Contents

BEMA Figures for comparing distance metrics using HTS bacteria.

```
library(wesanderson)
library(scales)
library(vegan)
```

```
## Loading required package: permute
## Loading required package: lattice
## This is vegan 2.0-10
```

```
# setwd('~/Dropbox/BEMA/bacteriaHTS/closed_ref_pickedOTUs/R/ordinations')
ordBCIn <- read.delim(pipe('cut -f1-4 ../../beta100_bc/bray_curtis_pc.txt'),
                      head=TRUE, row.names=1)
ordUNIIn <- read.delim(pipe('cut -f1-4 ../../beta100_ununif/unweighted_unifrac_pc.txt'),
                       head=TRUE, row.names=1)
taxaBCIn <- read.delim(pipe('cut -f1-4 ../../beta100_bc/3dplots_bc_L6_points.txt'),
                       head=TRUE, row.names=1)
taxaUNIIn <- read.delim(pipe('cut -f1-4 ../../beta100_ununif/3dplots_ununif_L6_points.txt'),
                        head=TRUE, row.names=1)

bokClosedIn <- read.delim(pipe('cut -f1-4 openClosed/bokulich_ununi_closed_pc.txt'),
                          head=TRUE, row.names=1)
bokOpenIn <- read.delim(pipe('cut -f1-4 openClosed/bokulich_ununi_open_pc.txt'),
                          head=TRUE, row.names=1)
lfClosedIn <- read.delim(pipe('cut -f1-4 openClosed/leff_flores_ununi_closed_pc.txt'),
                          head=TRUE, row.names=1)
lfOpenIn <- read.delim(pipe('cut -f1-4 openClosed/leff_flores_ununi_open_pc.txt'),
                          head=TRUE, row.names=1)
meaClosedIn <- read.delim(pipe('cut -f1-4 openClosed/meadow_ununi_closed_pc.txt'),
                          head=TRUE, row.names=1)
meaOpenIn <- read.delim(pipe('cut -f1-4 openClosed/meadow_ununi_open_pc.txt'),
                          head=TRUE, row.names=1)


bokOpenDivIn <- read.delim('openClosed/bokOpenAlphaDiv.txt', row.names=1)
bokClosedDivIn <- read.delim('openClosed/bokClosedAlphaDiv.txt', row.names=1)

load('../countData/dfWithColors.RData')
```

Remove extra info from PC dataframe. This has `eigvals` and `% variation explained` as the last two rows. Create a function to make new cleaned datasets.

```
cutEigs <- function(ord, ordName = 'newOrd', eigName = 'eigs') {
  assign(eigName, ord[c(nrow(ord)-1, nrow(ord)), ], pos=1)
  assign(ordName, ord[c(1:(nrow(ord)-2)), ], pos=1)
}
# cutEigs(bokOpenIn)
```

Big dataset - Bray Curtis vs UniFrac.

```
cutEigs(ord=ordBCIn, ordName='ordBC', eigName='eigBC')
cutEigs(ord=ordUNIIn, ordName='ordUNI', eigName='eigUNI')
```

Bokulich Dairy and NICU, closed and open reference datasets.

```
cutEigs(ord=bokClosedIn, ordName='bokClosed', eigName='eigBokClosed')
cutEigs(ord=bokOpenIn, ordName='bokOpen', eigName='eigBokOpen')
```

Leff and Flores Kitchens, closed and open reference datasets.

```
cutEigs(ord=lfClosedIn, ordName='lfClosed', eigName='eigLFClosed')
cutEigs(ord=lfOpenIn, ordName='lfOpen', eigName='eigLFOpen')
```

Meadow air and surfaces, closed and open reference datasets.

```
cutEigs(ord=meaClosedIn, ordName='meaClosed', eigName='eigMeaClosed')
cutEigs(ord=meaOpenIn, ordName='meaOpen', eigName='eigMeaOpen')
```

All samples are present but in a different order.

```
dim(bokClosed); dim(bokOpen)
```

```
## [1] 216   3
```

```
## [1] 216   3
```

```
all(row.names(bokOpen) %in% row.names(bokClosed))
```

```
## [1] TRUE
```

```
bokOpen <- bokOpen[row.names(bokClosed), ]
identical(row.names(bokOpen), row.names(bokClosed))
```

```
## [1] TRUE
```

```
dim(lfClosed); dim(lfOpen)
```

```
## [1] 630   3
```

```
## [1] 631   3
```

```
all(row.names(lfOpen) %in% row.names(lfClosed))
```

```
## [1] FALSE
```

```r
all(row.names(lfClosed) %in% row.names(lfOpen))
```

```
## [1] TRUE
```

```r
setdiff(row.names(lfOpen), row.names(lfClosed))  # one extra sample
```

```
## [1] "H27Ts.729226"
```

```r
lfOpen <- lfOpen[row.names(lfClosed), ]
identical(row.names(lfOpen), row.names(lfClosed))
```

```
## [1] TRUE
```

```r
dim(meaClosed); dim(meaOpen)
```

```
## [1] 368   3
```

```
## [1] 369   3
```

```r
all(row.names(meaClosed) %in% row.names(meaOpen))  # one extra sample
```

```
## [1] TRUE
```

```r
meaOpen <- meaOpen[row.names(meaClosed), ]
identical(row.names(meaOpen), row.names(meaClosed))
```

```
## [1] TRUE
```

```r
all(row.names(bokOpen) %in% row.names(dat))
```

```
## [1] TRUE
```

```r
bokMap <- dat[row.names(bokOpen), ]
lfMap <- dat[row.names(lfOpen), ]
meaMap <- dat[row.names(meaOpen), ]
```

```r
bokCO <- procrustes(bokClosed, bokOpen)
bokOC <- procrustes(bokOpen, bokClosed)
bokTest <- protest(bokOpen, bokClosed)
#
# plot(bokOC, kind=1)
#
# plot(bokOC, kind=0, las=1)
# lines(bokOC, col='gray20', type='arrows', code=1, length=.08)
# points(bokOC$Yrot, pch=21, bg=bokMap$col, col='gray20', lwd=.3)  # From points
```

```
lfCO <- procrustes(lfClosed, lfOpen)
lfOC <- procrustes(lfOpen, lfClosed)
lfTest <- protest(lfOpen, lfClosed)
#
# plot(lfOC, kind=1)
#
# plot(lfOC, kind=0, las=1)
# lines(lfOC, col='gray20', type='arrows', code=1, length=.08)
# points(lfOC$Yrot, pch=21, bg=lfMap$col, col='gray20', lwd=.3)  # From points
```

```
meaCO <- procrustes(meaClosed, meaOpen)
meaOC <- procrustes(meaOpen, meaClosed)
meaTest <- protest(meaOpen, meaClosed)

# plot(meaOC, kind=1)
#
# plot(meaOC, kind=0, las=1)
# lines(meaOC, col='gray20', type='arrows', code=1, length=.08)
# points(meaOC$Yrot, pch=21, bg=meaMap$col, col='gray20', lwd=.3)  # From points
```

Trying to find out why one study was much different closed vs open. Nothing convincing here.

```
dim(bokOpenDivIn)
```

```
## [1] 216   1
```

```
dim(bokClosedDivIn)
```

```
## [1] 216   1
```

```
all(row.names(bokMap) %in% row.names(bokClosedDivIn))
```

```
## [1] TRUE
```

```
bokMap$closed <- bokClosedDivIn[row.names(bokMap), 1]
bokMap$open <- bokOpenDivIn[row.names(bokMap), 1]
# plot(bokMap$closed, bokMap$open, pch=21, bg=bokMap$col)
# segments(0,0,1000,1000)
# boxplot(bokMap$open/bokMap$closed ~ factor(bokMap$phinchID))
```

bokMap[, c('phinchID', 'col')] points(.2, .15, pch=21, bg='#FD6467')

```
pdf('procrustes3panel.pdf', width=4, height=10, useDingbats=FALSE)
par(mfrow=c(3,1))
plot(bokOC, kind=0, las=1, main='(a) CA Dairy and NICU')
lines(bokOC, type='arrows', code=1, length=.03)
points(bokOC$Yrot, pch=21, bg=bokMap$col, lwd=.3)  # From points
legend('bottomright', legend=c('CA NICU', 'CA Dairy'),
       pch=21, pt.bg=unique(bokMap$col), bty='n', pt.cex=2)
mtext(paste('p =', bokTest$signif, '  '),
```

```
      side=3, adj=1, line=-1.5, cex=.7)
mtext(paste('corr =', format(bokTest$t0, digits=2), '  '),
      side=3, adj=1, line=-2.5, cex=.7)
# unique(meaMap$phinchID)
# unique(meaMap$col)

plot(lfOC, kind=0, las=1, main='(b) NC Homes and CO Residential Kitchens')
lines(lfOC, type='arrows', code=1, length=.03)
points(lfOC$Yrot, pch=21, bg=lfMap$col, lwd=.3)  # From points
legend('topleft', legend=c('NC Homes', 'CO Residential Kitches'),
       pch=21, pt.bg=unique(lfMap$col), bty='n', pt.cex=2)
mtext(paste('p =', lfTest$signif, '  '),
      side=3, adj=1, line=-1.5, cex=.7)
mtext(paste('corr =', format(lfTest$t0, digits=2), '  '),
      side=3, adj=1, line=-2.5, cex=.7)



plot(meaOC, kind=0, las=1, main='(c) OR Classroom Air and Surfaces')
lines(meaOC, type='arrows', code=1, length=.03)
points(meaOC$Yrot, pch=21, bg=meaMap$col, lwd=.3)  # From points
legend('bottomleft', legend=c('OR Classroom Air', 'OR Classroom Surfaces'),
       pch=21, pt.bg=unique(meaMap$col), bty='n', pt.cex=2)
mtext(paste('p =', meaTest$signif, '  '),
      side=1, adj=1, line=-2.5, cex=.7)
mtext(paste('corr =', format(meaTest$t0, digits=2), '  '),
      side=1, adj=1, line=-1.5, cex=.7)



dev.off()
```

```
## pdf
##   2
```

```
head(dat)  # Description
```

```
##                                          studyName       sampleName
## Summer.3.Balcony Adams_university_homes_Summer.txt Summer.3.Balcony
## Summer.1.Balcony Adams_university_homes_Summer.txt Summer.1.Balcony
## Summer.5.Balcony Adams_university_homes_Summer.txt Summer.5.Balcony
## Summer.9.Balcony Adams_university_homes_Summer.txt Summer.9.Balcony
## Summer.2.Balcony Adams_university_homes_Summer.txt Summer.2.Balcony
## Summer.6.Balcony Adams_university_homes_Summer.txt Summer.6.Balcony
##                  splitTotal otuTableTotal percentAssigned         SampleID
## Summer.3.Balcony       1462           393       0.2688098 Summer.3.Balcony
## Summer.1.Balcony       1449           427       0.2946860 Summer.1.Balcony
## Summer.5.Balcony       1393           357       0.2562814 Summer.5.Balcony
## Summer.9.Balcony       1225           369       0.3012245 Summer.9.Balcony
## Summer.2.Balcony       1175           426       0.3625532 Summer.2.Balcony
## Summer.6.Balcony       1109           373       0.3363390 Summer.6.Balcony
##                  BarcodeSequence LinkerPrimerSequence   Geolocation
## Summer.3.Balcony      TCTCTATGCG      CTGCTGCCTYCCGTA Albany_CA_USA
```

```
## Summer.1.Balcony      AGACGCACTC      CTGCTGCCTYCCGTA Albany_CA_USA
## Summer.5.Balcony      ACGACTACAG      CTGCTGCCTYCCGTA Albany_CA_USA
## Summer.9.Balcony      TACACACACT      CTGCTGCCTYCCGTA Albany_CA_USA
## Summer.2.Balcony      ATCAGACACG      CTGCTGCCTYCCGTA Albany_CA_USA
## Summer.6.Balcony      TAGAGACGAG      CTGCTGCCTYCCGTA Albany_CA_USA
##                   Building_Type Building_Type2 Room_Function
## Summer.3.Balcony  apt_building   apt_building       balcony
## Summer.1.Balcony  apt_building   apt_building       balcony
## Summer.5.Balcony  apt_building   apt_building       balcony
## Summer.9.Balcony  apt_building   apt_building       balcony
## Summer.2.Balcony  apt_building   apt_building       balcony
## Summer.6.Balcony  apt_building   apt_building       balcony
##                   Sequencing_Technology Target_Region Primer_name
## Summer.3.Balcony       454_GS_FLX_plus         V1_V2     8f_357r
## Summer.1.Balcony       454_GS_FLX_plus         V1_V2     8f_357r
## Summer.5.Balcony       454_GS_FLX_plus         V1_V2     8f_357r
## Summer.9.Balcony       454_GS_FLX_plus         V1_V2     8f_357r
## Summer.2.Balcony       454_GS_FLX_plus         V1_V2     8f_357r
## Summer.6.Balcony       454_GS_FLX_plus         V1_V2     8f_357r
##                   Sampling_Method Matrix Specific_Matrix Extraction_method
## Summer.3.Balcony    settle_plate    air             air             BB_PS
## Summer.1.Balcony    settle_plate    air             air             BB_PS
## Summer.5.Balcony    settle_plate    air             air             BB_PS
## Summer.9.Balcony    settle_plate    air             air             BB_PS
## Summer.2.Balcony    settle_plate    air             air             BB_PS
## Summer.6.Balcony    settle_plate    air             air             BB_PS
##                     phinchID            Description       H1  R         J
## Summer.3.Balcony Adams_univ Adams_university_homes 3.341877 45 0.8779021
## Summer.1.Balcony Adams_univ Adams_university_homes 4.221969 75 0.9778763
## Summer.5.Balcony Adams_univ Adams_university_homes 4.104851 71 0.9629742
## Summer.9.Balcony Adams_univ Adams_university_homes 3.682767 53 0.9275809
## Summer.2.Balcony Adams_univ Adams_university_homes 4.147422 72 0.9697792
## Summer.6.Balcony Adams_univ Adams_university_homes 4.154489 72 0.9714317
##                      col aveSeqLength    shDiv       pd percentAssigned4
## Summer.3.Balcony #F1BB7B        314.2 5.503370 10.74105      0.005221325
## Summer.1.Balcony #F1BB7B        314.2 7.246658 17.48570      0.007541157
## Summer.5.Balcony #F1BB7B        314.2 6.624400 14.59485      0.004313883
## Summer.9.Balcony #F1BB7B        314.2 5.863985 10.12272      0.008233057
## Summer.2.Balcony #F1BB7B        314.2 7.157339 18.61049      0.017277740
## Summer.6.Balcony #F1BB7B        314.2 6.822424 15.87730      0.012797029
```

```
head(ordBC)
```

```
##                         X1          X2          X3
## Swab.162.56     -0.24031427  0.01605995 -0.13927646
## Summer.6.K       0.04137722  0.02698765 -0.02230400
## H19Di.735897     0.06342659  0.06067658  0.05593397
## Swab.38.1135585  0.19521972 -0.42052570  0.06434218
## KiD54.630548     0.13858002  0.17643775  0.22278852
## KiC46.630516     0.15358584  0.14990360  0.25212928
```

```
dim(dat)
```

```
## [1] 3859   29
```

```
dim(ordBC)
```

```
## [1] 1724    3
```

```
dim(ordUNI)
```

```
## [1] 1724    3
```

```
all(row.names(ordBC) %in% row.names(ordUNI))
```

```
## [1] TRUE
```

```
all(row.names(ordBC) %in% row.names(dat))
```

```
## [1] FALSE
```

```
setdiff(row.names(ordBC), row.names(dat))
```

```
##  [1] "Winter.2.Balcony"  "Summer.8.K"        "Winter.2.LR"
##  [4] "Winter.4.Balcony"  "Winter.4.Bedroom"  "Winter.10.Balcony"
##  [7] "Winter.2.Kitchen"  "Winter.10.Bedroom" "Winter.8.Bedroom"
## [10] "Winter.8.Balcony"  "HVACdust3"         "Winter.2.Bedroom"
```

```
sameNames <- intersect(row.names(ordBC), row.names(dat))
ordBC <- ordBC[sameNames, ]
ordUNI <- ordUNI[sameNames, ]
dat <- dat[sameNames, ]
identical(row.names(ordBC), row.names(dat))
```

```
## [1] TRUE
```

```
taxaBC <- taxaBCIn
namesBC.tmp <- row.names(taxaBCIn)
namesBC.tmp <- gsub('\\;[a-z]\\_\\_$', '', namesBC.tmp)  # run twice to kill nonames
namesBC.tmp <- gsub('\\;[a-z]\\_\\_$', '', namesBC.tmp)
namesBC <- gsub('(.+)\\;[a-z]\\_\\_([[:alpha:]]{1,})$', '\\2', namesBC.tmp)
namesBC <- gsub('Enterobacteriaceae', 'Enterobact.', namesBC)
```

```
taxaUNI <- taxaUNIIn
namesUNI.tmp <- row.names(taxaUNI)
namesUNI.tmp <- gsub('\\;[a-z]\\_\\_$', '', namesUNI.tmp)  # run twice to kill nonames
namesUNI.tmp <- gsub('\\;[a-z]\\_\\_$', '', namesUNI.tmp)
namesUNI <- gsub('(.+)\\;[a-z]\\_\\_([[:alpha:]]{1,})$', '\\2', namesUNI.tmp)
namesUNI <- gsub('Enterobacteriaceae', 'Enterobact.', namesUNI)
```

```
posBC <- c(4, 4, 2, 2, 2, 4, 4, 2, 2, 4)
posUNI <- c(4, 4, 4, 2, 4, 2, 2, 2, 2, 2)
```

```
dat$colBuildingType <- ''
dat$colMatrix <- ''

dat$Building_Type3 <- dat$Building_Type
dat$Building_Type3[dat$Building_Type3 == 'apt_building'] <- 'home'
dat$Building_Type3 <- factor(dat$Building_Type3)

# wesBT <- data.frame(cols=c('cornflowerblue', 'darkolivegreen', 'darkorchid', 'darkorange'),
#                     levs=levels(factor(dat$Building_Type3)))
wesBT <- data.frame(cols=c(wes_palette('Darjeeling', 5)[c(1,3,5)], 'darkorchid')[c(1,4,2,3)],
                    levs=levels(factor(dat$Building_Type3)))
wesMAT <- data.frame(cols=c(wes_palette('Darjeeling', 5)[c(1,2,3)], 'darkorchid'),
                     levs=levels(factor(dat$Matrix)))
table(dat$Matrix)
```

```
##
##         air        dust kit_control        soil     surface       water
##         374         174           0           0        1159           5
```

```
dat$colBuildingType <- as.character(wesBT$cols[match(dat$Building_Type, wesBT$levs)])
dat$colMatrix <- as.character(wesMAT$cols[match(dat$Matrix, wesMAT$levs)])
```

names(dat) table(factor(dat$Matrix))$table(factor(dat$Building_Type)) table(factor(dat$Building_Type2))

Royal1, 1 Cavalcanti, 5 Rushmore, 5

```
# white lines with segments, fat lwd.

whiteLines <- function(coords=NULL, poss=NULL, Names=NULL, alpha=.8, start=0.02, end=0.013, lwd=14) {
  llen <- nchar(Names)*end
  for(i in 1:length(poss)) {
    if (poss[i] == 4) {
      segments(x0=coords[i, 1] + start,
               y0=coords[i, 2],
               x1=coords[i, 1] + llen[i],
               y1=coords[i, 2],
               col=alpha('white', alpha=alpha), lwd=14)
    }
    if (poss[i] == 2) {
      segments(x0=coords[i, 1] - start,
               y0=coords[i, 2],
               x1=coords[i, 1] - llen[i],
               y1=coords[i, 2],
               col=alpha('white', alpha=alpha), lwd=14)
    }
  }
}

# whiteLines(coords=ordUNI[, c(1,2)], poss=posUNI)
```

```
pdf('ordUni.pdf', useDingbats=FALSE, height=6, width=6)
par(mar=c(2, 2, 1, 1))
```

```
plot(ordUNI[, c(1, 2)], pch=16, col=alpha(dat$col, alpha=.9), cex=1, # col='gray20',
     bty='l', ann=FALSE, xaxt='n', yaxt='n', las=1)
mtext('PC1 (7.3%)', side=1, adj=1, line=.1)
mtext('PC2 (6.1%)', side=2, adj=1, line=.2)
# rect(-1, -1, 1, 1, col='gray')
whiteLines(coords=taxaUNI[, c(1,2)], poss=posUNI, Names=namesUNI, alpha=.8)
points(taxaUNI[, c(1, 2)], pch=24, bg='red', col='black', cex=1.5, lwd=2)
text(taxaUNI[, c(1,2)], labels=namesUNI, pos=posUNI, cex=.7, font=4)

dev.off()
```

```
pdf('ordThree.pdf', useDingbats=FALSE, height=10, width=10)
par(mfrow=c(2, 2), mar=c(2, 2, 1, 1))

# plot(ordBC[, c(1, 2)], pch=21, bg=alpha(dat$col, alpha=.7),
#      col=alpha(dat$col, alpha=.9), cex=1.2, # col='gray20',
#      bty='l', ann=FALSE, xaxt='n', yaxt='n')
# mtext('PC1 (7.4%)', side=1, adj=1, line=.1)
# mtext('PC2 (5.2%)', side=2, adj=1, line=.2)
# par(las=0, xpd=TRUE)
# mtext('   (a)', adj=0, line=-1, font=2, col='gray20', cex=1.4)
# # rect(-1, -1, 1, 1, col='gray')
# whiteLines(coords=taxaBC[, c(1,2)], poss=posBC, Names=namesBC, alpha=.8)
# points(taxaBC[, c(1, 2)], pch=24, bg='red', col='black', cex=1.5, lwd=2)
# text(taxaBC[, c(1,2)], labels=namesBC, pos=posBC, cex=.7, font=4)

plot(ordUNI[, c(1, 2)], pch=21, bg=alpha(dat$col, alpha=.7),
     col=alpha(dat$col, alpha=.9), cex=1.2, # col='gray20',
     bty='l', ann=FALSE, xaxt='n', yaxt='n', las=1)
mtext('PC1 (7.3%)', side=1, adj=1, line=.1)
mtext('PC2 (6.1%)', side=2, adj=1, line=.2)
mtext('   (a)', adj=0, line=-1, font=2, col='gray20', cex=1.4)
# rect(-1, -1, 1, 1, col='gray')
whiteLines(coords=taxaUNI[, c(1,2)], poss=posUNI, Names=namesUNI, alpha=.8)
points(taxaUNI[, c(1, 2)], pch=24, bg='red', col='black', cex=1.5, lwd=2)
text(taxaUNI[, c(1,2)], labels=namesUNI, pos=posUNI, cex=.7, font=4)

# exclude sources
studyNamesDFBE <- studyNamesDF[-c(2, 5, 6, 7, 11, 12, 23), ]

plot(0,0, axes=FALSE, ann=FALSE, type='n', xaxs='i', yaxs='i')
legend('bottomleft', legend=studyNamesDFBE$refNamesDisplay, pch=16,
       pt.cex=3, cex=1.5, col=as.character(studyNamesDFBE$refCol),
       bg='gray96', box.col='gray90')

purple <- which(dat$colMatrix == 'darkorchid')
plot(ordUNI[, c(1, 2)], pch=21, bg=alpha(dat$colMatrix, alpha=.7),
     col=alpha(dat$colMatrix, alpha=.9), cex=1.2, # col='gray20',
     bty='l', ann=FALSE, xaxt='n', yaxt='n', las=1)
points(ordUNI[, c(1, 2)][purple, ],
       pch=21, bg=alpha(dat$colMatrix[purple], alpha=.7),
       col=alpha(dat$colMatrix[purple], alpha=.9), cex=1.2)
mtext('PC1 (7.3%)', side=1, adj=1, line=.1)
```

```
mtext('PC2 (6.1%)', side=2, adj=1, line=.2)
mtext('   (b)', adj=0, line=-1, font=2, col='gray20', cex=1.4)
legend('bottomright', legend=wesMAT$levs, col=as.character(wesMAT$cols),
       pch=16, pt.cex=2.5, box.col='gray90', bg='gray96', cex=1.3)
par(xpd=FALSE)
abline(h=par()$usr[3])

plot(ordUNI[, c(1, 2)], pch=21, bg=alpha(dat$colBuildingType, alpha=.7),
     col=alpha(dat$colBuildingType, alpha=.9), cex=1.2, # col='gray20',
     bty='l', ann=FALSE, xaxt='n', yaxt='n', las=1)
mtext('PC1 (7.3%)', side=1, adj=1, line=.1)
mtext('PC2 (6.1%)', side=2, adj=1, line=.2)
mtext('   (c)', adj=0, line=-1, font=2, col='gray20', cex=1.4)
legend('bottomright', legend=wesBT$levs, col=as.character(wesBT$cols),
       pch=16, pt.cex=2.5, box.col='gray90', bg='gray96', cex=1.3)
abline(h=par()$usr[3])




dev.off()
```

```
## pdf
##   2
```

```
pdf('ordBC.pdf', useDingbats=FALSE, height=6, width=10)
par(mfrow=c(1, 2), mar=c(2, 2, 1, 1))

plot(ordBC[, c(1, 2)], pch=16, col=alpha(dat$col, alpha=.9), cex=1, # col='gray20',
     bty='l', ann=FALSE, xaxt='n', yaxt='n')
mtext('PC1 (7.4%)', side=1, adj=1, line=.1)
mtext('PC2 (5.2%)', side=2, adj=1, line=.2)
par(las=0, xpd=TRUE)
mtext('   (a)', adj=0, line=-1, font=2, col='gray20', cex=1.4)
# rect(-1, -1, 1, 1, col='gray')
whiteLines(coords=taxaBC[, c(1,2)], poss=posBC, Names=namesBC, alpha=.8)
points(taxaBC[, c(1, 2)], pch=24, bg='red', col='black', cex=1.5, lwd=2)
text(taxaBC[, c(1,2)], labels=namesBC, pos=posBC, cex=.7, font=4)

plot(0,0, axes=FALSE, ann=FALSE, type='n', xaxs='i', yaxs='i')
legend('bottomleft', legend=studyNamesDFBE$refNamesDisplay, pch=16,
       pt.cex=3, cex=1.5, col=as.character(studyNamesDFBE$refCol),
       bg='gray96', box.col='gray90')


dev.off()
```