bb16.data.exploration

Kendra Maas

2017-08-08

- Read in data
- Alpha Diversity
- Beta Diversity
    - jc= Jaccard dissimilarity measure presence absence
    - bc= Bray-Curtis dissimilarity measure that incoorporates abundance
    - tyc= Theta-YC dissimilarity measure that strongly weights abundance
    - Hypothesis testing on beta diversity
- Indicator species

Read in data

```
# otu <- read.table(file = "../bb16.oc.trim.contigs.good.unique.good.filter.precluster
.pick.pick.opti_mcc.0.03.subsample.shared", header=T, stringsAsFactors = FALSE, row.na
mes=2)
# otu <- select(otu, -label, -numOtus)
#
#
# taxa <- read.table(textConnection(gsub("\\(.+?\\);", "\t", readLines("../bb16.oc.tri
m.contigs.good.unique.good.filter.precluster.pick.pick.opti_mcc.0.03.cons.taxonomy")))
, col.names=c("OTU", "Size", "Kingdom", "Phylum", "Class", "Order", "Family", "Genus")
, skip=1)
# taxa <- taxa[taxa$OTU %in% names(otu),]
#
# # get OTU abundance for this subsampling
# sub.size <- data.frame(OTU = colnames(otu), size.sub = colSums(otu))
#
# taxa <- full_join(taxa, sub.size, by = "OTU", copy=TRUE)
#
#
# maxab <- apply(otu, 2, max)
# n1 <- names(which(maxab < 10))
# otu.ab <- otu[,-which(names(otu) %in% n1)]
# taxa.ab <- taxa[-which(taxa$OTU %in% n1),]
#
#
# otu.total <- otu
# taxa.total <- taxa
#
# otu <- otu.ab
# taxa <- taxa.ab
#
# write.csv(otu, file="otu.maxab10.csv")
# write.csv(taxa, file="taxa.maxab10.csv")

#### Reading in the full OTU matrix takes a long time, storing otu and taxa with max a
bundance 10. then reading that in each time. Also added "preferred" to taxa

otu <- read.csv(file="otu.maxab10.csv", header=T, stringsAsFactors = FALSE, row.names=
1)
taxa <- read.csv(file="taxa.maxab10.csv")
taxa$preferred <- factor(taxa$preferred, levels=taxa$preferred[order(taxa$Phylum)], or
dered = TRUE)
```

```
## Warning in `levels<-`(`*tmp*`, value = if (nl == nL) as.character(labels)
## else paste0(labels, : duplicated levels in factors are deprecated
```

```
taxa$preferred <- factor(taxa$preferred[,drop=TRUE])
```

```
## Warning in `levels<-`(`*tmp*`, value = if (nl == nL) as.character(labels)
## else paste0(labels, : duplicated levels in factors are deprecated
```

```
# get OTU abundance for this subsampling

maxab <- apply(otu, 2, max)
n1 <- names(which(maxab < 200))
otu.ab <- otu[,-which(names(otu) %in% n1)]
taxa.ab <- taxa[-which(taxa$OTU %in% n1),]

# Alpha diversity

alpha <- read.table(file="../bb16.oc.trim.contigs.good.unique.good.filter.precluster.p
ick.pick.opti_mcc.groups.ave-std.summary", header=T, stringsAsFactors = FALSE)
alpha <- filter(alpha, label== "0.03" & method == "ave")

# Beta diversity
jc <- parseDistanceDF("../bb16.oc.trim.contigs.good.unique.good.filter.precluster.pick
.pick.opti_mcc.jest.0.03.lt.ave.dist")
bc <- parseDistanceDF("../bb16.oc.trim.contigs.good.unique.good.filter.precluster.pick
.pick.opti_mcc.braycurtis.0.03.lt.ave.dist")
tyc <- parseDistanceDF("../bb16.oc.trim.contigs.good.unique.good.filter.precluster.pic
k.pick.opti_mcc.thetayc.0.03.lt.ave.dist")

# experimental data
expdata <- read.table(file="../BioBlitz2016_env.txt", header=T, stringsAsFactors = FAL
SE, fill=TRUE)
expdata <- arrange(expdata, group)

expdata$detail.type <- factor(expdata$detail.type, levels = expdata$detail.type[order(
expdata$bioblitz.transect)], ordered=TRUE)
```

```
## Warning in `levels<-`(`*tmp*`, value = if (nl == nL) as.character(labels)
## else paste0(labels, : duplicated levels in factors are deprecated
```

```
expdata$detail.type <- factor(expdata$detail.type[,drop=TRUE])
```

```
## Warning in `levels<-`(`*tmp*`, value = if (nl == nL) as.character(labels)
## else paste0(labels, : duplicated levels in factors are deprecated
```

```
alpha.expdata <- left_join(alpha, expdata, on="group")
```

```
## Joining, by = "group"
```

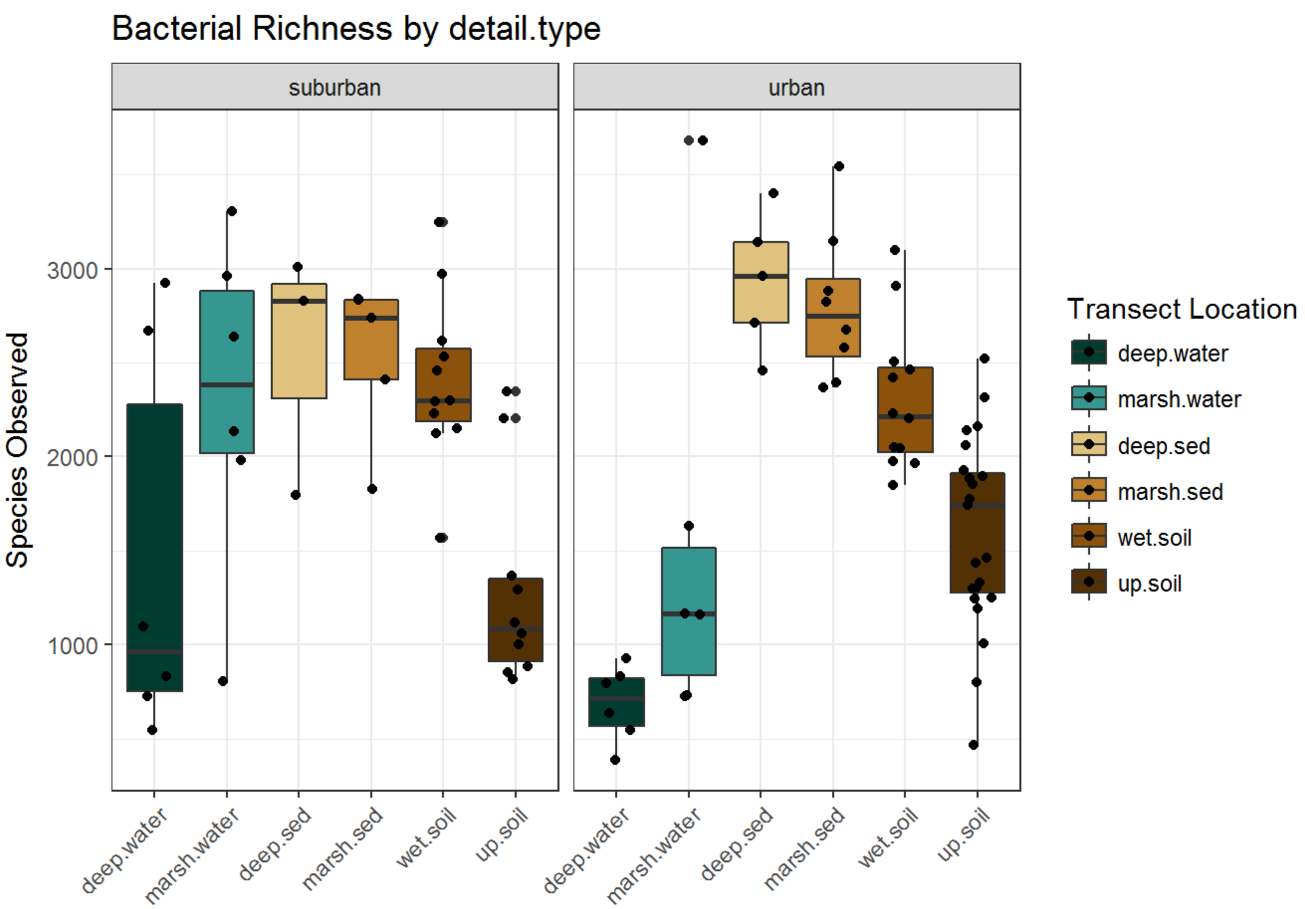
```
bb <- alpha.expdata$experiment == "bioblitz.transect"

alpha.expdata <- alpha.expdata[bb,]
alpha.expdata$detail.type <- factor(alpha.expdata$detail.type[,drop=TRUE])
jc <- jc[bb,bb]
bc <- bc[bb,bb]
tyc <- tyc[bb,bb]
otu <- otu[bb,]
otu.ab <- otu.ab[bb,]
```

Alpha Diversity

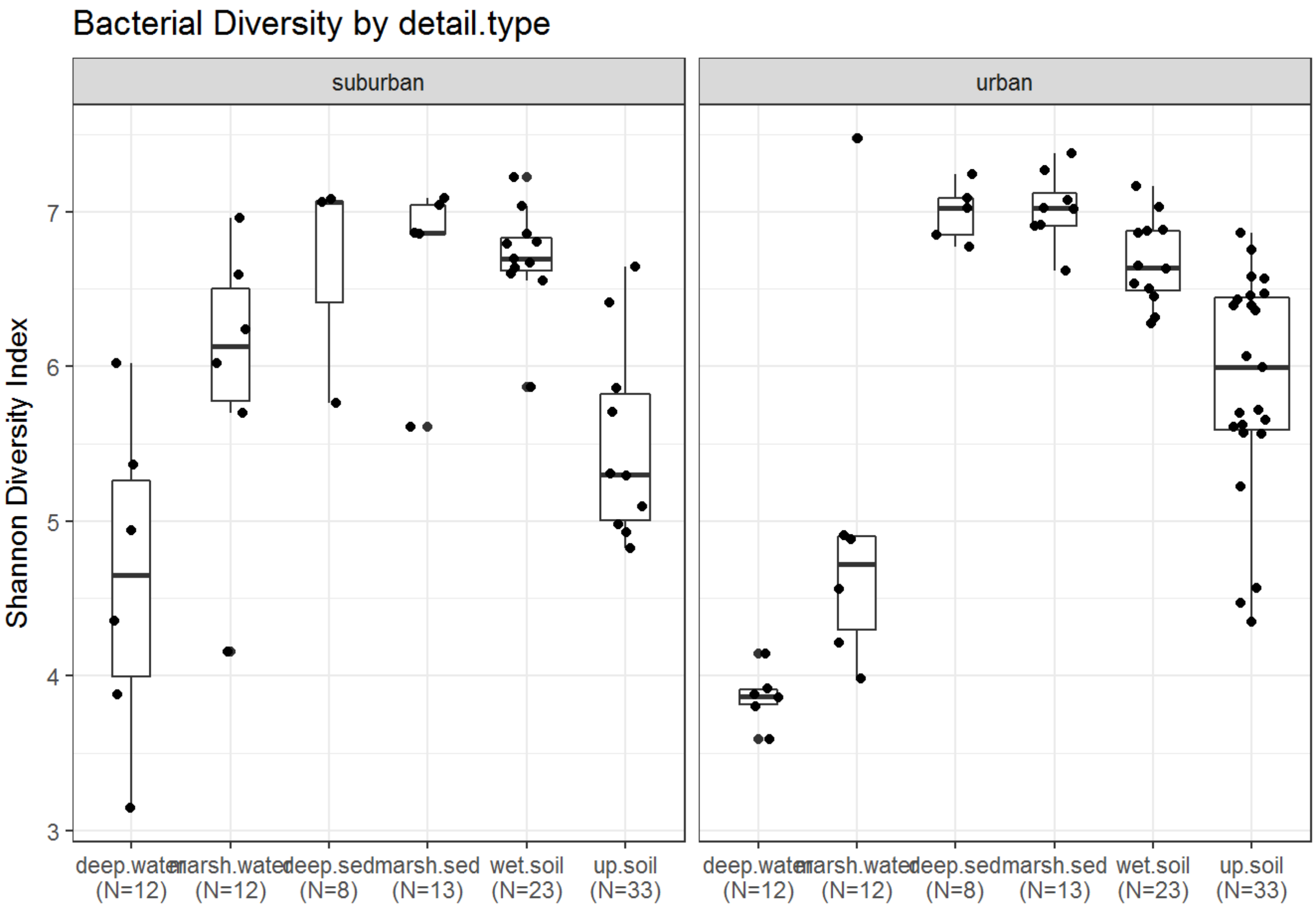
Boxplots of alpha diversity measures (diversity within a sample)

I'm adding the actual data points to the first one just to demonstrate the possiblility. Sometimes I like it in addition to the boxplot

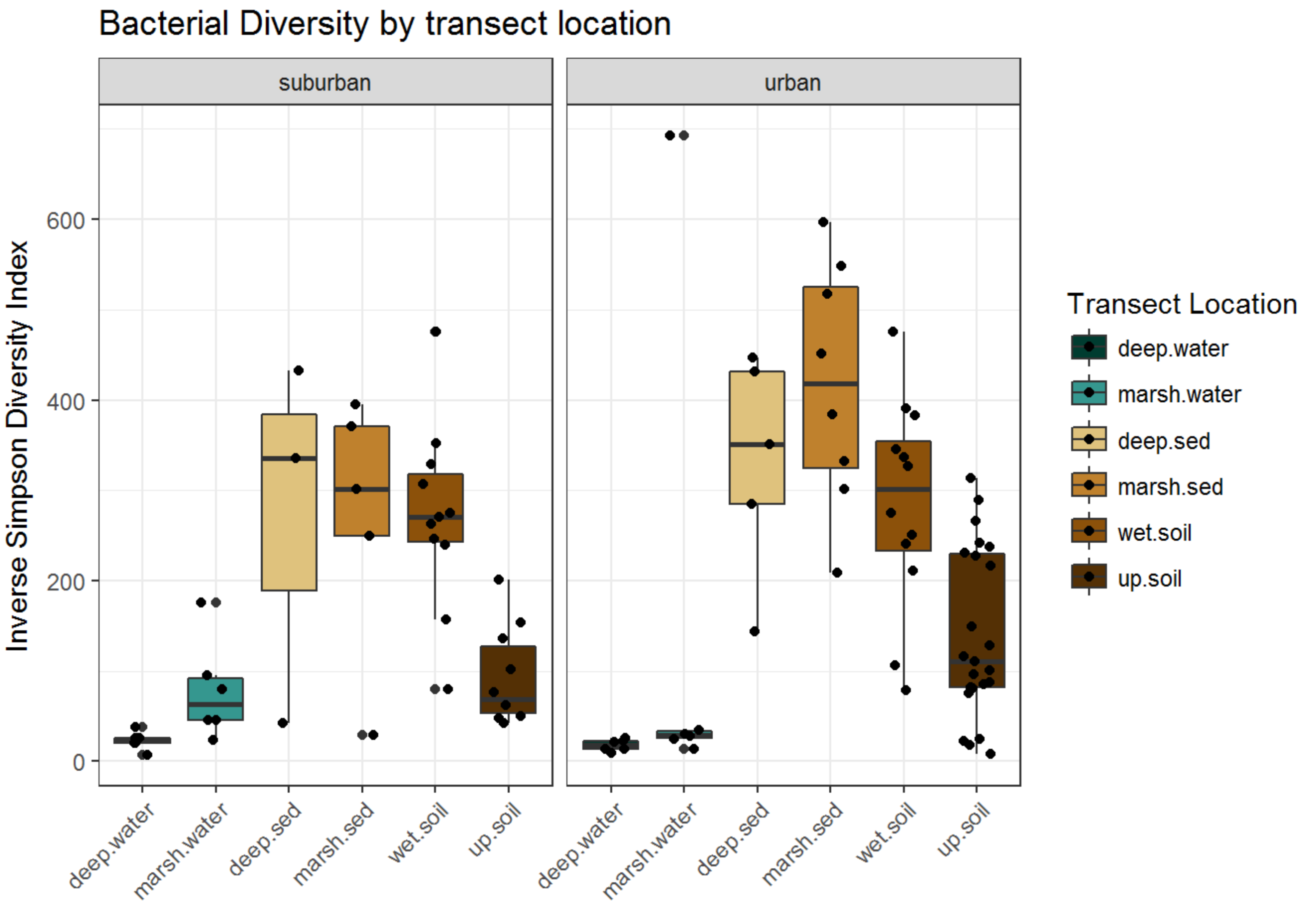
```
# alpha diversity (diversity within a sample)

treat.col <- c( "deep.water" = "#003c30", "marsh.water" = "#35978f", "deep.sed" = "#df
c27d",   "marsh.sed" = "#bf812d", "wet.soil" = "#8c510a", "up.soil" = "#543005")



xlab <- paste(levels(as.factor(alpha.expdata$detail.type)),"\n(N=", table(alpha.expdat
a$detail.type),")", sep="")
ggplot(data=alpha.expdata, (aes(x=detail.type, y=sobs, fill=detail.type)))+
    geom_boxplot()+
    scale_fill_manual(values=treat.col, name= "Transect Location")+
    geom_jitter(width=0.2)+
    # scale_x_discrete(labels=xlab)+
    theme_bw()+
    ggtitle("Bacterial Richness by detail.type")+
    labs(y="Species Observed", x="")+
    facet_grid(.~alpha.expdata$urban)+
    theme(axis.text.x = element_text(angle =  45, hjust = 1))
```



```
ggsave(file="richness.pdf", width=8, height=4)

xlab <- paste(levels(as.factor(alpha.expdata$detail.type)),"\n(N=", table(alpha.expdata$detail.type),")", sep="")
ggplot(data=alpha.expdata, (aes(x=detail.type, y=shannon)))+
    geom_boxplot(varwidth = T)+
    geom_jitter(width=0.2)+
    scale_x_discrete(labels=xlab)+
    theme_bw()+
    ggtitle("Bacterial Diversity by detail.type")+
    labs(y="Shannon Diversity Index", x="")+
    facet_grid(.~alpha.expdata$urban)
```



```
xlab <- paste(levels(as.factor(alpha.expdata$detail.type)),"\n(N=", table(alpha.expdat
a$detail.type),")", sep="")
ggplot(data=alpha.expdata, (aes(x=detail.type, y=invsimpson, fill=detail.type)))+
    geom_boxplot()+
    scale_fill_manual(values=treat.col, name= "Transect Location")+
    geom_jitter(width=0.2)+
    # scale_x_discrete(labels=xlab)+
    theme_bw()+
    ggtitle("Bacterial Diversity by transect location")+
    labs(y="Inverse Simpson Diversity Index", x="")+
    facet_grid(.~alpha.expdata$urban)+
    theme(axis.text.x = element_text(angle =  45, hjust = 1))
```



```
ggsave(file="simp.pdf", width=8, height=4)
```

Beta Diversity

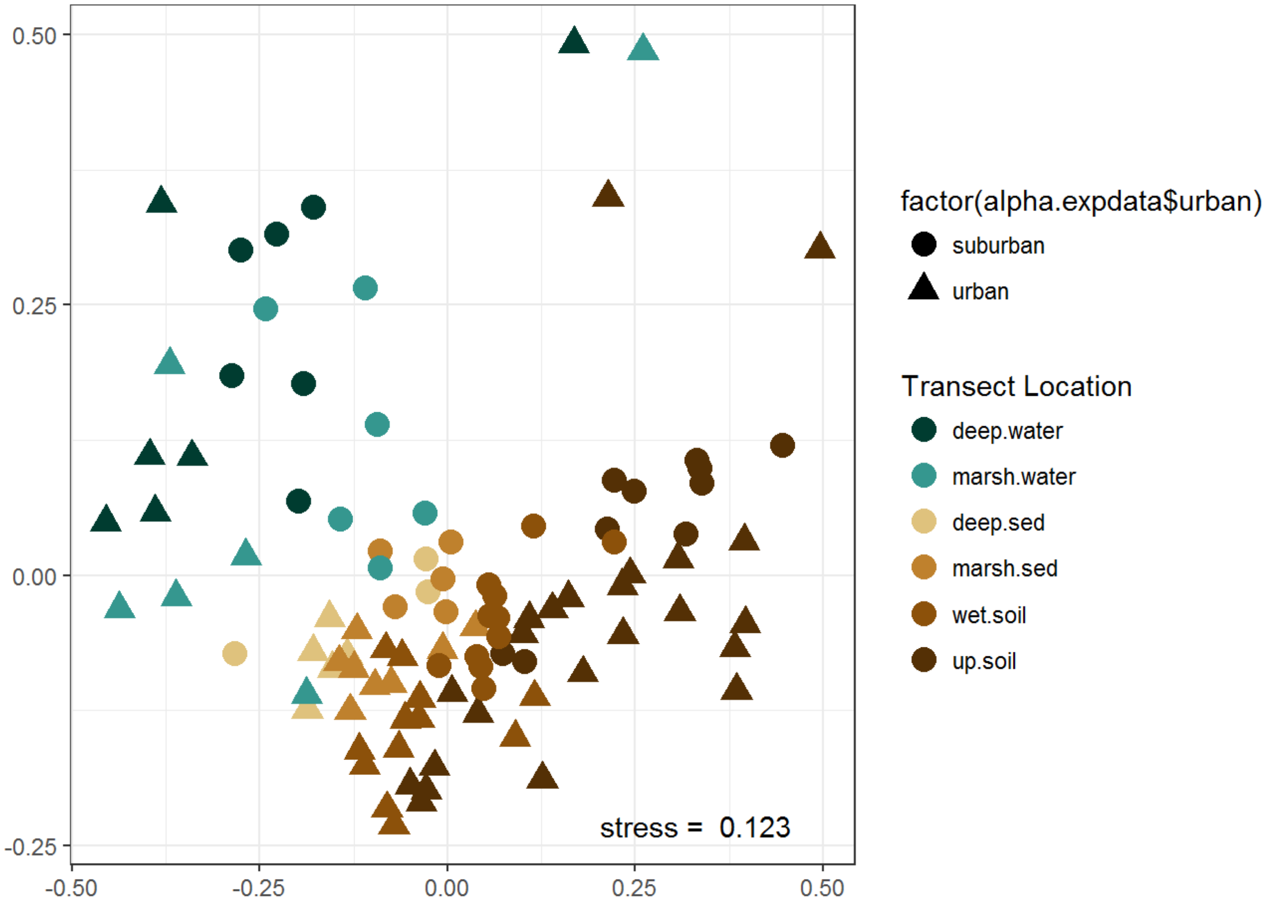
Non-metric multidimensional scaling to visualize between sample diversity

jc= Jaccard dissimilarity measure presence absence

```
## Whole dataset


jc.nms <- metaMDS(as.dist(jc), k=2, trymin=50, trymax=500, wascores=F)
```

```
## Run 0 stress 0.1232428
## Run 1 stress 0.1504962
## Run 2 stress 0.1306755
## Run 3 stress 0.1231006
## ... New best solution
## ... Procrustes: rmse 0.005024819  max resid 0.04366727
## Run 4 stress 0.1231893
## ... Procrustes: rmse 0.004669763  max resid 0.04377058
## Run 5 stress 0.1236013
## Run 6 stress 0.1457962
## Run 7 stress 0.1231895
## ... Procrustes: rmse 0.004668765  max resid 0.04375518
## Run 8 stress 0.1236017
## Run 9 stress 0.1487857
## Run 10 stress 0.1231003
## ... New best solution
## ... Procrustes: rmse 0.0001440472  max resid 0.001188463
## ... Similar to previous best
## Run 11 stress 0.1231018
## ... Procrustes: rmse 0.000413814  max resid 0.003056931
## ... Similar to previous best
## Run 12 stress 0.1232437
## ... Procrustes: rmse 0.003581118  max resid 0.02900675
## Run 13 stress 0.1232372
## ... Procrustes: rmse 0.003241484  max resid 0.02531527
## Run 14 stress 0.1232428
## ... Procrustes: rmse 0.003393778  max resid 0.02690156
## Run 15 stress 0.1456277
## Run 16 stress 0.1231825
## ... Procrustes: rmse 0.004641348  max resid 0.04375962
## Run 17 stress 0.1525975
## Run 18 stress 0.13067
## Run 19 stress 0.1231832
## ... Procrustes: rmse 0.004655156  max resid 0.04395332
## Run 20 stress 0.1538909
## *** Solution reached
```

```
jc.points <- data.frame(jc.nms$points)
jc.plot <- ggplot(jc.points, aes(x=MDS1, y=MDS2,label=rownames(jc)))

x <- max(jc.points$MDS1)/1.5
y <- min(jc.points$MDS2)

treat.col <- c( "deep.water" = "#003c30", "marsh.water" = "#35978f", "deep.sed" = "#df
c27d",   "marsh.sed" = "#bf812d", "wet.soil" = "#8c510a", "up.soil" = "#543005")

urban.shape <- c("suburban" = 21, "urban"=24)

jc.plot + geom_point(aes(color=factor(alpha.expdata$detail.type), shape=factor(alpha.e
xpdata$urban)), size=4)+
    theme_bw()+
    scale_color_manual(values=treat.col, name="Transect Location")+
    annotate("text", x, y, label=paste("stress = ", round(jc.nms$stress, digits = 3)))
+
    labs(x="", y="")
```



```
ggsave(file="jc.detail.type.nms.jpg")
```

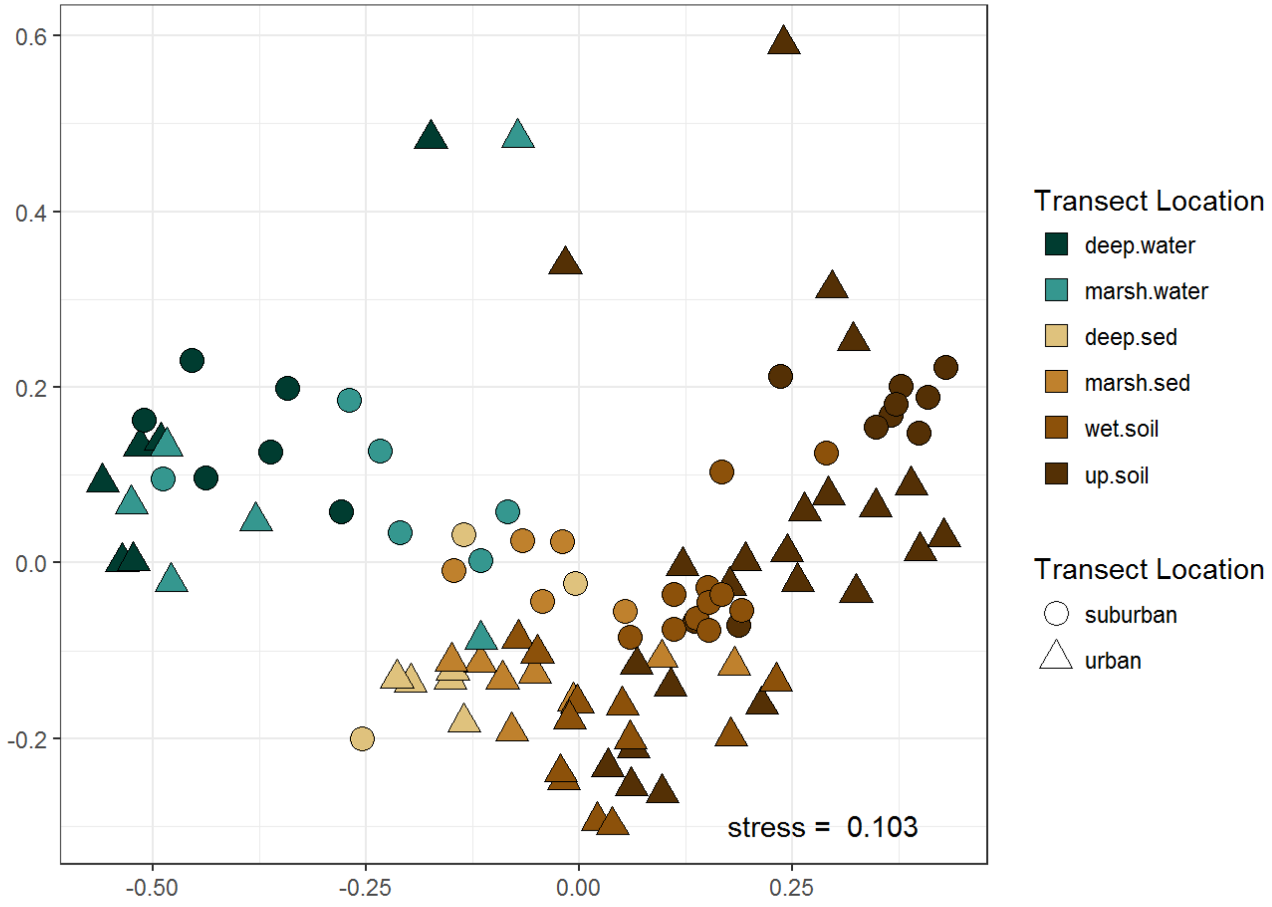
```
## Saving 7 x 5 in image
```

bc= Bray-Curtis dissimilarity measure that incoorporates abundance

```
## Whole dataset



bc.nms <- metaMDS(as.dist(bc), k=2, trymin=50, trymax=500, wascores=F)
```

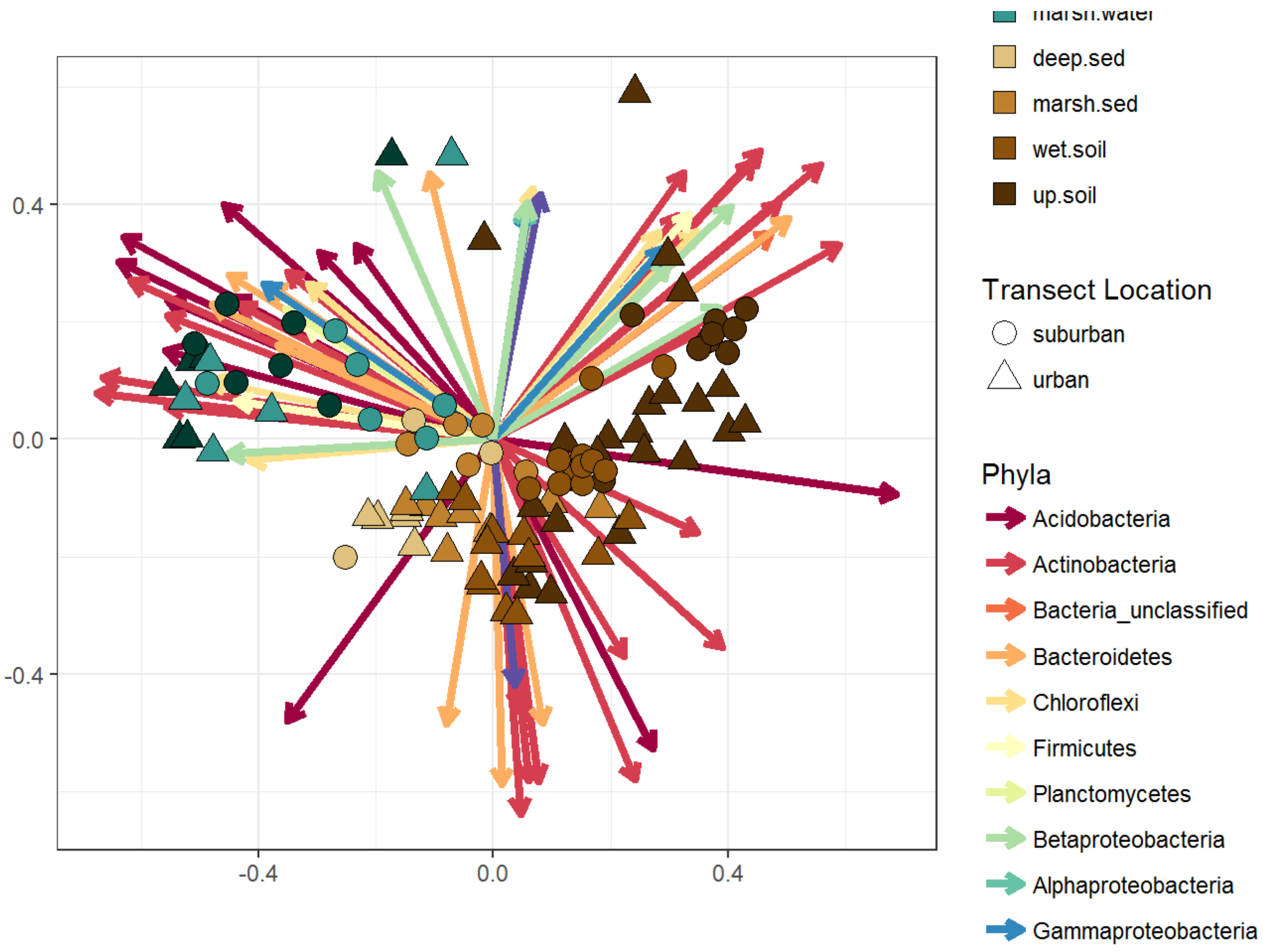
```
## Run 0 stress 0.1025638
## Run 1 stress 0.1050749
## Run 2 stress 0.127317
## Run 3 stress 0.1298013
## Run 4 stress 0.1267666
## Run 5 stress 0.1143354
## Run 6 stress 0.106252
## Run 7 stress 0.1218801
## Run 8 stress 0.1356712
## Run 9 stress 0.1254444
## Run 10 stress 0.1032106
## Run 11 stress 0.1089012
## Run 12 stress 0.1149467
## Run 13 stress 0.1343586
## Run 14 stress 0.13285
## Run 15 stress 0.1330561
## Run 16 stress 0.1143256
## Run 17 stress 0.134131
## Run 18 stress 0.1413847
## Run 19 stress 0.1144384
## Run 20 stress 0.1052155
## Run 21 stress 0.130211
## Run 22 stress 0.1398594
## Run 23 stress 0.123241
## Run 24 stress 0.1076848
## Run 25 stress 0.1149321
## Run 26 stress 0.137755
## Run 27 stress 0.1025231
## ... New best solution
## ... Procrustes: rmse 0.001416578  max resid 0.0105251
## Run 28 stress 0.1143261
## Run 29 stress 0.1264826
## Run 30 stress 0.1281436
## Run 31 stress 0.1273164
## Run 32 stress 0.1281335
## Run 33 stress 0.1232369
## Run 34 stress 0.1057721
## Run 35 stress 0.1077067
## Run 36 stress 0.1032123
## Run 37 stress 0.1324654
## Run 38 stress 0.1032107
## Run 39 stress 0.1434659
## Run 40 stress 0.1104111
## Run 41 stress 0.1273
## Run 42 stress 0.1234015
## Run 43 stress 0.1149343
## Run 44 stress 0.1143256
## Run 45 stress 0.1454965
## Run 46 stress 0.1233739
## Run 47 stress 0.1206225
## Run 48 stress 0.1312704
## Run 49 stress 0.1164771
## Run 50 stress 0.1168084
## Run 51 stress 0.116885
## Run 52 stress 0.1027826
## ... Procrustes: rmse 0.006496002  max resid 0.06099496
## Run 53 stress 0.1212984
## Run 54 stress 0.1027399
## ... Procrustes: rmse 0.006374129  max resid 0.06105917
## Run 55 stress 0.1320576
## Run 56 stress 0.1303498
## Run 57 stress 0.1086549
## Run 58 stress 0.1409169
## Run 59 stress 0.1298073
## Run 60 stress 0.130446
## Run 61 stress 0.1256331
## Run 62 stress 0.1232367
## Run 63 stress 0.132153
## Run 64 stress 0.111044
## Run 65 stress 0.114947
## Run 66 stress 0.1143684
## Run 67 stress 0.1220938
## Run 68 stress 0.1414331
## Run 69 stress 0.1448332
## Run 70 stress 0.1297591
## Run 71 stress 0.1320246
## Run 72 stress 0.1323271
## Run 73 stress 0.1319906
## Run 74 stress 0.1333652
## Run 75 stress 0.1319913
## Run 76 stress 0.1328586
## Run 77 stress 0.1395662
## Run 78 stress 0.114293
## Run 79 stress 0.1142924
## Run 80 stress 0.1294599
## Run 81 stress 0.111772
## Run 82 stress 0.1418816
## Run 83 stress 0.1370095
## Run 84 stress 0.1268202
## Run 85 stress 0.1213095
## Run 86 stress 0.1029476
## ... Procrustes: rmse 0.01202865  max resid 0.116027
## Run 87 stress 0.1032613
## Run 88 stress 0.1348418
## Run 89 stress 0.1258104
## Run 90 stress 0.1421339
## Run 91 stress 0.1344899
## Run 92 stress 0.1149329
## Run 93 stress 0.1174583
## Run 94 stress 0.1483528
## Run 95 stress 0.135648
## Run 96 stress 0.1176862
## Run 97 stress 0.1077081
## Run 98 stress 0.1230097
## Run 99 stress 0.130345
## Run 100 stress 0.1077056
## Run 101 stress 0.1310185
## Run 102 stress 0.1304473
## Run 103 stress 0.1081699
## Run 104 stress 0.1229762
## Run 105 stress 0.1370252
## Run 106 stress 0.1108476
## Run 107 stress 0.1143593
## Run 108 stress 0.1224274
## Run 109 stress 0.105075
## Run 110 stress 0.1171458
## Run 111 stress 0.1142923
## Run 112 stress 0.1164869
## Run 113 stress 0.1029478
## ... Procrustes: rmse 0.01202754  max resid 0.1159677
## Run 114 stress 0.1361015
## Run 115 stress 0.1062518
## Run 116 stress 0.1212894
## Run 117 stress 0.1143685
## Run 118 stress 0.1234001
## Run 119 stress 0.128321
## Run 120 stress 0.1312869
## Run 121 stress 0.1328492
## Run 122 stress 0.1203655
## Run 123 stress 0.1105993
## Run 124 stress 0.1171447
## Run 125 stress 0.114947
## Run 126 stress 0.1054661
## Run 127 stress 0.102564
## ... Procrustes: rmse 0.001443342  max resid 0.01070274
## Run 128 stress 0.1312658
## Run 129 stress 0.1255172
## Run 130 stress 0.1281406
## Run 131 stress 0.1395678
## Run 132 stress 0.1025632
## ... Procrustes: rmse 0.001301347  max resid 0.009562448
## ... Similar to previous best
## *** Solution reached
```

```
bc.points <- data.frame(bc.nms$points)
bc.plot <- ggplot(bc.points, aes(x=MDS1, y=MDS2,label=rownames(bc)))

x <- max(bc.points$MDS1)/1.5
y <- min(bc.points$MDS2)


bc.plot + geom_point(aes(fill=factor(alpha.expdata$detail.type), shape=factor(alpha.ex
pdata$urban)), size=4)+
    theme_bw()+
    scale_fill_manual(values=treat.col, name= "Transect Location")+
    scale_shape_manual(values=urban.shape, name="Transect Location")+
    guides(fill = guide_legend(override.aes = list(shape = 22)))+
    annotate("text", x, y, label=paste("stress = ", round(bc.nms$stress, digits = 3)))
+
    labs(x="", y="")
```



```
ggsave(file="bc.nms.pdf", width=8, height=6)
```

```
scrs <- as.data.frame(scores(bc.nms, display ='sites'))
scrs <- cbind(scrs, detail.type = alpha.expdata$detail.type, location = alpha.expdata$urban)


spec.ab <- envfit(bc.nms, otu.ab, perm=999)

spp.scrs <- as.data.frame((scores(spec.ab, display="vectors")))
spp.scrs <- cbind(spp.scrs, Species =rownames(spp.scrs))
spp.scrs <- cbind(spp.scrs, pVal=spec.ab$vectors$pvals)
# spp.scrs <- cbind(spp.scrs, Taxa=taxa$prefered, phy.col=taxa$phy.col)
spp.scrs <- cbind(spp.scrs, Phyla=taxa.ab$preferred)

###color vectors by phyla

ggplot(scrs)+
    coord_fixed()+
    geom_segment(data=spp.scrs[spp.scrs$pVal<0.0011,],
                 aes(x=0, xend=NMDS1, y=0, yend=NMDS2, color = Phyla),
                 arrow=arrow(length=unit (0.25, "cm")), size=1.5)+
    scale_color_brewer(palette = "Spectral" )+
    geom_point(mapping=aes(x=NMDS1, y=NMDS2, fill=factor(alpha.expdata$detail.type), shape=factor(alpha.expdata$urban)), size=4)+
    theme_bw()+
    scale_fill_manual(values=treat.col, name= "Transect Location")+
    scale_shape_manual(values=urban.shape, name="Transect Location")+
    guides(fill = guide_legend(override.aes = list(shape = 22)))+
    labs(x="",y="")+
    theme_bw()
```



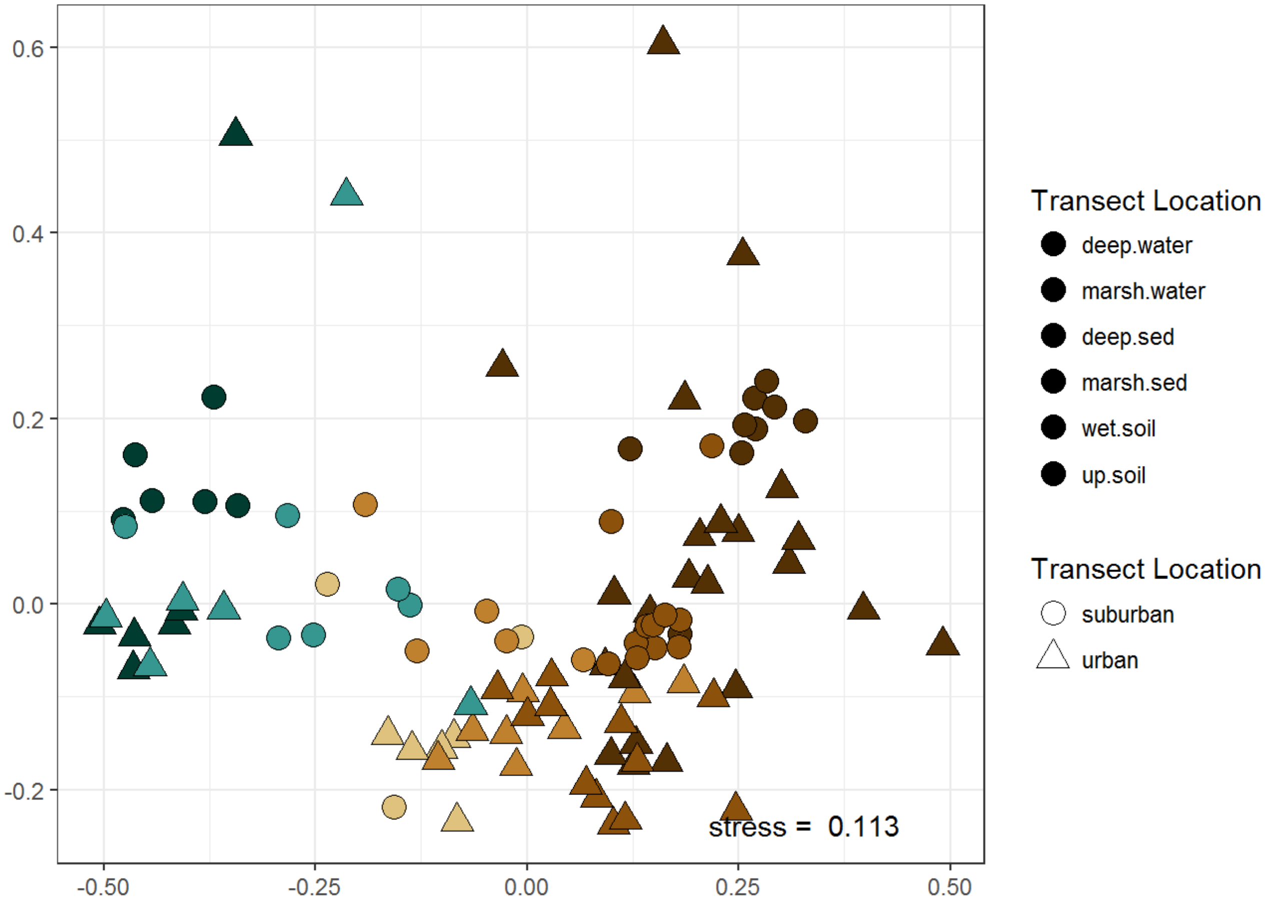
```
ggsave(file="bc.nms.vectors.pdf", width=8, height=7)
```

tyc= Theta-YC dissimilarity measure that strongly weights abundance

```
## Whole dataset


tyc.nms <- metaMDS(as.dist(tyc), k=2, trymin=50, trymax=500, wascores=F)
```

```
## Run 0 stress 0.1177659
## Run 1 stress 0.1497705
## Run 2 stress 0.1445921
## Run 3 stress 0.1384742
## Run 4 stress 0.1133492
## ... New best solution
## ... Procrustes: rmse 0.02062982  max resid 0.1555316
## Run 5 stress 0.1313834
## Run 6 stress 0.1480553
## Run 7 stress 0.1200737
## Run 8 stress 0.1234166
## Run 9 stress 0.139227
## Run 10 stress 0.117458
## Run 11 stress 0.4122395
## Run 12 stress 0.1239196
## Run 13 stress 0.1236222
## Run 14 stress 0.1196333
## Run 15 stress 0.129406
## Run 16 stress 0.1387303
## Run 17 stress 0.1299114
## Run 18 stress 0.1393253
## Run 19 stress 0.1200724
## Run 20 stress 0.1550468
## Run 21 stress 0.1177659
## Run 22 stress 0.1154396
## Run 23 stress 0.1234162
## Run 24 stress 0.147601
## Run 25 stress 0.1438576
## Run 26 stress 0.1297386
## Run 27 stress 0.1329918
## Run 28 stress 0.1375822
## Run 29 stress 0.1297382
## Run 30 stress 0.1420912
## Run 31 stress 0.1436677
## Run 32 stress 0.1441431
## Run 33 stress 0.14057
## Run 34 stress 0.1200751
## Run 35 stress 0.1391676
## Run 36 stress 0.1298883
## Run 37 stress 0.1259147
## Run 38 stress 0.1413657
## Run 39 stress 0.1420945
## Run 40 stress 0.1174581
## Run 41 stress 0.1133796
## ... Procrustes: rmse 0.001513682  max resid 0.01187592
## Run 42 stress 0.1236562
## Run 43 stress 0.1260291
## Run 44 stress 0.125312
## Run 45 stress 0.1236219
## Run 46 stress 0.1200731
## Run 47 stress 0.1342233
## Run 48 stress 0.1168542
## Run 49 stress 0.1260294
## Run 50 stress 0.1495676
## Run 51 stress 0.121034
## Run 52 stress 0.1341828
## Run 53 stress 0.120074
## Run 54 stress 0.1329754
## Run 55 stress 0.1295047
## Run 56 stress 0.1472969
## Run 57 stress 0.1133789
## ... Procrustes: rmse 0.001382201  max resid 0.01127132
## Run 58 stress 0.1294412
## Run 59 stress 0.1387315
## Run 60 stress 0.1297644
## Run 61 stress 0.116854
## Run 62 stress 0.1256669
## Run 63 stress 0.1362042
## Run 64 stress 0.1560825
## Run 65 stress 0.1196326
## Run 66 stress 0.1239429
## Run 67 stress 0.1328143
## Run 68 stress 0.1274469
## Run 69 stress 0.144338
## Run 70 stress 0.1256622
## Run 71 stress 0.1375823
## Run 72 stress 0.1282094
## Run 73 stress 0.1342375
## Run 74 stress 0.1381505
## Run 75 stress 0.1133786
## ... Procrustes: rmse 0.001400923  max resid 0.01133635
## Run 76 stress 0.1221735
## Run 77 stress 0.1236562
## Run 78 stress 0.1368292
## Run 79 stress 0.1133506
## ... Procrustes: rmse 0.0002807599  max resid 0.001269059
## ... Similar to previous best
## *** Solution reached
```

```
tyc.points <- data.frame(tyc.nms$points)
tyc.plot <- ggplot(tyc.points, aes(x=MDS1, y=MDS2,label=rownames(tyc)))

x <- max(tyc.points$MDS1)/1.5
y <- min(tyc.points$MDS2)


tyc.plot + geom_point(aes(fill=factor(alpha.expdata$detail.type), shape=factor(alpha.e
xpdata$urban)), size=4)+
    theme_bw()+
    scale_fill_manual(values=treat.col, name= "Transect Location")+
    scale_shape_manual(values=urban.shape, name="Transect Location")+    annotate("tex
t", x, y, label=paste("stress = ", round(tyc.nms$stress, digits = 3)))+
    labs(x="", y="")
```



```
ggsave(file="tyc.detail.type.nms.jpg")
```

```
## Saving 7 x 5 in image
```

Hypothesis testing on beta diversity

Permanova results that I look at significance (Pr(>F)) and the R2 which can be interpreted as roughly the % varibility in your whole community that can be explained by the model factor

```
permanova <- adonis(as.dist(jc)~alpha.expdata$detail.type, perm=99, rm.na=TRUE)
permanova
```

```
##
## Call:
## adonis(formula = as.dist(jc) ~ alpha.expdata$detail.type, permutations = 99,      r
m.na = TRUE)
##
## Permutation: free
## Number of permutations: 99
##
## Terms added sequentially (first to last)
##
##                            Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## alpha.expdata$detail.type   5     4.985 0.99697  2.6735 0.12335   0.01 **
## Residuals                  95    35.426 0.37290         0.87665
## Total                     100    40.411                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
permanova <- adonis(as.dist(bc)~alpha.expdata$detail.type*alpha.expdata$urban, perm=99
, rm.na=TRUE)

permanova
```

```
##
## Call:
## adonis(formula = as.dist(bc) ~ alpha.expdata$detail.type * alpha.expdata$urban,
permutations = 99, rm.na = TRUE)
##
## Permutation: free
## Number of permutations: 99
##
## Terms added sequentially (first to last)
##
##                                                Df SumsOfSqs MeanSqs
## alpha.expdata$detail.type                         5     9.250 1.85003
## alpha.expdata$urban                               1     1.489 1.48931
## alpha.expdata$detail.type:alpha.expdata$urban     5     2.513 0.50254
## Residuals                                        89    22.601 0.25394
## Total                                           100    35.853
##                                               F.Model      R2 Pr(>F)
## alpha.expdata$detail.type                      7.2853 0.25800   0.01 **
## alpha.expdata$urban                            5.8648 0.04154   0.01 **
## alpha.expdata$detail.type:alpha.expdata$urban  1.9790 0.07008   0.01 **
## Residuals                                              0.63037
## Total                                                  1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
permanova <- adonis(as.dist(tyc)~alpha.expdata$detail.type*alpha.expdata$urban, perm=9
9, rm.na=TRUE)
permanova
```

```
##
## Call:
## adonis(formula = as.dist(tyc) ~ alpha.expdata$detail.type * alpha.expdata$urban,
permutations = 99, rm.na = TRUE)
##
## Permutation: free
## Number of permutations: 99
##
## Terms added sequentially (first to last)
##
##                                                Df SumsOfSqs MeanSqs
## alpha.expdata$detail.type                       5    10.368 2.07364
## alpha.expdata$urban                             1     1.623 1.62325
## alpha.expdata$detail.type:alpha.expdata$urban   5     2.967 0.59348
## Residuals                                      89    24.103 0.27082
## Total                                         100    39.062
##                                               F.Model      R2 Pr(>F)
## alpha.expdata$detail.type                      7.6570 0.26543   0.01 **
## alpha.expdata$urban                            5.9939 0.04156   0.01 **
## alpha.expdata$detail.type:alpha.expdata$urban  2.1914 0.07597   0.01 **
## Residuals                                             0.61704
## Total                                                 1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Indicator species

Method for pulling out the species that coorespond with grouping variable (i.e. Treatment). See DeCaceres and Legendre 2009 for details.

```
# otu[,-1] tells multipatt to ignore the first column which holds the name


indic <- multipatt(otu[,-1], as.factor(alpha.expdata$detail.type),control = how(nperm=
99))

write.csv(file="indicator.species.csv",indic$sign%>%
    rownames_to_column(var = "OTU")%>%
    mutate(p.fdr = round(p.adjust(p.value, "fdr"),3))%>%
    right_join(taxa, by = "OTU")%>%
    filter(p.fdr < 0.05) %>%
    arrange(index))
```

```
## Warning in right_join_impl(x, y, by$x, by$y, suffix$x, suffix$y): joining
## character vector and factor, coercing into character vector
```

```
sig.otus <- indic$sign%>%
    add_rownames(var="OTU")%>%
    filter(p.adjust(p.value,"fdr") < 0.05)
```

```
## Warning: Deprecated, use tibble::rownames_to_column() instead.
```

```
otu.indic <- otu[,which(names(otu) %in% sig.otus$OTU)]
taxa.indic <- taxa[which(taxa$OTU %in% sig.otus$OTU),]
```