; 792 residual
```

```r
contrast(fit.min.delta,
  list(Time=levels(dd$Time)[4], Group = levels(dd$Group)[2]),
  list(Time=levels(dd$Time)[4], Group = levels(dd$Group)[1])
)
```

```
gls model parameter contrast

 Contrast  S.E. Lower Upper     t  df Pr(>|t|)
    -6.74 0.617 -7.95 -5.53 -10.9 792        0
```

Appendix B. Optimized composite via logistic regression of group status

Appendix B.1. Derive the component change scores

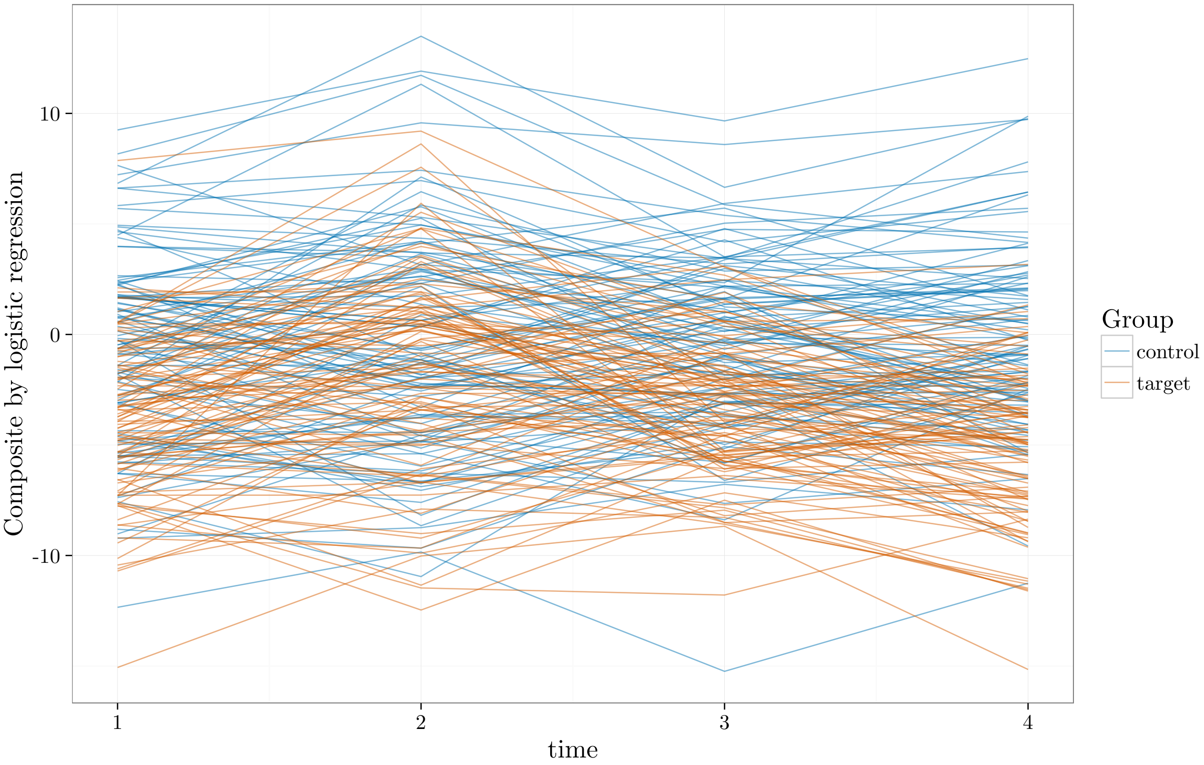
```
  id group Z1.time Z2.time Z3.time Z4.time
1  1     0   0.135   0.616 -0.2625 -0.9623
2  2     0  -0.175   0.611  0.1136 -0.6167
3  3     0   0.437   0.292  0.0983 -0.0734
4  4     0   0.298   0.270 -0.0532 -1.5165
5  5     0   0.133   0.361  0.4870 -0.7063
6  6     0   0.362   0.199  0.5981  0.1439
```

Appendix B.2. Fit logistic regression model

```
Call:
glm(formula = group ~ Z1.time + Z2.time + Z3.time + Z4.time,
    family = binomial, data = slopes)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.808  -1.088  -0.113   1.108   1.953

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.123640   0.153037   -0.81  0.41914
Z1.time     -0.633629   0.500755   -1.27  0.20575
Z2.time     -1.292543   0.370830   -3.49  0.00049 ***
Z3.time     -0.752636   0.339590   -2.22  0.02667 *
Z4.time     -0.000453   0.283759    0.00  0.99873
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 277.26  on 199  degrees of freedom
Residual deviance: 255.17  on 195  degrees of freedom
AIC: 265.2

Number of Fisher Scoring iterations: 4
```

Appendix B.3. Apply logistic regression weights

```r
dd$Y.logistic <- apply(dd[, paste0('Z', 1:4)], 1,
  FUN = weighted.mean, w = weight_fit$coef[-1], na.rm = TRUE)*4
ggplot(dd, aes(x = time, y = Y.logistic, color = Group, group=id)) +
  geom_line(alpha=0.5) +
  ylab('Composite by logistic regression')
```


Composite by logistic regression
10
0
-10
1
2
3
4
time
Group
control
target

```
fit.logistic <- gls(Y.logistic ~ Time*Group, data = dd,
  correlation = corCompSymm(form = ~ time | id),
  weights = varIdent(form = ~ 1 | time)
)
summary(fit.logistic)

Generalized least squares fit by REML
  Model: Y.logistic ~ Time * Group
  Data: dd
   AIC  BIC logLik
  4024 4084  -1999

Correlation Structure: Compound symmetry
 Formula: ~time | id
 Parameter estimate(s):
  Rho
0.693
Variance function:
 Structure: Different standard deviations per stratum
 Formula: ~1 | time
 Parameter estimates:
    1     2     3     4
1.000 1.231 0.895 0.973

Coefficients:
                  Value Std.Error t-value p-value
(Intercept)       -0.49     0.392   -1.26  0.2076
Time2              0.08     0.353    0.23  0.8156
Time3              0.22     0.293    0.75  0.4521
Time4              0.48     0.303    1.58  0.1152
Grouptarget       -3.37     0.555   -6.07  0.0000
Time2:Grouptarget  2.25     0.499    4.51  0.0000
Time3:Grouptarget  0.32     0.415    0.77  0.4407
Time4:Grouptarget -1.67     0.429   -3.89  0.0001

 Correlation:
                  (Intr) Time2  Time3  Time4  Grptrg Tm2:Gr Tm3:Gr
Time2             -0.163
Time3             -0.507  0.431
Time4             -0.420  0.435  0.533
Grouptarget       -0.707  0.115  0.359  0.297
Time2:Grouptarget  0.115 -0.707 -0.305 -0.308 -0.163
Time3:Grouptarget  0.359 -0.305 -0.707 -0.377 -0.507  0.431
Time4:Grouptarget  0.297 -0.308 -0.377 -0.707 -0.420  0.435  0.533

Standardized residuals:
     Min       Q1      Med       Q3      Max
-4.26438 -0.67296  0.00579  0.66871  3.27230

Residual standard error: 3.92
Degrees of freedom: 800 total; 792 residual
```

```
contrast(fit.logistic,
  list(Time=levels(dd$Time)[4], Group = levels(dd$Group)[2]),
  list(Time=levels(dd$Time)[4], Group = levels(dd$Group)[1])
)

gls model parameter contrast

 Contrast S.E. Lower Upper     t  df Pr(>|t|)
    -5.04 0.54  -6.1 -3.98 -9.33 792        0
```

Appendix C. Distribution of component Z scores across studies

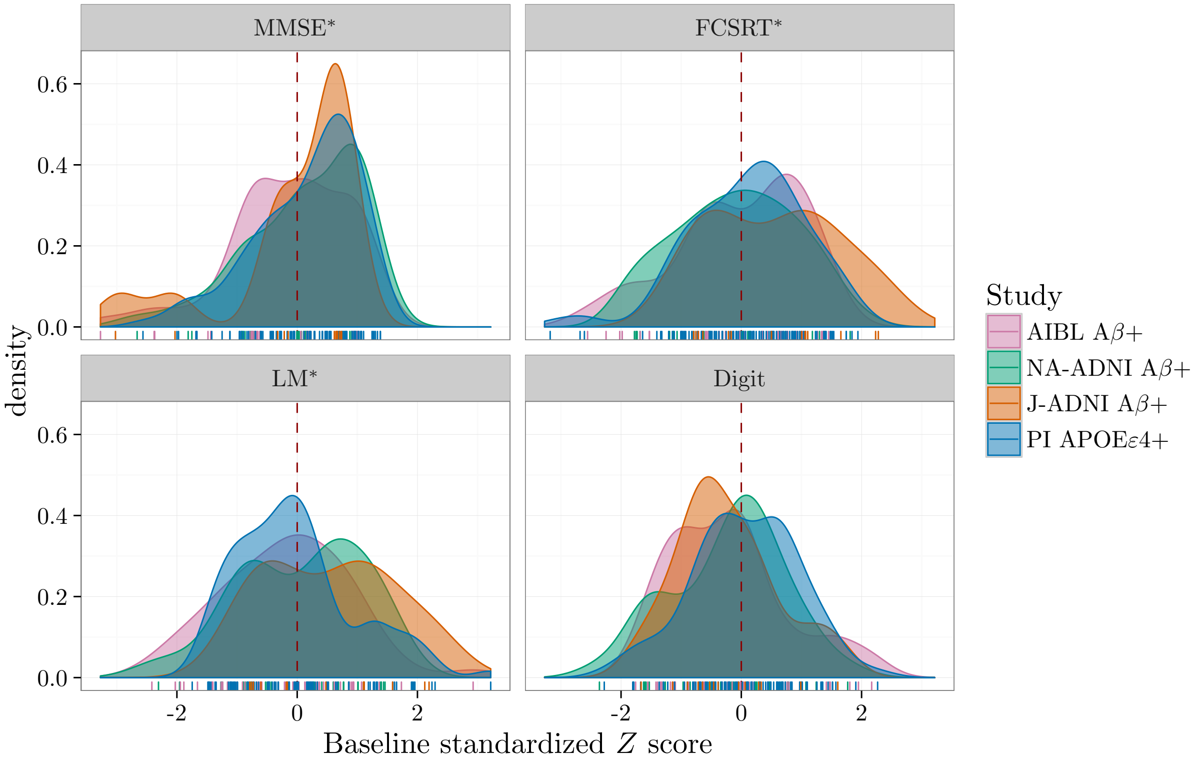


Figure C.1: Density estimates show the distribution of each component score at baseline, mean centered and standardized relative to the each sample's component standard deviation. For each study, we restricted the sample to the at risk Aβ+ population (or APOEε4+ for PI study).
AIBL = Australian Imaging, Biomarkers and Lifestyle; ADNI = Alzheimer's Disease Neuroimaging Initiative; NA-ADNI = North American ADNI; J-ADNI = Japan-ADNI; PI = Alzheimer's Disease Cooperative Study Prevention Instrument; CDR-G = Clinical Dementia Rating Global; MMSE = Mini-Mental State Exam; FCSRT = Free and Cued Selective Reminding Test; LM = Logical Memory; Digit = Digit Symbol Substitution.
* See Table 1 for actual tests used in each study.

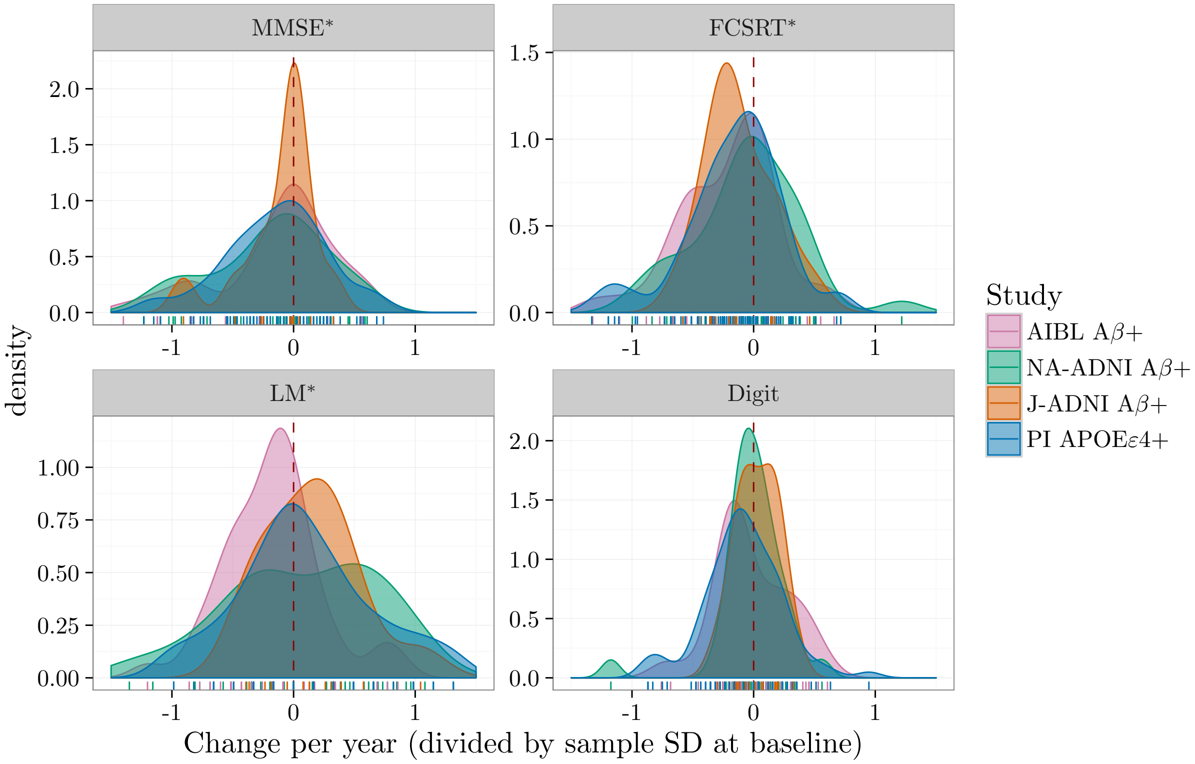


Figure C.2: Density estimates show the distribution of each component change per year standardized relative to the each sample's baseline standard deviation. Change scores are the slope parameters from ordinary least squares linear regression models fit separately for each subject. For each study, we restricted the sample to the at risk Aβ+ population (or APOEε4+ for PI study). AIBL = Australian Imaging, Biomarkers and Lifestyle; ADNI = Alzheimer's Disease Neuroimaging Initiative; NA-ADNI = North American ADNI; J-ADNI = Japan-ADNI; PI = Alzheimer's Disease Cooperative Study Prevention Instrument; CDR-G = Clinical Dementia Rating Global; MMSE = Mini-Mental State Exam; FCSRT = Free and Cued Selective Reminding Test; LM = Logical Memory; Digit = Digit Symbol Substitution.
* See Table 1 for actual tests used in each study.