Supplemental File B of

MicrobiotaProcess: A comprehensive R package for managing and analyzing microbiome and other ecological data within the tidy framework

Shuangbin Xu, Li Zhan, Wenli Tang, Qianwen Wang, Zehan Dai, Lang Zhou, Tingze Feng, Meijun Chen, Shanshan Liu, Xiaocong Fu, Tianzhi Wu, Erqiang Hu and Guangchuang Yu*

*correspondence: Guangchuang Yu <gcyu1@smu.edu.cn>

1 Comparing the performance of the models built based on significant differential clades and significant differential OTUs.

To evaluate the supervised classification accuracy with clinical outcomes based on differential clades and differential OTUs, we built the randomForest model and assessed the classifier performance for some diseases (Hall et al. 2017; Ijaz et al. 2017) (Half of the samples are used for training and the other half is used for prediction, which was repeated sampling 100 times, and finally the AUC score was calculated) with the significant differential clades and OTUs respectively.

```
mpse2 <- readRDS('./data/IBD_data/mpse2.RDS')
mpse3 <- mpse2 %>% dplyr::filter(Class != 'c__un_p__Proteobacteria')
mpse3 %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='geometric.mean'
    ) -> mpse.balance.node

mpse.balance.node %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = Group,
      fc.method = 'compare_mean'
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_Group)) -> ba.node.sign

bla.sign.da.mean <- mpse3 %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='mean'
    ) %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = Group,
      fc.method = 'compare_mean'
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_Group)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, Group) %>%
```

```r
    tidyr::pivot_wider(id_cols=c('Sample', 'Group'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('Group', as.factor)

bla.sign.da.median <- mpse3 %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='median'
    ) %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = Group,
      fc.method = 'compare_mean'
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_Group)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, Group) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'Group'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('Group', as.factor)

ba.node.sign2 <- ba.node.sign %>%
                 tidyr::unnest(Balance_offspring) %>%
                 tidyr::unnest(offspringTiplabel)

sample.da.CD <- mpse3 %>% mp_extract_sample %>%
    dplyr::select(Sample, Group)

bla.sign.da <- ba.node.sign %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, Group) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'Group'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('Group', as.factor)

otu.sign.da <- mpse3 %>% mp_extract_feature() %>%
    filter(!is.na(Sign_Group)) %>%
    tidyr::unnest(RareAbundanceBySample) %>%
    select(OTU, RelRareAbundanceBySample, Sample, Group) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'Group'), names_from='OTU', values_from=RelRareAbundanceBySample) %>
    dplyr::mutate_at('Group', as.factor)

# xx <- curatedMetagenomicData('HallAB_2017.relative_abundance', dryrun=F)
# mpse.ibd.HallAB <- xx[[1]] %>% as.MPSE
mpse.ibd.HallAB <- readRDS('./data/curatedMetagenomicData/mpse.ibd_HallAB_2017.RDS')

# xx <- curatedMetagenomicData('IjazUZ_2017.relative_abundance', dryrun=F)
# mpse.ibd.IjazUZ <- xx[[1]] %>% as.MPSE
mpse.ibd.IjazUZ <- readRDS('./data/curatedMetagenomicData/mpse.ibd_IjazUZ_2017.RDS')

sample.da.HallAB <- mpse.ibd.HallAB %>% mp_extract_sample %>%
    dplyr::select(Sample, disease)

sample.da.IjazUZ <- mpse.ibd.IjazUZ %>% mp_extract_sample %>%
    dplyr::select(Sample, disease)
```

```
###
###

mpse.ibd.HallAB %>%
    mp_diff_analysis(
       .abundance = Abundance,
       force = TRUE,
       relative = FALSE,
       .group = disease,
       fc.method = 'compare_mean',
       #ldascore = 3
    ) %>%
    mp_extract_feature() %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    otu.sign.da.ibd.HallAB

mpse.ibd.HallAB %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='geometric.mean'
    ) -> mpse.balance.node.gm.ibd.HallAB

mpse.balance.node.gm.ibd.HallAB %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = disease,
      fc.method = 'compare_mean',
      #ldascore = 3
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    bla.sign.da.gm.ibd.HallAB

mpse.ibd.HallAB %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='mean'
    ) -> mpse.balance.node.mean.ibd.HallAB

mpse.balance.node.mean.ibd.HallAB %>%
    mp_diff_analysis(
```

```
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = disease,
      fc.method = 'compare_mean',
      #ldascore = 3
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    bla.sign.da.mean.ibd.HallAB

mpse.ibd.HallAB %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='median'
    ) -> mpse.balance.node.median.ibd.HallAB

mpse.balance.node.median.ibd.HallAB %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = disease,
      fc.method = 'compare_mean',
      #ldascore = 3
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    bla.sign.da.median.ibd.HallAB

####
####

mpse.ibd.IjazUZ %>%
    mp_diff_analysis(
       .abundance = Abundance,
       force = TRUE,
       relative = FALSE,
       .group = disease,
       fc.method = 'compare_mean',
       #ldascore = 3
    ) %>%
    mp_extract_feature() %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
```

```
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    otu.sign.da.ibd.IjazUZ

mpse.ibd.IjazUZ %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='geometric.mean'
    ) -> mpse.balance.node.gm.ibd.IjazUZ

mpse.balance.node.gm.ibd.IjazUZ %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = disease,
      fc.method = 'compare_mean',
      #ldascore = 3
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    bla.sign.da.gm.ibd.IjazUZ

mpse.ibd.IjazUZ %>%
    mp_balance_clade(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='mean'
    ) -> mpse.balance.node.mean.ibd.IjazUZ

mpse.balance.node.mean.ibd.IjazUZ %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = disease,
      fc.method = 'compare_mean',
      #ldascore = 3
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    bla.sign.da.mean.ibd.IjazUZ

mpse.ibd.IjazUZ %>%
    mp_balance_clade(
```

```
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      pseudonum = 1,
      balance_fun='median'
    ) -> mpse.balance.node.median.ibd.IjazUZ

mpse.balance.node.median.ibd.IjazUZ %>%
    mp_diff_analysis(
      .abundance = Abundance,
      force = TRUE,
      relative = FALSE,
      .group = disease,
      fc.method = 'compare_mean',
      ldascore = 3
    ) %>%
    mp_extract_feature %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, AbundanceBySample) %>%
    tidyr::unnest(AbundanceBySample) %>%
    select(OTU, Sample, Abundance, disease) %>%
    tidyr::pivot_wider(id_cols=c('Sample', 'disease'), values_from=Abundance, names_from=OTU) %>%
    dplyr::mutate_at('disease', as.factor) ->
    bla.sign.da.median.ibd.IjazUZ

make_rf_model <- function(train, test, sample.da, group){
    formula <- as.formula(paste0(group, "~."))
    level <- sample.da %>% dplyr::pull(group) %>% unique()
    formula2 <- as.formula(paste0(group, " ~ ", level[1]))
    mod <- randomForest(formula, data = train)
    res <- predict(mod, test, type='prob') %>%
              tibble::as_tibble(rownames='Sample') %>%
              dplyr::mutate_all(as.vector) %>%
              dplyr::left_join(sample.da, by='Sample') %>%
              pROC::roc(formula2, data=., levels=level, quiet = TRUE) %>%
              magrittr::extract2('auc')

    return (res)
}

make_rf_by_random_sample <- function(dat, prob=2/3, sample.da, group){
    train <- dat %>%
    dplyr::group_split(!!rlang::sym(group)) %>%
    lapply(function(x)x[sample(nrow(x), size=prob * nrow(x)),]) %>%
    dplyr::bind_rows() %>%
    tibble::column_to_rownames(var='Sample')
    test <- dat %>% dplyr::filter(!Sample %in% rownames(train)) %>%
        tibble::column_to_rownames(var='Sample')
    res <- make_rf_model(train = train, test = test, sample.da=sample.da, group = group)
    return (res)
}

otu.auc.ibd.HallAB <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=otu.sign.da.ibd.HallAB, prob = 1/2, sample.da.HallAB, group='disease')
   )
)

bla.auc.ibd.HallAB <- withr::with_seed(123,
    replicate(100,
```

```
      make_rf_by_random_sample(dat=bla.sign.da.gm.ibd.HallAB, prob = 1/2, sample.da.HallAB, group='disease')
   )
)

bla.auc.mean.ibd.HallAB <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=bla.sign.da.mean.ibd.HallAB, prob = 1/2, sample.da.HallAB, group='disease')
    )
)

bla.auc.median.ibd.HallAB <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat = bla.sign.da.median.ibd.HallAB, prob = 1/2, sample.da.HallAB, group='diseas
    )
)

###
#
###

otu.auc.ibd.IjazUZ <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=otu.sign.da.ibd.IjazUZ, prob = 1/2, sample.da.IjazUZ, group='disease')
   )
)

bla.auc.ibd.IjazUZ <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=bla.sign.da.gm.ibd.IjazUZ, prob = 1/2, sample.da.IjazUZ, group='disease')
   )
)

bla.auc.mean.ibd.IjazUZ <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=bla.sign.da.mean.ibd.IjazUZ, prob = 1/2, sample.da.IjazUZ, group='disease')
    )
)

bla.auc.median.ibd.IjazUZ <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat = bla.sign.da.median.ibd.IjazUZ, prob = 1/2, sample.da.IjazUZ, group='diseas
    )
)

###
# CD in example1
###
otu.auc.cd <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=otu.sign.da, prob = 1/2, sample.da.CD, group='Group')
   )
)

bla.auc.cd <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat=bla.sign.da, prob = 1/2, sample.da.CD, group='Group')
   )
)

bla.auc.mean.cd <- withr::with_seed(123,
```

```
    replicate(100,
      make_rf_by_random_sample(dat=bla.sign.da.mean, prob = 1/2, sample.da.CD, group='Group')
    )
)

bla.auc.median.cd <- withr::with_seed(123,
    replicate(100,
      make_rf_by_random_sample(dat = bla.sign.da.median, prob = 1/2, sample.da.CD, group='Group')
    )
)

dd.cd <- data.frame(
        SignOTU = otu.auc.cd,
        SignBalance.gm = bla.auc.cd,
        SignBalance.mean = bla.auc.mean.cd,
        SignBalance.median = bla.auc.median.cd
      ) %>%
      dplyr::mutate(study='CD') %>%
      tidyr::pivot_longer(cols=!study, names_to = 'Type', values_to = 'AUC')


dd.ibd.HallAB <- data.frame(
        SignOTU = otu.auc.ibd.HallAB,
        SignBalance.gm = bla.auc.ibd.HallAB,
        SignBalance.mean = bla.auc.mean.ibd.HallAB,
        SignBalance.median = bla.auc.median.ibd.HallAB
      ) %>%
      dplyr::mutate(study='IBD.HallAB') %>%
      tidyr::pivot_longer(cols=!study, names_to = 'Type', values_to = 'AUC')

dd.ibd.IjazUZ <- data.frame(
        SignOTU = otu.auc.ibd.IjazUZ,
        SignBalance.gm = bla.auc.ibd.IjazUZ,
        SignBalance.mean = bla.auc.mean.ibd.IjazUZ,
        SignBalance.median = bla.auc.median.ibd.IjazUZ
      ) %>%
      dplyr::mutate(study='IBD.IjazUZ') %>%
      tidyr::pivot_longer(cols=!study, names_to = 'Type', values_to = 'AUC')

dd <- dplyr::bind_rows(dd.cd, dd.ibd.HallAB, dd.ibd.IjazUZ)


comparelist <- dd %>%
               pull(Type) %>%
               unique() %>%
               utils::combn(2) %>%
               apply(2, list) %>%
               unlist(recursive = FALSE)

Sign.Balance.AUC.p <- dd %>% #dplyr::filter(study != "CD") %>%
    ggplot(aes(x=Type, y=AUC, fill=Type)) +
     geom_boxplot() +
     geom_jitter(color='grey', width=.2, show.legend = FALSE, alpha = .5) +
     ggsignif::geom_signif(
       comparisons = comparelist,
       test = 'wilcox.test',
       step_increase = .05,
       textsize = 3,
       size = .25,
       tip_length = .01
```

```
    ) +
    scale_fill_manual(values = c("#264653", "#2A9D8F", "#E9C46A", "#F4A261")) +
    theme_bw() +
    xlab(NULL) +
    facet_grid(~study) +
    theme(
      strip.background.x = element_rect(color = NA, fill = 'grey'),
      strip.text.x = element_text(face = 'bold'),
      legend.position = c(.86, .2),
      axis.text.x = element_blank(),
      axis.ticks.x = element_blank()
    )
Sign.Balance.AUC.p
```

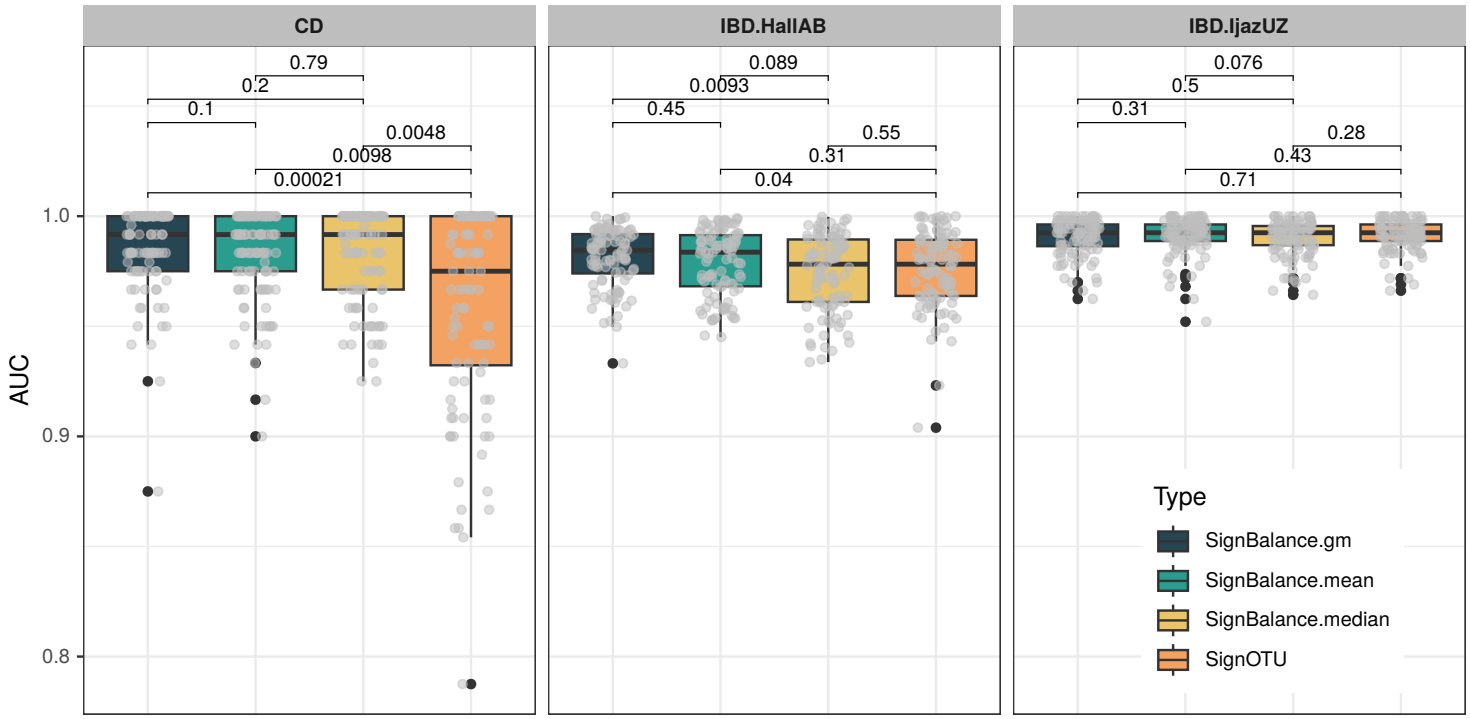

Fig. SB.1: **The AUC boxplot of different study based on the significant differential clades and OTUs**. We found the performance of the models built based on significant differential clades were almost better than the model based on significant differential OTUs.

2 The comparison of KEGG enrichment results based on the results of common methods of differential analysis.

```
mpseKO <- readRDS('./data/CD_RF_microbiome/mpse_KO.rds')
mp.xx <- mpseKO %>%
    mp_extract_feature() %>%
    dplyr::filter(!is.na(Sign_disease)) %>%
    select(OTU, Sign_disease)
##### limma ############
mpseKO %>% test_differential_abundance(
     .formula = ~disease,
     method = 'limma_voom',
     scaling_method = 'none',
  ) %>%
  mp_extract_feature() %>%
  dplyr::filter(P.Value <= 0.05) %>%
```

```
  dplyr::mutate(Sign_disease = ifelse(logFC>0, 'CN', 'CD')) %>%
  select(OTU, Sign_disease) -> limma.xx
### edgeR ######
mpseKO %>% test_differential_abundance(
     .formula = ~disease,
     method = 'edgeR_quasi_likelihood',
     scaling_method = 'none'
     ) %>%
     mp_extract_feature() %>%
     dplyr::filter(PValue <= .05) %>%
     dplyr::mutate(Sign_disease = ifelse(logFC>0, 'CN', 'CD')) %>%
     select(OTU, Sign_disease) -> edgeR.xx
sample.da.KO <- mpseKO %>% mp_extract_sample %>% tibble::column_to_rownames(var='Sample')
dat <- mpseKO %>% mp_extract_assays(.abundance = Abundance, byRow=F)
### kruskal.test #########
kwres <- multi_compare(fun="kruskal.test", data=merge(dat, sample.da.KO, by=0),
                       feature=colnames(dat), factorNames = "disease")
resp <- unlist(lapply(kwres,function(x)x$p.value))
kwres <- data.frame(f=colnames(dat), pvalue=resp)
kw.xx <- data.frame(OTU = as.vector(kwres[kwres$pvalue<=0.05, 1]))
total.sign <- mpseKO %>%
    mp_extract_abundance() %>%
    tidyr::unnest(AbundanceBySample) %>%
    dplyr::group_by(disease, label) %>%
    dplyr::summarize(meanAbu=mean(Abundance)) %>%
    dplyr::mutate_at("label", as.character) %>%
    tidyr::pivot_wider(id_cols=label, names_from=disease, values_from=meanAbu) %>%
    tibble::column_to_rownames(var='label') %>%
    #magrittr::extract(kw.xx$OTU, ) %>%
    apply(., 1, function(x)ifelse(x[[1]] > x[[2]], 'CD', "CN"))
kw.xx$Sign_disease <-
    total.sign %>% magrittr::extract(kw.xx$OTU) %>%
    base::unname()
### metagenomeSeq ########
library(metagenomeSeq)
saa <- AnnotatedDataFrame(sample.da.KO)
nMR <- newMRexperiment(t(dat), phenoData=saa)
nMR <- cumNorm(nMR, p = 0.5)
nMRpd <- pData(nMR)
mod <- model.matrix(~1 + disease, data = nMRpd)
res5 <- fitFeatureModel(nMR, mod)
ms.xx <- MRcoefs(res5) %>%
         tibble::as_tibble(rownames = 'OTU') %>%
         filter(pvalues <= .05) %>%
         dplyr::mutate(Sign_disease = ifelse(logFC>0, 'CN', 'CD')) %>%
         select(OTU, Sign_disease)
######## LEfSe ############
sample.da.KO %<>% select(disease)
lefseda <- merge(sample.da.KO, dat, by=0) %>%
           select(-Row.names)
lefseda <- data.frame(t(lefseda), check.names=FALSE) %>% tibble::rownames_to_column(var="feature")
tmpfile1 <- tempfile()
tmpfile2 <- paste0(tmpfile1, ".format")
outfile <- paste0(tmpfile1, ".out")
write.table(lefseda, tmpfile1, row.names=FALSE, col.names=FALSE, quote=FALSE, sep="\t")
CMD1 <- paste("format_input.py", tmpfile1, tmpfile2, "-c 1 -o 1000000 ", sep=" ")
CMD2 <- paste("run_lefse.py", tmpfile2, outfile, "--min_c 3 -f 1 -b 1 -l 2 -y 1", sep=" ")
system(CMD1)
system(CMD2)
lefseout <- read.table(outfile, sep="\t", header=F, row.names=1)
```

```
flags <- suppressWarnings(ifelse(is.na(as.numeric(lefseout$V5) <= 0.05),FALSE,TRUE))
lefse.xx <- data.frame(OTU = rownames(lefseout[flags, ]),
                       Sign_disease = total.sign %>%
                            magrittr::extract(rownames(lefseout[flags, ])) %>%
                            base::unname()
            )
### ANCOMBC ###############
library(ANCOMBC)
psKO <- mpseKO %>% as.phyloseq()
ANCOMBC.xx <- ancombc(psKO, formula = 'disease',
                      p_adj_method = "holm",
                      lib_cut = 0,
                      group = "disease",
                      struc_zero = FALSE,
                      neg_lb = FALSE,
                      tol = 1e-5,
                      max_iter = 100,
                      conserve = FALSE,
                      alpha = 0.05,
                      global = FALSE)
signs.AB <- ANCOMBC.xx$res$p_val %>% dplyr::filter(diseaseCN<=0.05) %>% rownames()
ANCOMBC.xx <- data.frame(
           OTU = signs.AB,
           Sign_disease = total.sign %>%
               magrittr::extract(signs.AB) %>%
               base::unname())
### combine all the differential results
total.xx <- dplyr::bind_rows(list(mp_diff_analysis = mp.xx,
                                  Limma = limma.xx,
                                  edgeR = edgeR.xx,
                                  `kruskal-test` = kw.xx,
                                  metagenomeSeq = ms.xx,
                                  LEfSe = lefse.xx,
                                  ANCOMBC = ANCOMBC.xx
                                  ),
                             .id = 'Method') %>%
            dplyr::distinct()
enrich.res <- compareCluster(OTU ~ Sign_disease + Method, data = total.xx, fun = enrichKO)
compare.enrich.p <- dotplot(enrich.res, x = 'Sign_disease') +
  facet_grid(~Method) +
  xlab(NULL) +
  scale_color_gradientn(
     colours = c("#b3eebe", "#46bac2", "#371ea3"),
     guide = guide_colorbar(reverse=TRUE, order=1)
   ) +
   labs(x = NULL) +
   guides(size = guide_legend(override.aes=list(shape=1))) +
   theme(
     axis.text.x = element_text(angle = -45, hjust=0),
     strip.text.x = element_text(face = 'bold', size = 9),
     strip.background.x = element_rect(fill = 'grey', color = NA),
     panel.grid.major.y = element_line(linetype='dotted', color='#808080'),
     panel.grid.major.x = element_blank()
   )
```

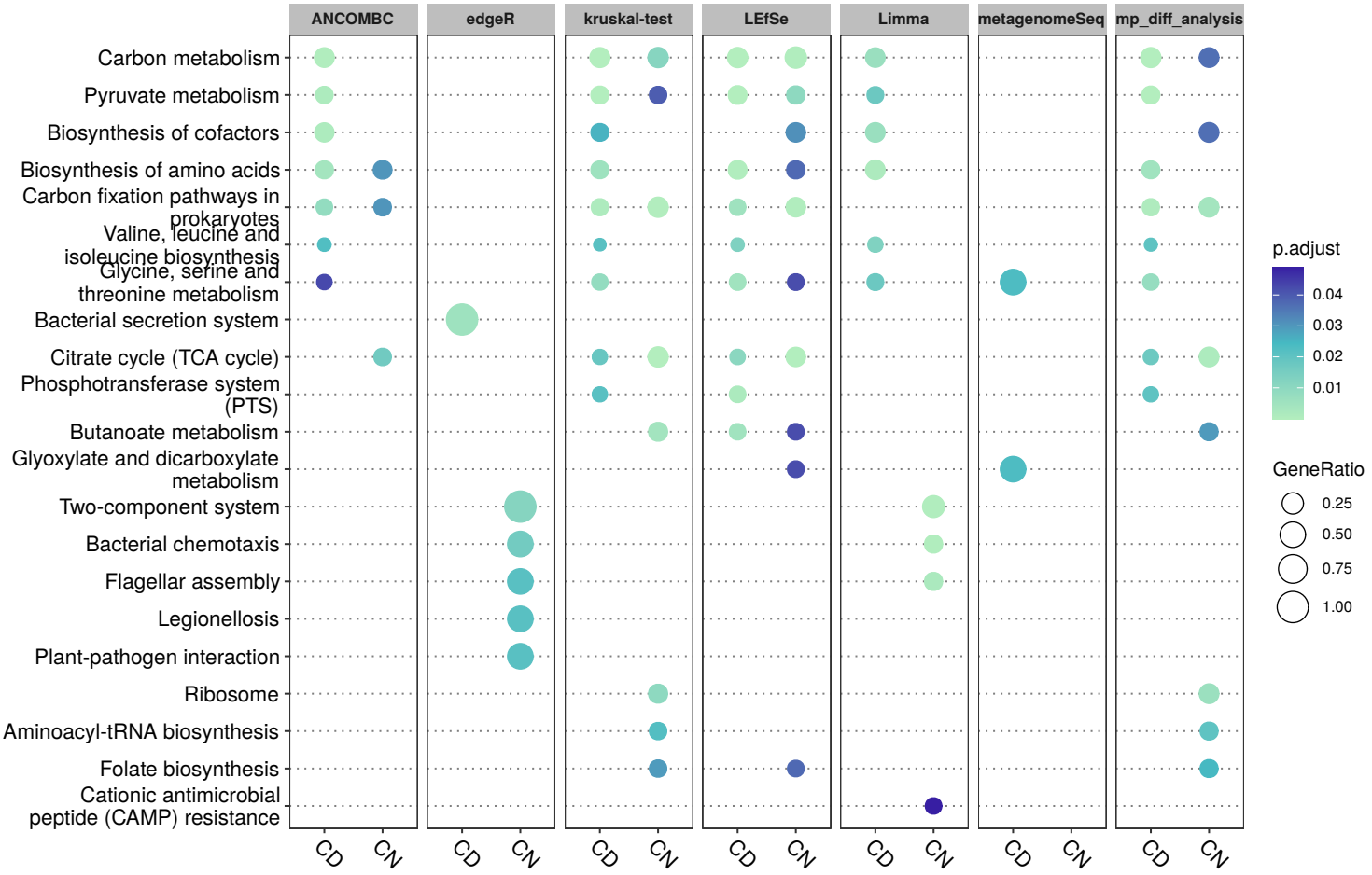

Fig. SB.2: **The comparison of KEGG enrichment results based on the differential genes identified by different methods**

3 The simulation results of biomarker discovery using *mp_diff_analysis* of MicrobiotaProcess

To estimate the performance of *mp_diff_analysis* and other tools for microbiome census data, we constructed a collection of simulated datasets based on lognormal, and normal distributions. To evaluate the sample size, half of the datasets with different distributions have 100 samples and 1000 features, and the other half have 20 samples and 1000 features. All samples in the dataset are divided into 2 categories, and each category is further divided into two sub-categories. At the same time, there are 1000 features in each dataset, half of which are sampled from a specified distribution with different mean between two categories (group) (denoted as positive features), and the other half are from the same mean sampled from the specified distribution (denoted as negative features). Since the mean and variance of the features might be affected the sensitivity and specificity of the detection, positive features are constructed using a range of different means and standard deviations (Please refer to the detailed description below). The false positive rate (FPR) and false negative rate (FNR) were used as methods for evaluating results. False positive rate is the number of erroneously detected positive features divided by the total number of positive features. False negative rate is the number of erroneously detected negative features divided by the total number of negative features. The lower FPR and FNR, the better the performance. The source codes are available from the repository[1]

3.1 Simulation datasets from lognormal distribution

The datasets were generated using *rlnorm* of *R*. **(1)** Negative features of first collections have the same $meanlog = 10$ and $sdlog = 1$, whereas the positive features of one class have the $meanlog = 10$ - i ($sdlog = 1$) and the other $meanlog = 10 + i$ ($sdlog = 1$), where i is a parameter ranging from 0.01 to 3 ($step = 0.01$, each $step$ will generate a dataset). **(2)** All features of second collections have the same $sdlog = s$, where s is also a parameter ranging from 1 to 3 ($step = 0.01$, each $step$ will generate a dataset), whereas the $meanlog$ of negative features is 10, positive features of one class is 8, the other is 12. **(3)** The subclass distribution of negative class is different in the third collection. In details, the $meanlog$ of features is equal between

[1] https://github.com/YuLab-SMU/MP_supplementary_file/tree/main/supplemental_fileB_codes

the second subclass of first class and the first subclass of the second class. However, the *meanlog* of features is different in other two subclasses (*meanlog* = 10 - *i* and *meanlog* = 10 + *i*, *i* = 2, *step* = 0.01, each *step* also will generate a dataset). But the features should be considered the negative features, since the difference is inconsistent between the subclass. And the way of positive features defined is the same with second collections (*meanlog* = 10 - *i* and *meanlog* = 10 + *i*, *i* = 2, and *sdlog* = *s*, where *s* is parameter ranging from 1 to 3 (*step* = 0.01, each *step* will generate a dataset)). The defined methods of the other three collections containing 20, 40, 60, 80 samples are similar to the methods of 100 samples.

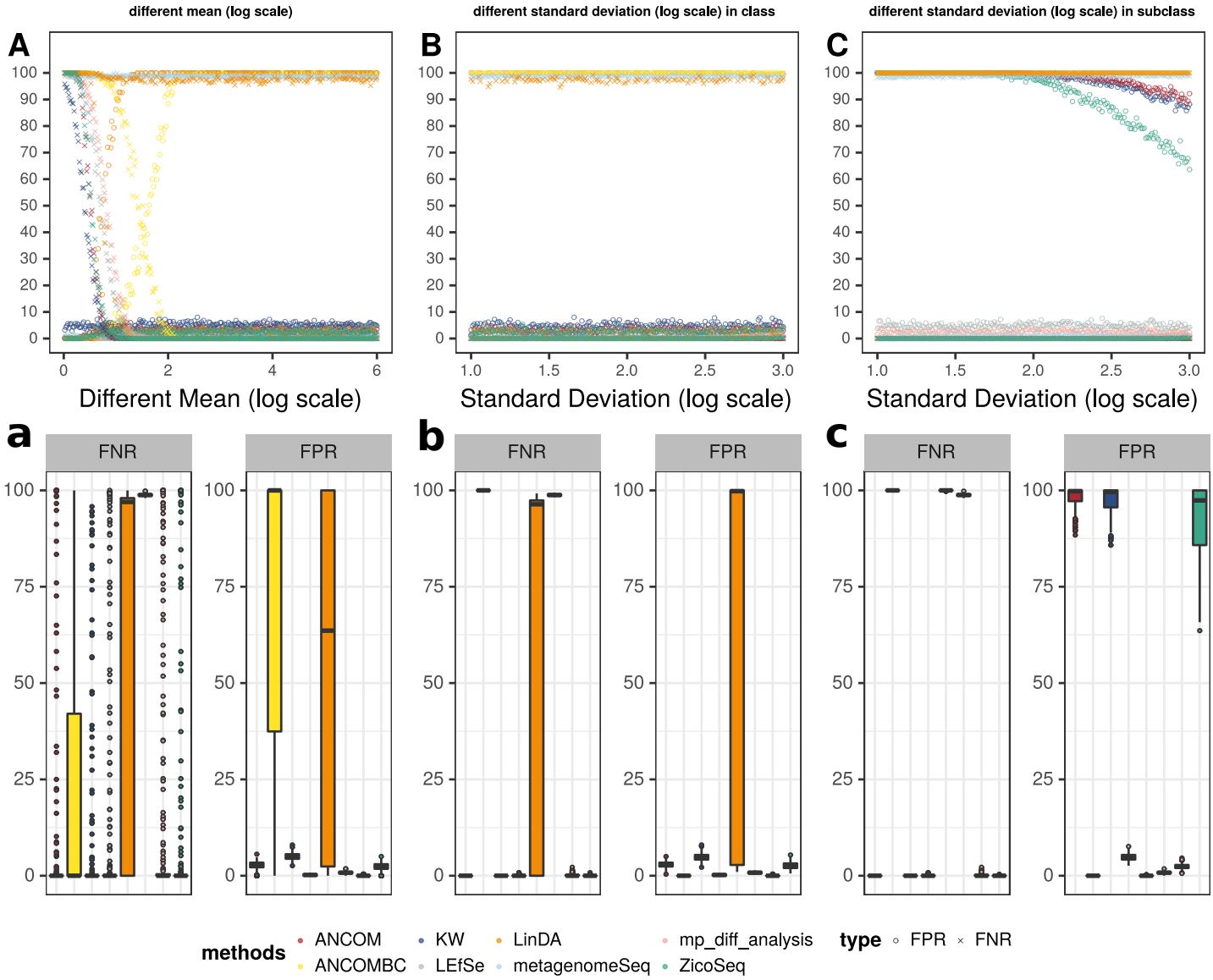

Fig. SB.3: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (100 samples) based on lognormal distribution.** (A and a) the false negative rates of *ANCOM*, *mp_diff_analysis*, *Kruskal-Wallis rank sum test* and *LEfSe* was decreased with the increasing values of difference between classes mean (log scale). Whereas *mp_diff_analysis* and *LEfSe* has better control of false positive rate. (B and b) the *mp_diff_analysis* had better false positive and negative rate. *ANCOM* and *Kruskal-Wallis rank sum test* had higher false positive rate with increasing values of standard deviation (log scale) between classes. (C and c) the *mp_diff_analysis* also had better control of false positive and negative rate with increasing of standard deviation within inconsistent subclasses.

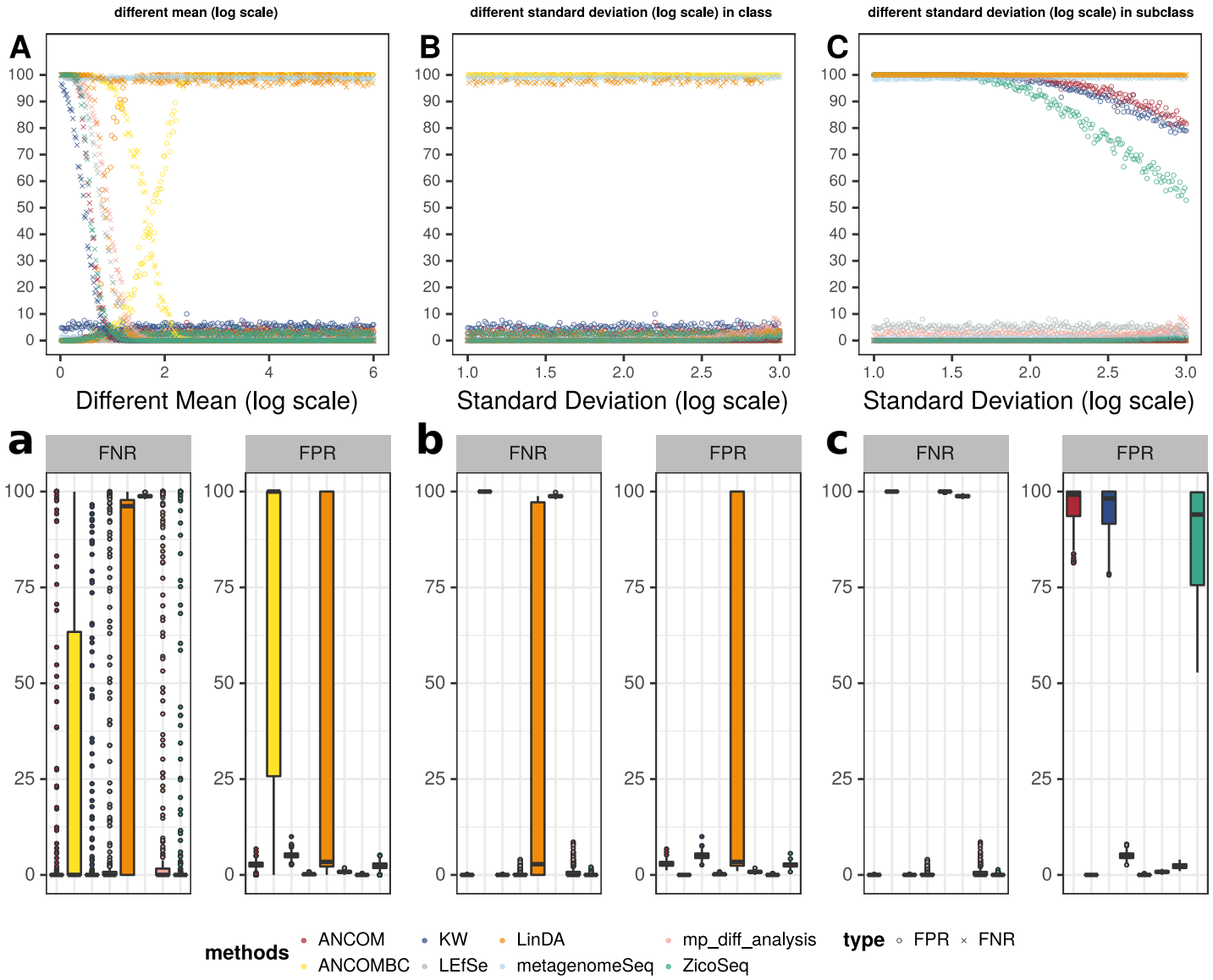

Fig. SB.4: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (80 samples) based on lognormal distribution.**

3.2 Simulation datasets from normal distribution

The datasets were simulated using *norm* of *R*. **(1)** Negative features of first collections have the same *mean* = 10000 and *sd* = 100, whereas the positive features of one class have the *mean* = 10000 - *i* (*sd* = 100) and the other *mean* = 10000 + i (sd = 100), where *i* is a parameter ranging from 1 to 150 (*step*=1, each *step* will generate a dataset). **(2)** All features of second collections have the same *sd* = *s*, where *s* is also a parameter ranging from 100 to 2050 (*step*=10, each *step* will generate a dataset), whereas the mean of negative features is 10000, positive features of one class are 9000, the other is 11000. **(3)** The subclass distribution of the negative class is different in the third collection. In detail, the mean of features is equal between the second subclass of first class and the first subclass of the second class. However, the mean of features is different in other two subclasses (*mean* = 10000 - *i* and *mean* = 10000 + *i*, *i* =1000). But the features should be considered the negative features, since the difference is inconsistent between the subclass. And the way of positive features defined is the same with second collections (*mean* = 10000 - *i* and *meanlog* = 10000 + *i*, *i* = 1000, and *sd* = *s*, where *s* is parameter ranging from 100 to 2050 (*step* = 10, each *step* will generate a dataset)). The defined methods of the other three collections containing 20, 40, 60, 80 samples are similar to the methods of 100 samples

The *mp_diff_analysis* of *MicrobiotaProcess* achieved a better false positive rate compared with other methods. It was more robust at different distribution and different condition (different mean and different standard deviation).

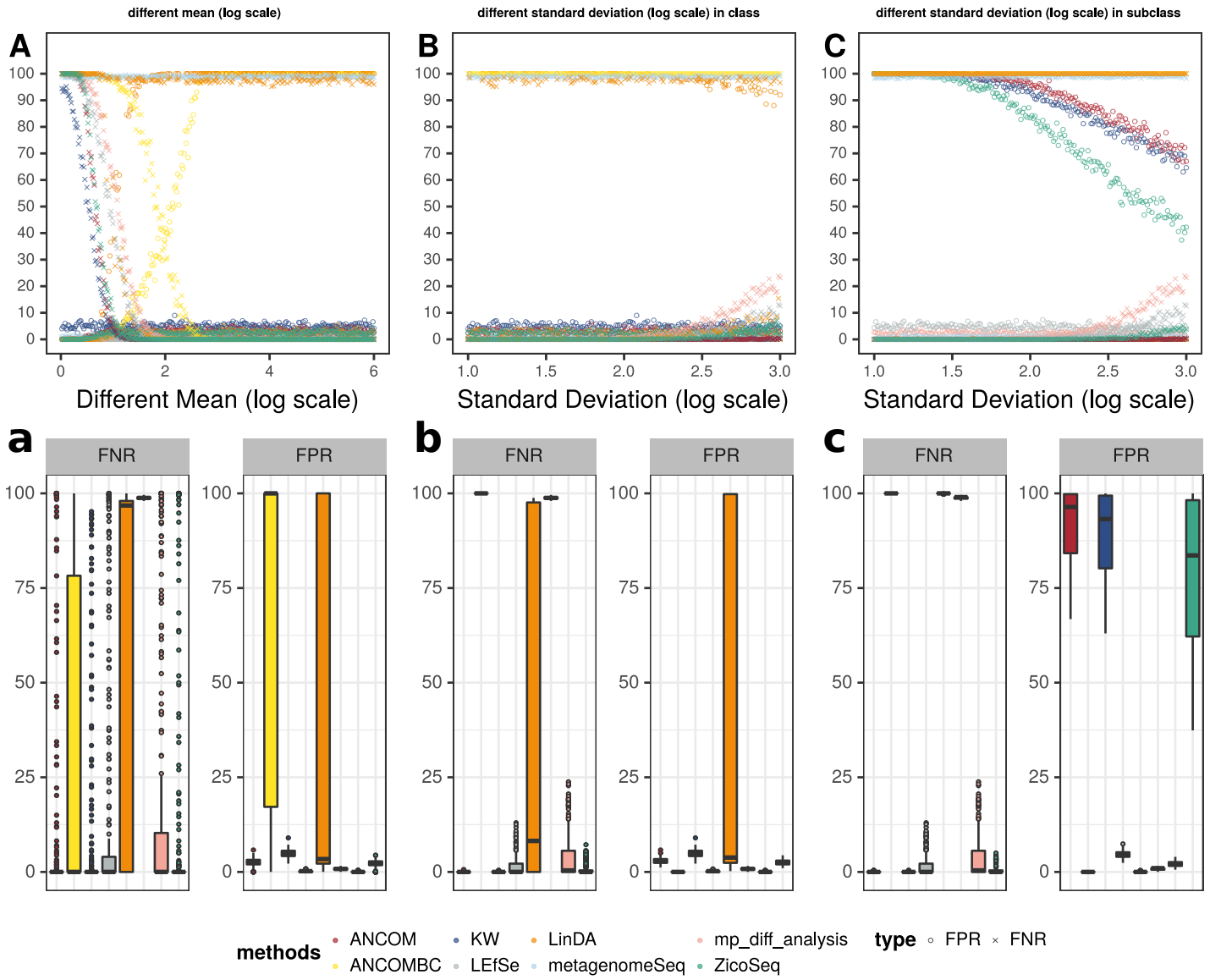

Fig. SB.5: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (60 samples) based on lognormal distribution.**

4 The comparison between the MicrobiotaProcess and other DAA (differential abundance analysis) tools on real data.

We had performed the additional evaluations of the *mp_diff_analysis* in *MicrobiotaProcess* and other DAA tools (using pvalue or correction pvalue to identify significant features) across 10 two-group 16S rRNA datasets (Schubert et al. 2014; Yurgel et al. 2017; Scher et al. 2013; Charlson 2010; Kesy et al. 2019; Turnbaugh et al. 2009; Morgan et al. 2012; Baxter et al. 2016; Douglas et al. 2018), which were from soil, marine plastic and human stool environment collected in the article (Zhou et al. 2022; Nearing et al. 2022). The features that found in fewer than 10% of samples were removed in each dataset, and the relative abundance table was used to detect the significant features. As expected, the results used pvalue and correction pvalue to identify were separated among all the methods. We found that significant ASV (OTU) detected by our methods tended to be also detected by other tools (Fig. SB.13). Our method tended to identify a relatively small number of significant ASV (Fig. SB.13, SB.14 and SB.15). In addition, we also found our method (using Bonferroni correction and pvalue modes) is similar to the p-value correction result of LEfSe output (Fig. SB.13) (Bonferroni correction method, which might be stricter than the others, but we can not obtain the pvalue of all features to correct with FDR correction method owing to the design of the tool). This hint our method might be more conservative and better precision but with a possible loss of sensitivity. This also was consistent with result of simulation.

Baxter, Nielson T., Mack T. Ruffin, Mary A. M. Rogers, and Patrick D. Schloss. 2016. "Microbiota-Based Model Improves the Sensitivity of Fecal Immunochemical Test for Detecting Colonic Lesions." *Genome Medicine* 8 (1): 37. https://doi.org/10.1186/s13073-016-0290-3.

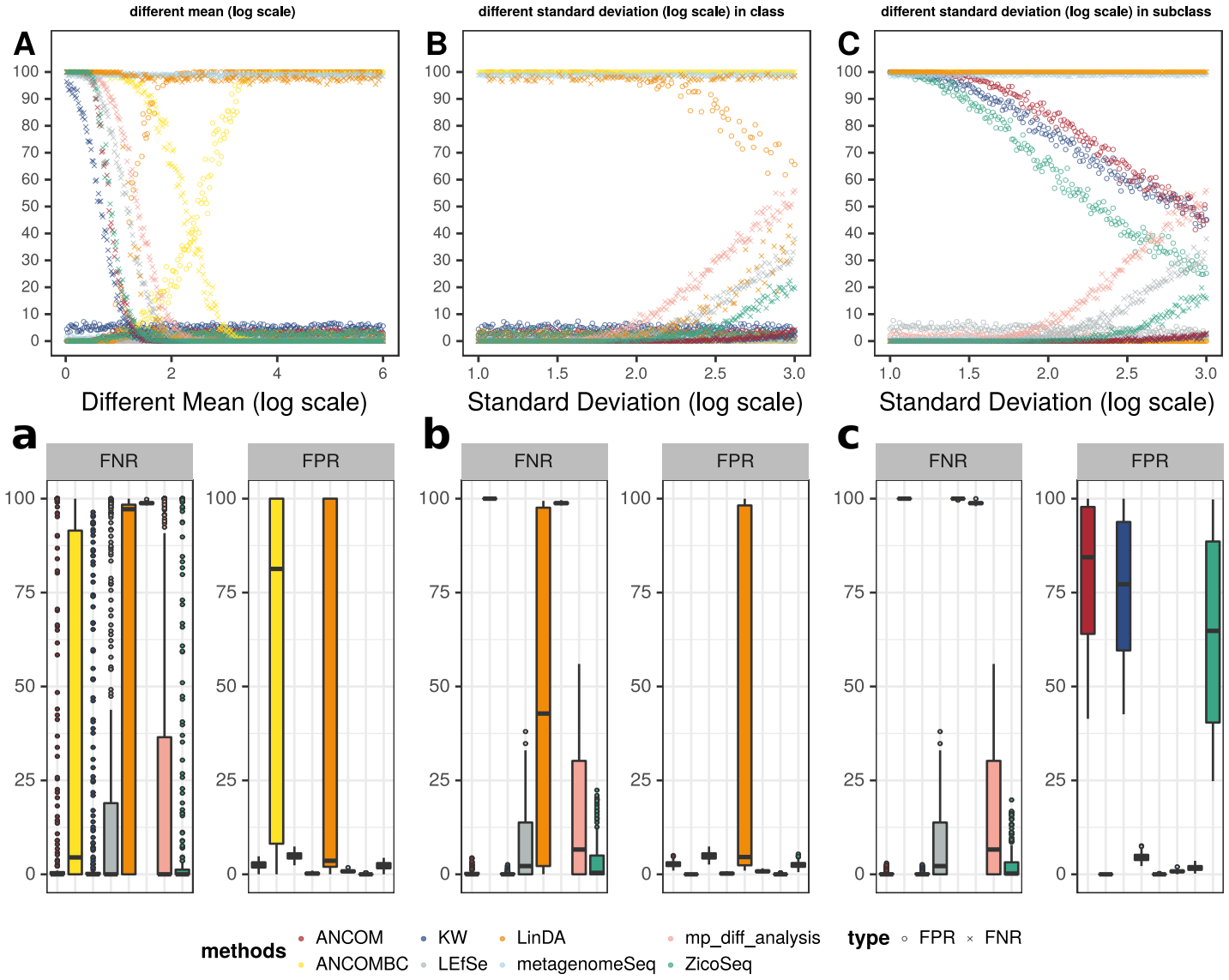

Fig. SB.6: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (40 samples) based on lognormal distribution.**

Charlson, Jun AND Custers-Allen, Emily S. AND Chen. 2010. "Disordered Microbial Communities in the Upper Respiratory Tract of Cigarette Smokers." *PLOS ONE* 5 (12): 1–10. https://doi.org/10.1371/journal.pone.0015216.

Douglas, Gavin M., Richard Hansen, Casey M. A. Jones, Katherine A. Dunn, AndrÃ©M. Comeau, Joseph P. Bielawski, Rachel Tayler, et al. 2018. "Multi-Omics Differentially Classify Disease State and Treatment Outcome in Pediatric Crohn's Disease." *Microbiome* 6 (1): 13. https://doi.org/10.1186/s40168-018-0398-3.

Hall, Andrew Brantley, Moran Yassour, Jenny Sauk, Ashley Garner, Xiaofang Jiang, Timothy Arthur, Georgia K. Lagoudas, et al. 2017. "A Novel Ruminococcus Gnavus Clade Enriched in Inflammatory Bowel Disease Patients." *Genome Medicine* 9 (1): 103. https://doi.org/10.1186/s13073-017-0490-5.

Ijaz, Umer Zeeshan, Christopher Quince, Laura Hanske, Nick Loman, Szymon T. Calus, Martin Bertz, Christine A. Edwards, et al. 2017. "The Distinct Features of Microbial 'Dysbiosis' of Crohn's Disease Do Not Occur to the Same Extent in Their Unaffected, Genetically-Linked Kindred." *PLOS ONE* 12 (2): 1–13. https://doi.org/10.1371/journal.pone.0172605.

Kesy, Katharina, Sonja Oberbeckmann, Bernd Kreikemeyer, and Matthias Labrenz. 2019. "Spatial Environmental Heterogeneity Determines Young Biofilm Assemblages on Microplastics in Baltic Sea Mesocosms." *Frontiers in Microbiology* 10. https://doi.org/10.3389/fmicb.2019.01665.

Morgan, Xochitl C., Timothy L. Tickle, Harry Sokol, Dirk Gevers, Kathryn L. Devaney, Doyle V. Ward, Joshua A. Reyes, et al. 2012. "Dysfunction of the Intestinal Microbiome in Inflammatory Bowel Disease and Treatment." *Genome Biology* 13 (9): R79. https://doi.org/10.1186/gb-2012-13-9-r79.

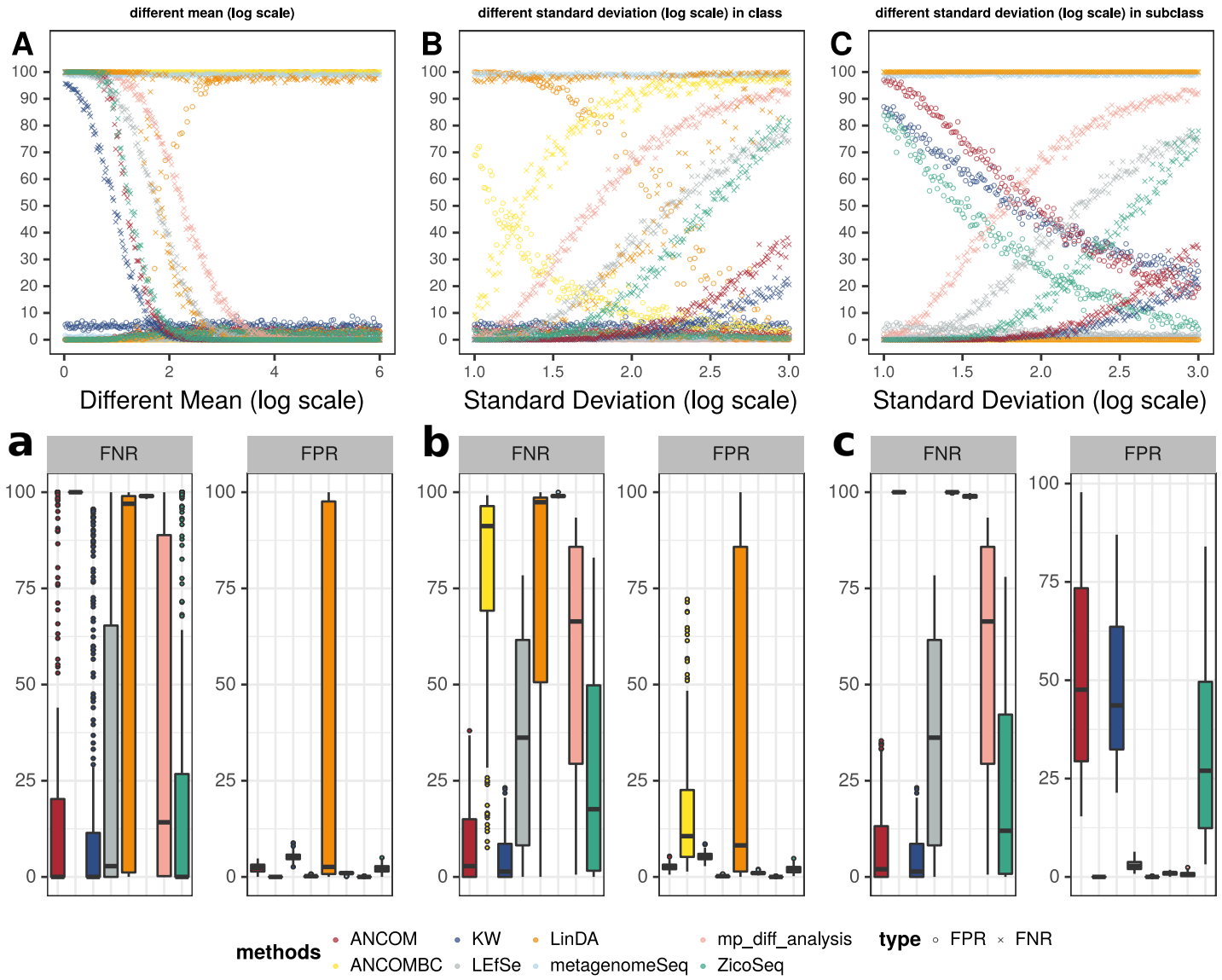

Fig. SB.7: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (20 samples) based on log normal distribution.** The *mp_diff_analysis* had better false positive rate at the price of power decrease of test, it was robust at different condition. (A and a) at different mean of log scale. (B and b) at different standard deviation of log scale in different group. (C and c) at different standard deviation of log scale in different subgroup. Although *LEfSe* had better false negative rate than *mp_diff_analysis*, *mp_diff_analysis* achieved better false positive rate.

Nearing, Jacob T., Gavin M. Douglas, Molly G. Hayes, Jocelyn MacDonald, Dhwani K. Desai, Nicole Allward, Casey M. A. Jones, et al. 2022. "Microbiome Differential Abundance Methods Produce Different Results Across 38 Datasets." *Nature Communications* 13 (1): 342. https://doi.org/10.1038/s41467-022-28034-z.

Scher, Jose U, Andrew Sczesnak, Randy S Longman, Nicola Segata, Carles Ubeda, Craig Bielski, Tim Rostron, et al. 2013. "Expansion of Intestinal *Prevotella Copri* Correlates with Enhanced Susceptibility to Arthritis." Edited by Diane Mathis. *eLife* 2 (November): e01202. https://doi.org/10.7554/eLife.01202.

Schubert, Alyxandria M., Mary A. M. Rogers, Cathrin Ring, Jill Mogle, Joseph P. Petrosino, Vincent B. Young, David M. Aronoff, and Patrick D. Schloss. 2014. "Microbiome Data Distinguish Patients with Clostridium Difficile Infection and Non-c. Difficile-Associated Diarrhea from Healthy Controls." *mBio* 5 (3): e01021–14. https://doi.org/10.1128/mBio.01021-14.

Turnbaugh, Peter J., Micah Hamady, Tanya Yatsunenko, Brandi L. Cantarel, Alexis Duncan, Ruth E. Ley, Mitchell L. Sogin, et al. 2009. "A Core Gut Microbiome in Obese and Lean Twins." *Nature* 457 (7228): 480–84. https://doi.org/10.1038/nature07540.

Yurgel, Svetlana N., Gavin M. Douglas, André M. Comeau, Melissa Mammoliti, Ashley Dusault, David Percival, and Morgan G. I. Langille. 2017. "Variation in Bacterial and Eukaryotic Communities Associated with Natural and Managed Wild

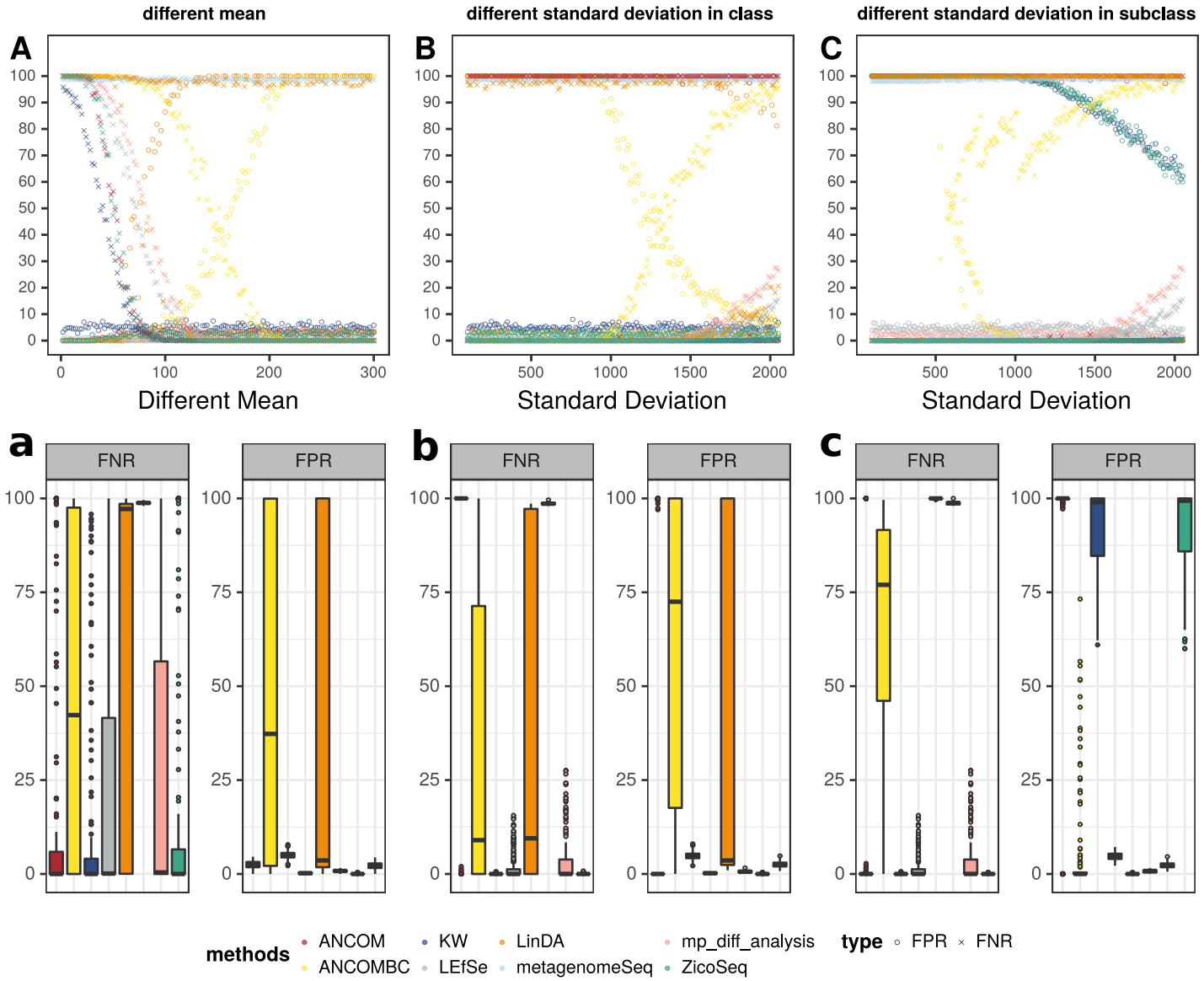

Fig. SB.8: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (100 samples) based on normal distribution**. We found the *mp_diff_analysis* also had better false positive rate at the price of some detection capabilities. (A and a) at different mean. (B and b) at different standard deviation in different group. (C and c) at different standard deviation in different subgroup.

Blueberry Habitats." *Phytobiomes Journal* 1 (2): 102–13. https://doi.org/10.1094/PBIOMES-03-17-0012-R.

Zhou, Huijuan, Kejun He, Jun Chen, and Xianyang Zhang. 2022. "LinDA: Linear Models for Differential Abundance Analysis of Microbiome Compositional Data." *Genome Biology* 23 (1): 95. https://doi.org/10.1186/s13059-022-02655-5.

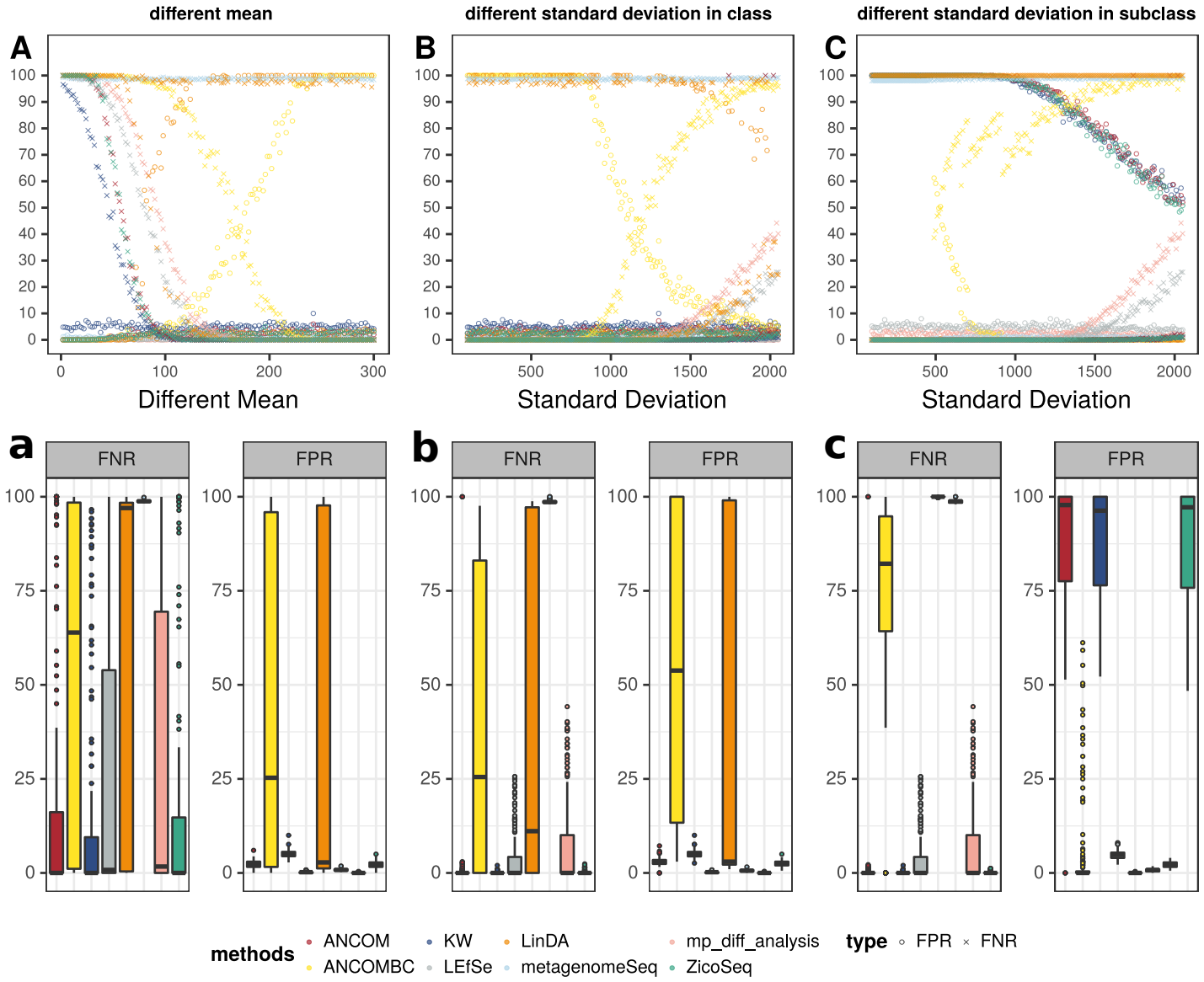

Fig. SB.9: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (80 samples) based on normal distribution.**

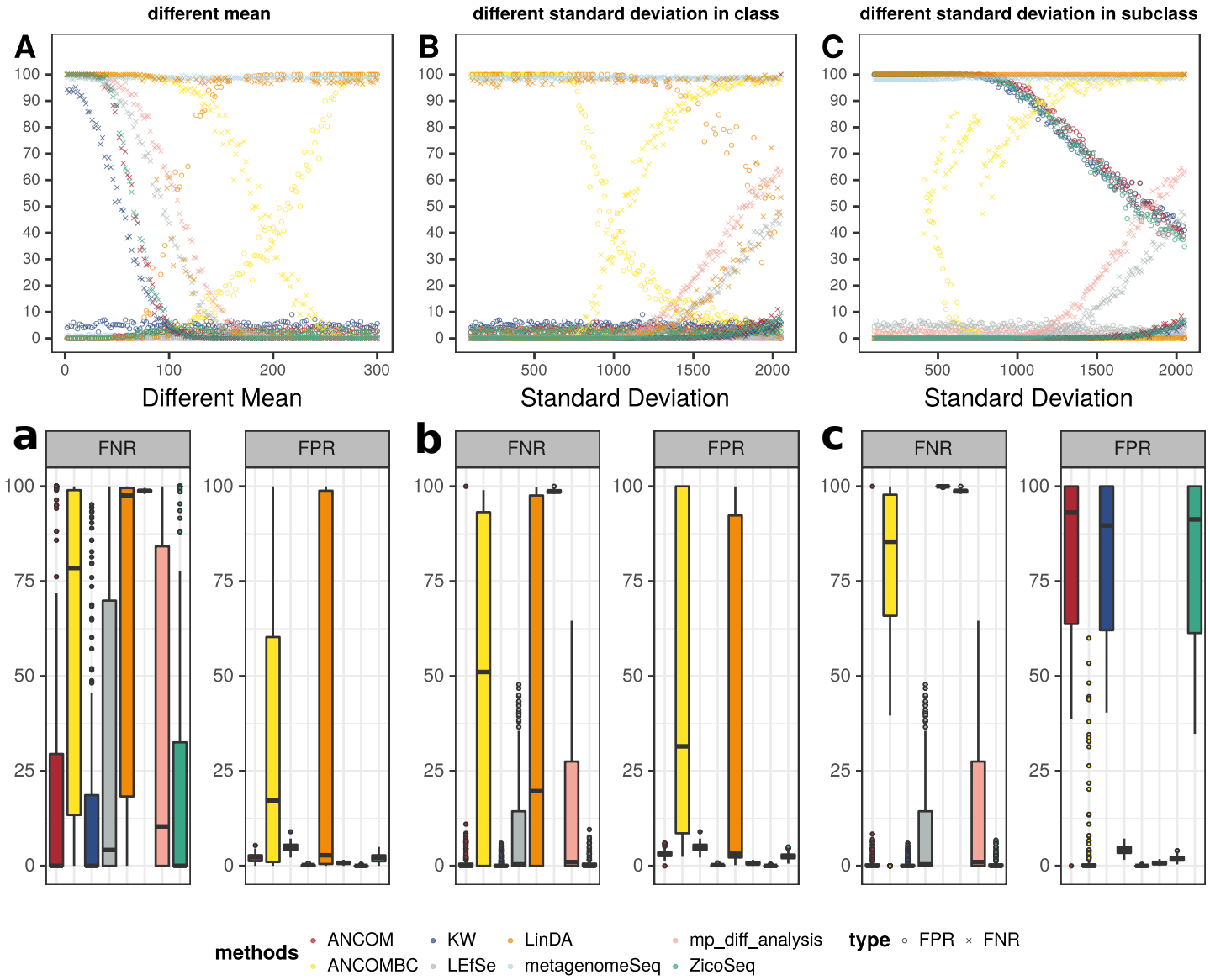

Fig. SB.10: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (60 samples) based on normal distribution.**

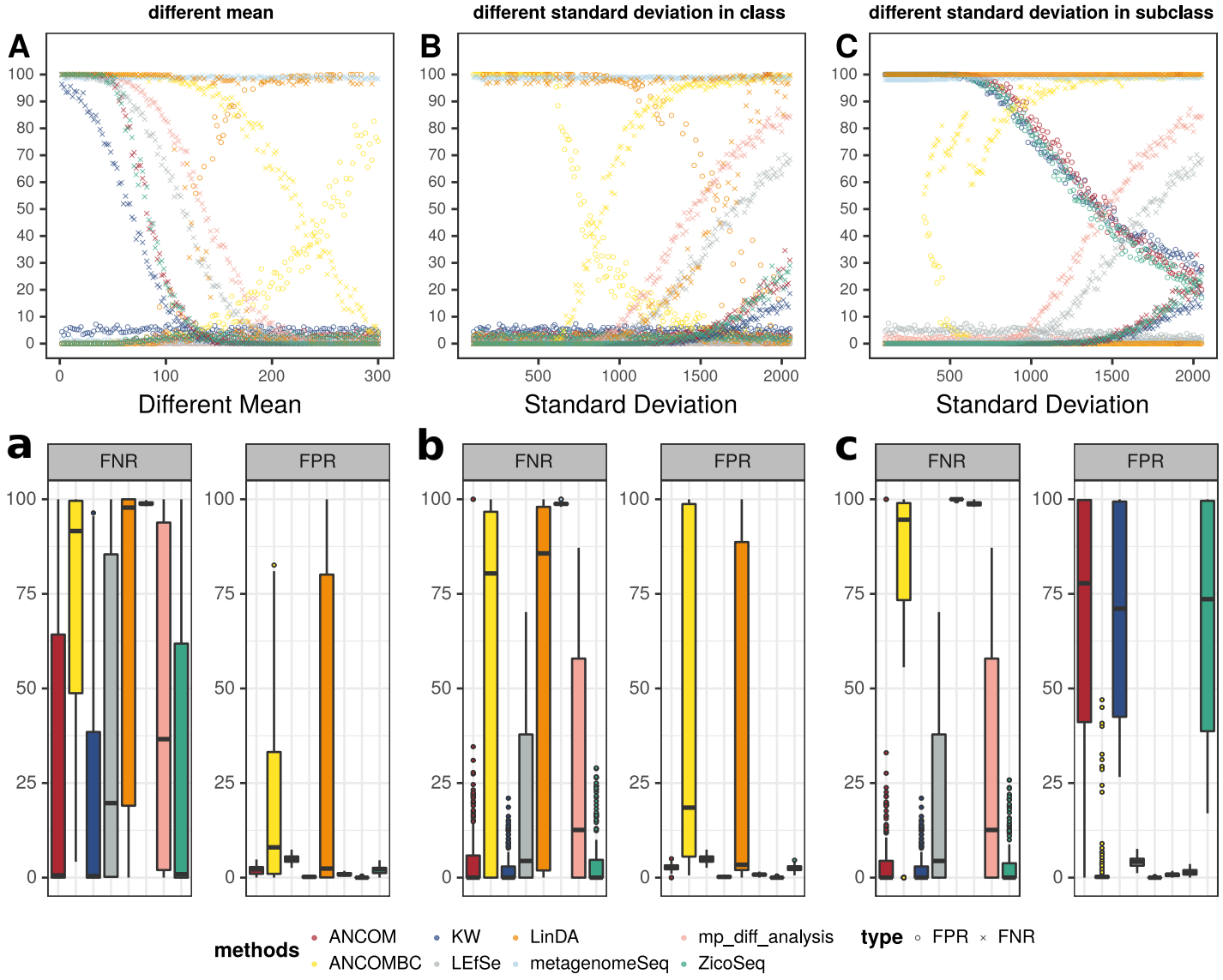

Fig. SB.11: **Comparison between MicrobiotaProcess and other common methods for false positive and negative rates in simulation datasets (40 samples) based on normal distribution.**

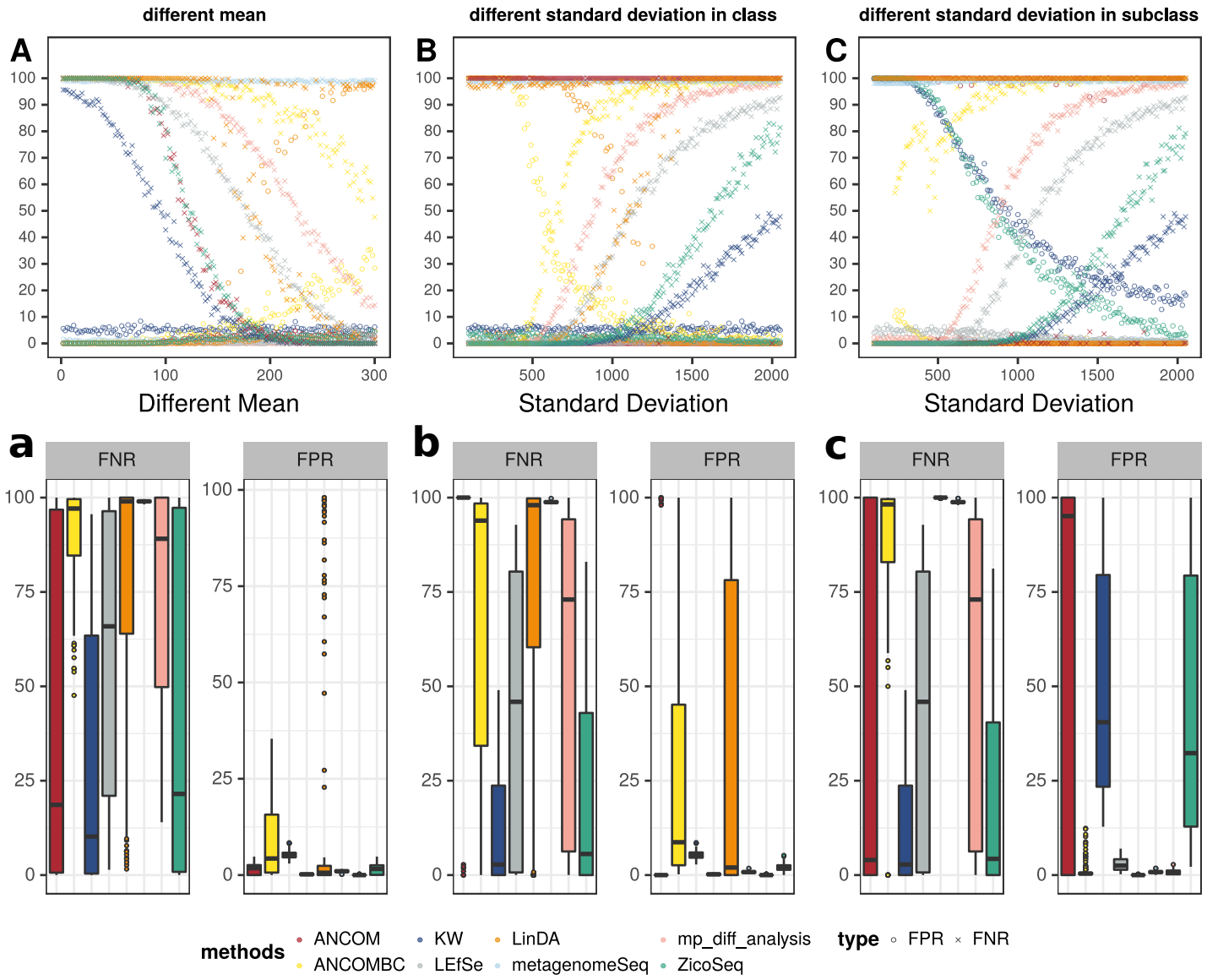

Fig. SB.12: **Comparison between MicrobiotaProcess and others common methods for false positive and negative rates in simulation datasets (20 samples) based on normal distribution**. Although the false negative rate of *mp_diff_analysis* is larger than the 100 samples, it also had better false positive rate.

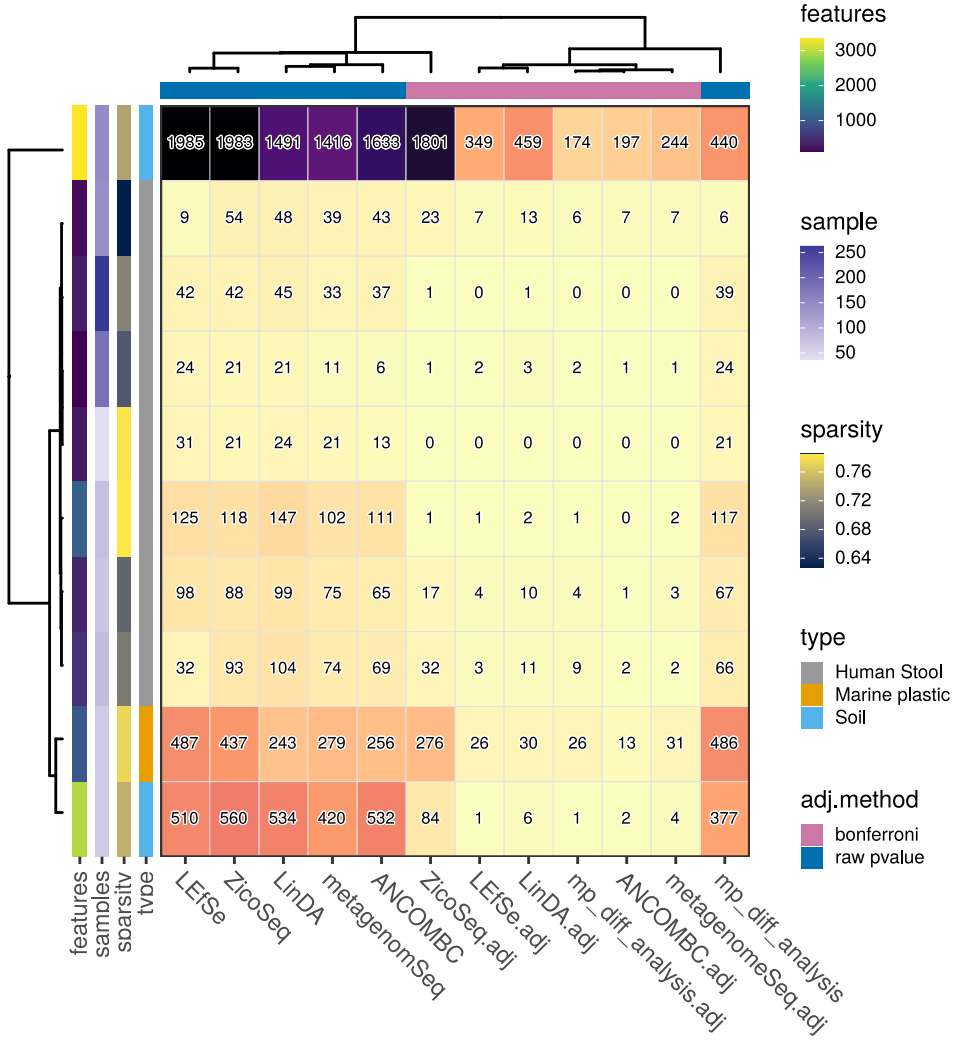

Fig. SB.13: **The heatmap of significant features number detected by different DAA tools across the 10 datasets.**

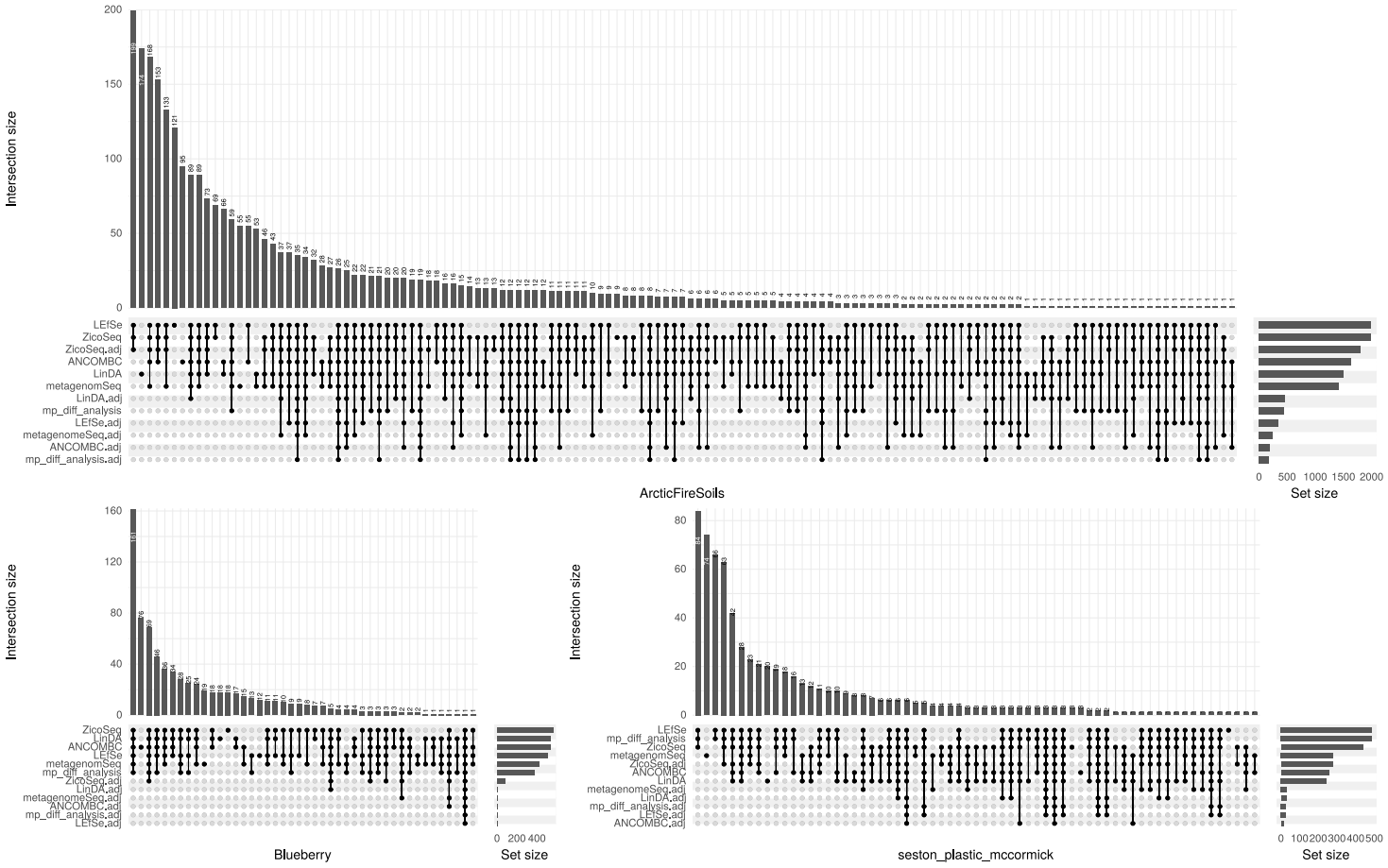

Fig. SB.14: **The upset plot of significant features detected by different DAA tools across the sample from soil or marine plastic.**

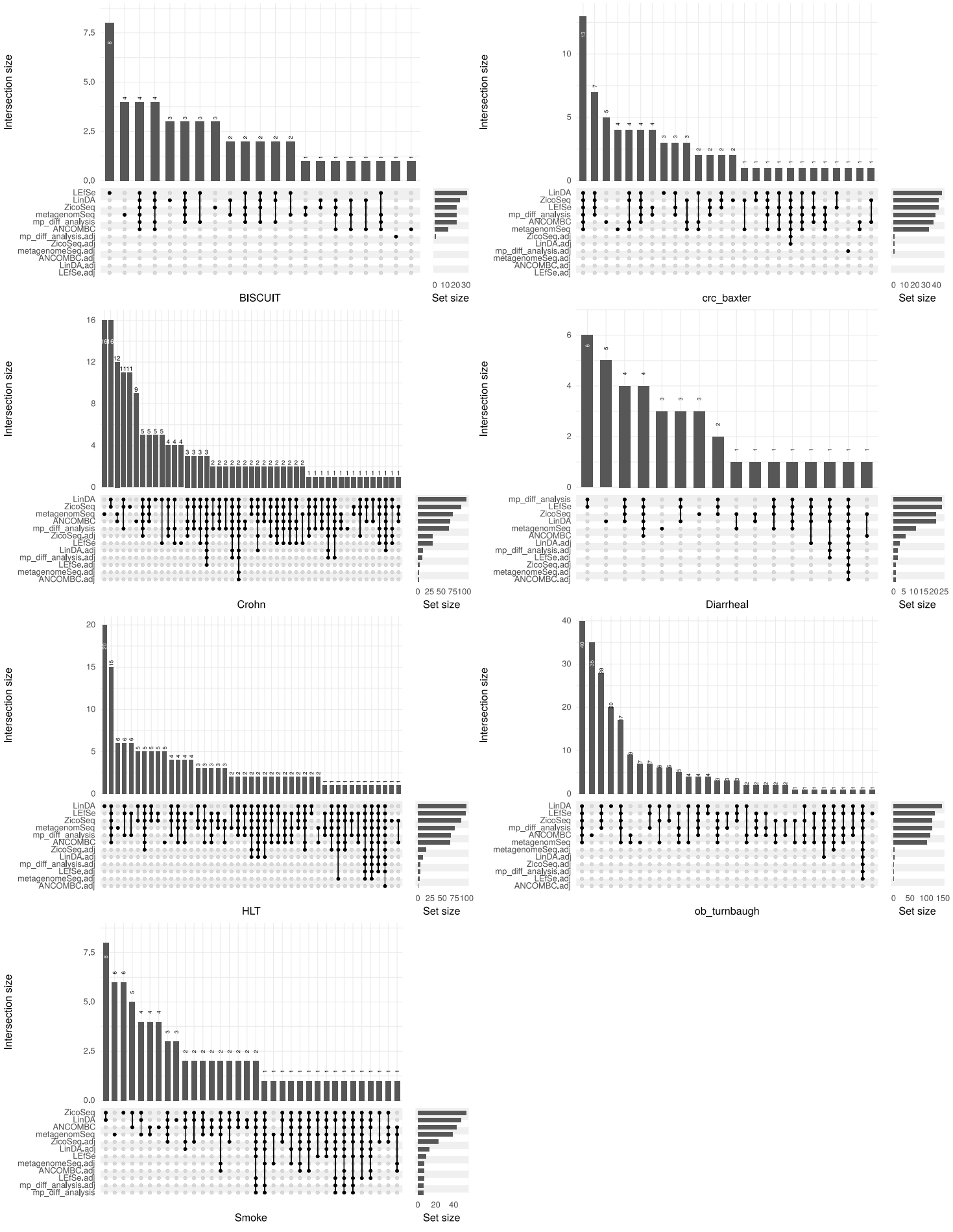

Fig. SB.15: **The upset plot of significant features detected by different DAA tools across the sample from human stool.**