R统计分析-主坐标分析（PCoA）及作图方法示例

此处结合微生物群落研究中的16S扩增子分析数据，给大家分享怎样在R中进行主坐标分析（PCoA），顺便使用此处的PCoA排序结果，给大家展示怎样结合ggplot2绘制“好看”的PCoA排序图。

在R中，可用于进行PCoA分析的R包有很多可供选择，如“vegan”、“ade4”等，这些均是在生态统计中常用的R包。此处作为示例介绍其中的“vegan”包。

首先介绍示例数据。我们此处共有96个16S测序样本，均来自土壤。这96个样本共涉及了4个采样地点（地点A、B、C、D）；2种处理梯度，即添加某化学物质的低浓度处理（low）以及高浓度处理（high）；4个采样时期（时期1、2、3、4）；对于每个采样地的每种处理梯度的每个时期下，各自进行了3个重复，共计4×2×4×3=96。

此处我们希望通过PCoA分析，查看样本间细菌群落组成是否具有显著不同。

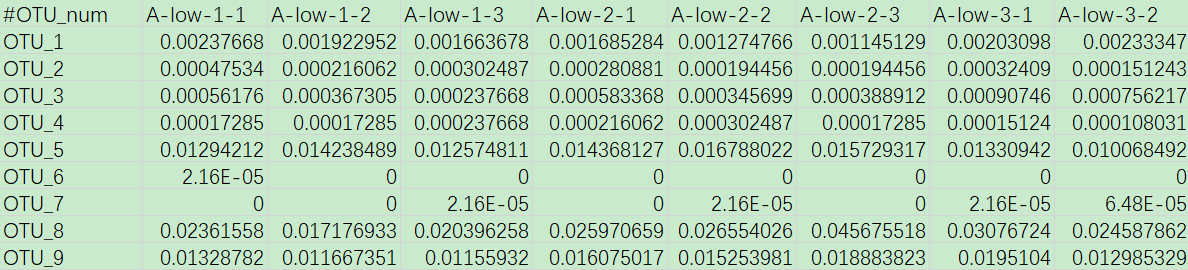
作图示例文件、R脚本等，已上传至百度盘，无提取码

<https://pan.baidu.com/s/1mL92U9DxMZKjqTNRhDW9ZA>

## 示例文件简要

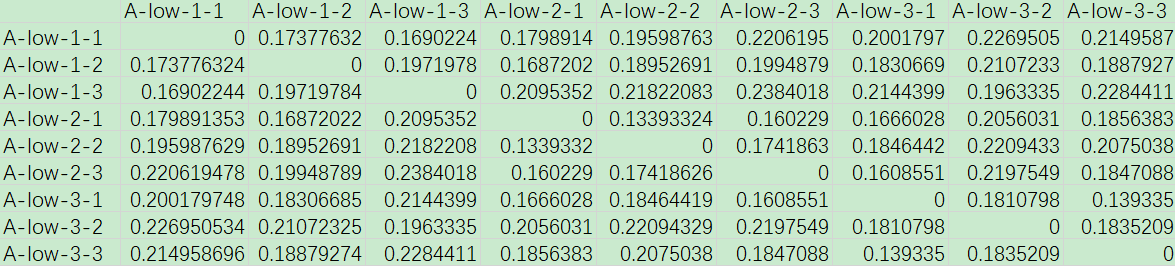
文件“otu\_table.txt”为OTU丰度表格，其内容展示如下。

每一列为一个样本，每一行为一种OTU，交叉区域为每种OTU在各样本中的丰度。



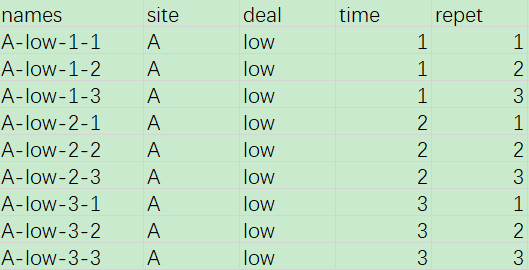
文件“bray.txt”为提前计算得到的样本距离矩阵文件（此处展示的是样本间Bray-curtis距离），其内容展示如下。

每一列为一个样本，每一行为一个样本，交叉区域为样本间的Bray-curtis距离（取值范围0-1，越接近于1表明样本间细菌群落组成差异越大）。



文件“group.txt”为样本分组信息，其内容展示如下。

第一列（names）为各样本名称；第二列（site）为各样本的采样地点，即4个采样地点（地点A、B、C、D）；第三列（deal）为2种处理梯度，即添加某化学物质的低浓度处理（low）以及高浓度处理（high）；。第四列（time）各样本的4个采样时期（时期1、2、3、4）；第五列（repet）为每个采样地的每种处理梯度的每个时期下各自进行的3个重复（1、2、3）。



## 使用vegan包进行PCoA排序分析

首先导入数据。我们可选导入原始的OTU丰度表格文件，也可使用已经计算好的样本距离矩阵文件，同时导入样本分组文件。

#OTU 丰度表

otu <- read.delim('otu\_table.txt', row.names=1, sep = '\t', stringsAsFactors = F, check.names=F)

otu <- data.frame(t(otu))

#或者现有的距离矩阵

dis <- read.delim('bray.txt', row.names=1, sep = '\t', stringsAsFactors = F, check.names=F)

#样本分组文件

group <- read.delim('group.txt', sep = '\t', stringsAsFactors = F)

然后加载vegan包，并进行PCoA分析。

library(vegan)

#排序（基于 OTU 丰度表）

distance <- vegdist(otu, method = 'bray')

pcoa <- cmdscale(distance, k = (nrow(otu) - 1), eig = TRUE)

#或者（基于现有的距离矩阵）

pcoa <- cmdscale(as.dist(dis), k = (nrow(dis) - 1), eig = TRUE)

若我们之前导入的是OTU丰度表，则我们需要首先根据OTU的丰度组成，计算样本间距离，然后使用计算好的样本间距离对样本进行PCoA排序。vegdist()用于计算样本间距离，此处使用Bray-curtis距离；cmdscale()用于PCoA排序。

若我们之前导入的是现有的距离矩阵，则我们可直接基于先有距离对样本进行PCoA排序。首先使用as.dist ()转化读入的矩阵，并使用cmdscale()进行PCoA排序。

此处展示了两种过程，若两种过程中所使用的“距离类型”是一致的，则最后所得结果也是相同的。

可以简要地查看结果。

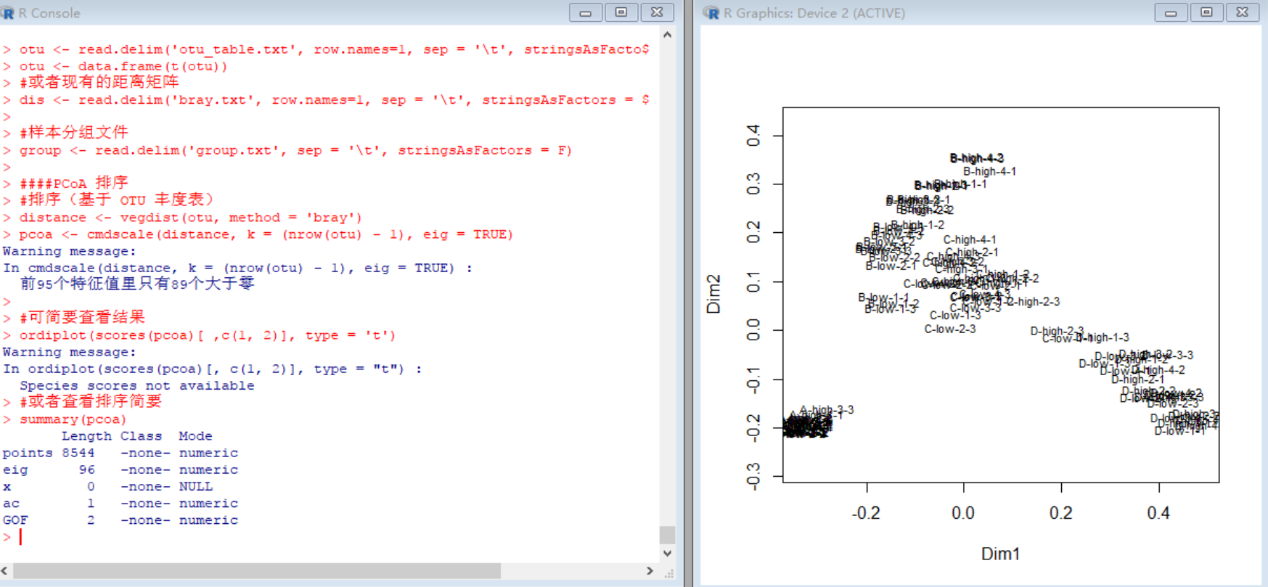
#使用vegan内置命令ordiplot()简要做图展示

ordiplot(scores(pcoa)[ ,c(1, 2)], type = 't')

#或者查看排序简要

summary(pcoa)

vegan内置命令ordiplot()方便我们直接查看排序结果。若结果可观，我们可以考虑将排序结果中的各项指标提取出，绘制效果更好的排序图（例如借助ggplot2）。



summary(pcoa)的打印结果中，给出了排序结果中的各项重要指标。

主要关注两个重要指标，eig记录了PCoA排序结果中，主要排序轴的特征值（再除以特征值总和就是各轴的解释量）；points记录了各样本在各排序轴中的坐标值。

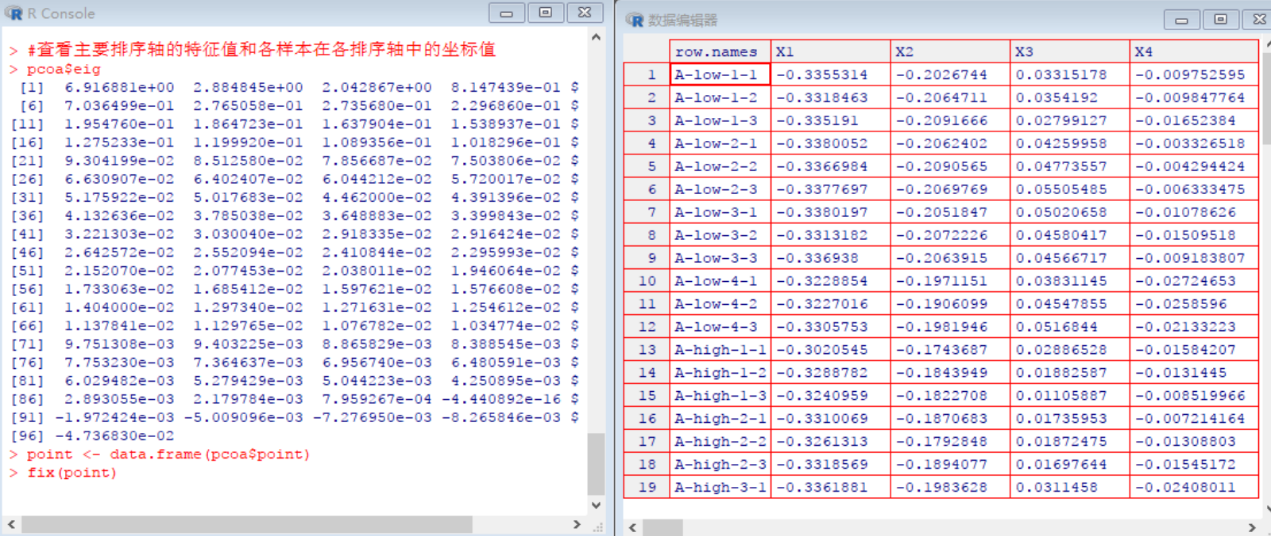
#查看主要排序轴的特征值和各样本在各排序轴中的坐标值

pcoa$eig

point <- data.frame(pcoa$point)

#可将样本坐标转化为数据框后导出，例如导出为 csv 格式

write.csv(point, 'pcoa.sample.csv')



我们还可使用vegan包中的命令wascores()，得到各OTU的排序坐标。因OTU数据量较大，因此在这里只展示前两个排序轴。

#可使用 wascores() 计算物种坐标

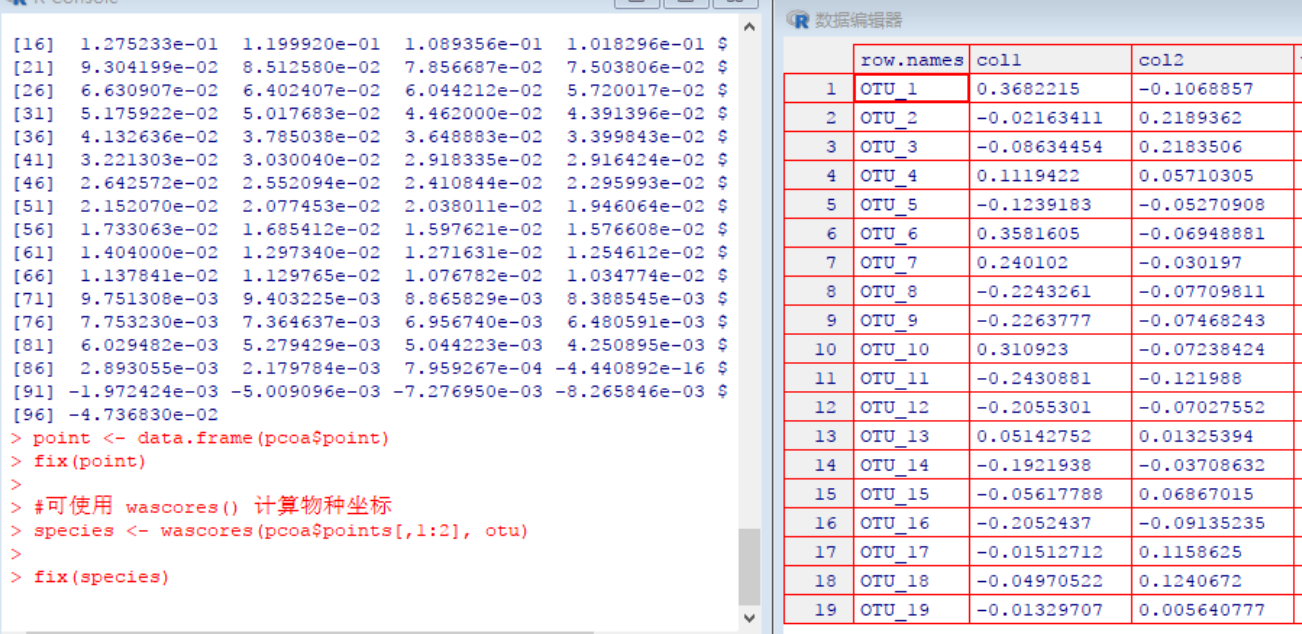
species <- wascores(pcoa$points[,1:2], otu)

#可将物种坐标转化为数据框后导出，例如导出为 csv 格式

write.csv(species, 'pcoa.otu.csv')

此处使用到了PCoA样本排序坐标数据，以及原始的OTU丰度表格文件。

计算所得结果如下。



## 使用ggplot2包进行PCoA作图

一般PCoA作图时，只展示前两个主要的轴（视情况而定，有时会展示出第三轴、第四轴等）。本次示例中，我们考虑将前两个轴的排序坐标和解释量提取出，同时将排序结果与样本分组信息合并。

#坐标轴解释量（前两轴）

pcoa\_eig <- (pcoa$eig)[1:2] / sum(pcoa$eig)

#提取样本点坐标（前两轴）

sample\_site <- data.frame({pcoa$point})[1:2]

sample\_site$names <- rownames(sample\_site)

names(sample\_site)[1:2] <- c('PCoA1', 'PCoA2')

#为样本点坐标添加分组信息

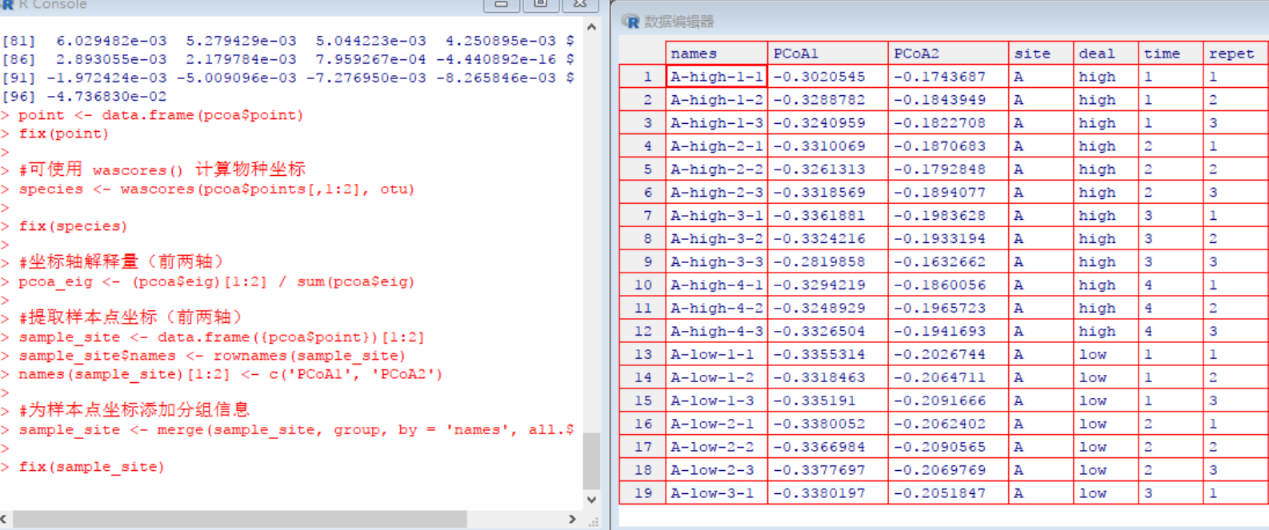
sample\_site <- merge(sample\_site, group, by = 'names', all.x = TRUE)

#可选输出，例如输出为 csv 格式

write.csv(sample\_site, 'sample\_site.csv', quote = F)

我们将前两个轴的排序坐标提取出，转换为数据框赋值给“sample\_site”，并将两个轴命名为“PCoA1”和“PCoA2”。然后，根据“names”列（样本名称列），将各样本排序结果与分组信息一一对应。

此时的数据框“sample\_site”记录了各样本的PCoA排序结果（第一轴和第二轴坐标）以及各样本的分组信息。



我们将各分组类型转化为因子数据，方便作图识别。

同时调用plyr包，计算“site”分组（样本采样来源地A、B、C、D）中的样本顶点坐标。这么做的目的：本示例数据中，在影响细菌群落组成的因素中，土壤类型是最主要的因素，因此4种采样地间的细菌群落组成差异最大；因此我们计算“顶点坐标”，以方便后续绘图时使用多边形标注最明显的分组。

sample\_site$site <- factor(sample\_site$site, levels = c('A', 'B', 'C', 'D'))

sample\_site$deal <- factor(sample\_site$deal, levels = c('low', 'high'))

sample\_site$time <- factor(sample\_site$time, levels = c('1', '2', '3', '4'))

library(plyr)

group\_border <- ddply(sample\_site, 'site', function(df) df[chull(df[[2]], df[[3]]), ])

#注：group\_border 作为下文 geom\_polygon() 的做图数据使用

然后使用ggplot2进行PCoA排序图绘制。

此处分组较多，因此在本示例中，考虑使用多边形区域展示不同采样来源（绘制方法可参考<http://blog.sciencenet.cn/home.php?mod=space&uid=3406804&do=blog&id=1155528>），使用两种形状区分2种梯度的处理，使用渐变颜色区分4个采样时期。

library(ggplot2)

pcoa\_plot <- ggplot(sample\_site, aes(PCoA1, PCoA2, group = site)) +

theme(panel.grid = element\_line(color = 'gray', linetype = 2, size = 0.1), panel.background = element\_rect(color = 'black', fill = 'transparent'), legend.key = element\_rect(fill = 'transparent')) + #去掉背景框

geom\_vline(xintercept = 0, color = 'gray', size = 0.4) +

geom\_hline(yintercept = 0, color = 'gray', size = 0.4) +

geom\_polygon(data = group\_border, aes(fill = site)) + #绘制多边形区域

geom\_point(aes(color = time, shape = deal), size = 1.5, alpha = 0.8) + #可在这里修改点的透明度、大小

scale\_shape\_manual(values = c(17, 16)) + #可在这里修改点的形状

scale\_color\_manual(values = c('yellow', 'orange', 'red', 'red4')) + #可在这里修改点的颜色

scale\_fill\_manual(values = c('#C673FF2E', '#73D5FF2E', '#49C35A2E', '#FF985C2E')) + #可在这里修改区块的颜色

guides(fill = guide\_legend(order = 1), shape = guide\_legend(order = 2), color = guide\_legend(order = 3)) + #设置图例展示顺序

labs(x = paste('PCoA axis1: ', round(100 \* pcoa\_eig[1], 2), '%'), y = paste('PCoA axis2: ', round(100 \* pcoa\_eig[2], 2), '%')) +

#可通过修改下面四句中的点坐标、大小、颜色等，修改“A、B、C、D”标签

annotate('text', label = 'A', x = -0.31, y = -0.15, size = 5, colour = '#C673FF') +

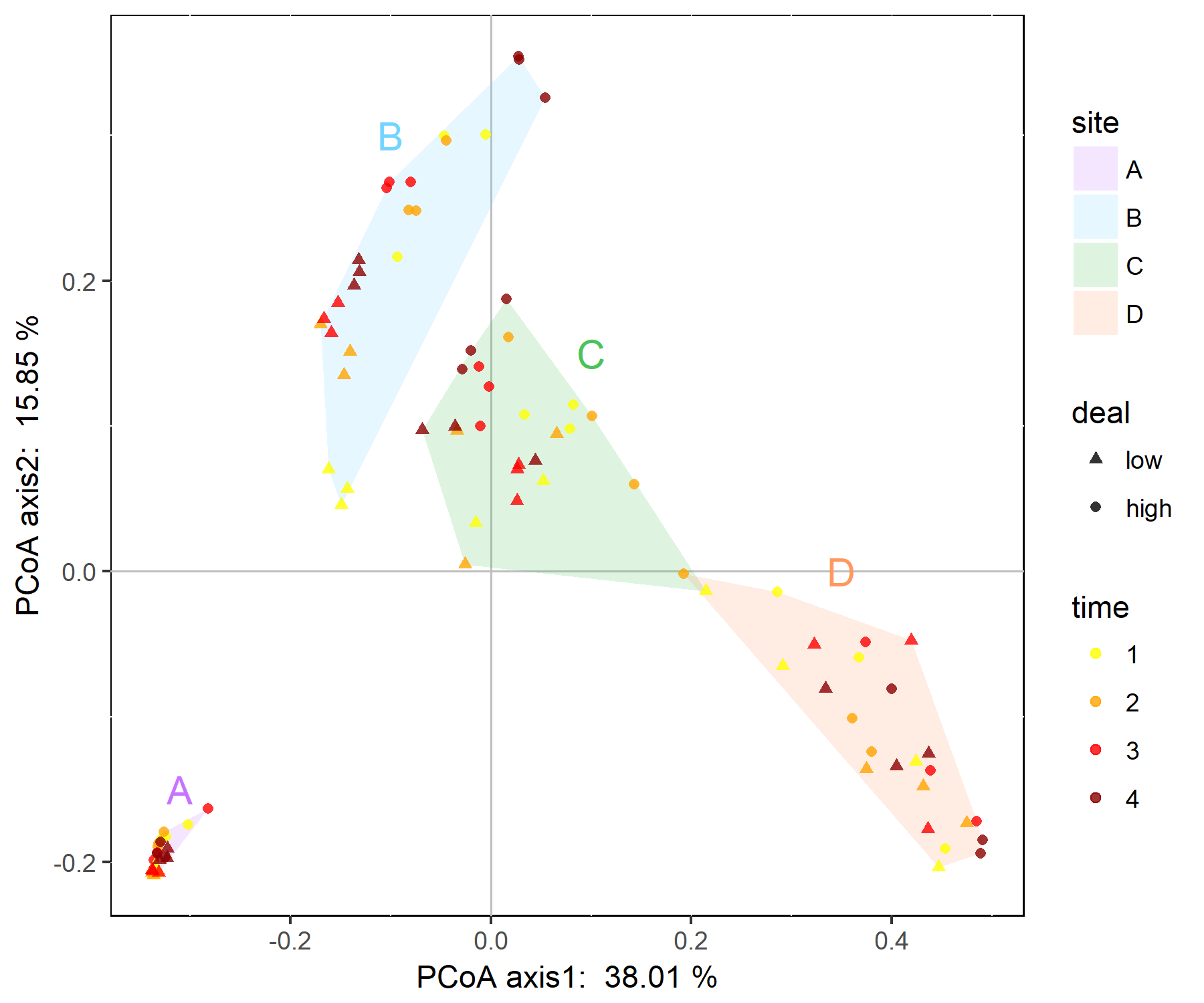
annotate('text', label = 'B', x = -0.1, y = 0.3, size = 5, colour = '#73D5FF') +

annotate('text', label = 'C', x = 0.1, y = 0.15, size = 5, colour = '#49C35A') +

annotate('text', label = 'D', x = 0.35, y = 0, size = 5, colour = '#FF985C')

ggsave('PCoA.png', pcoa\_plot, width = 6, height = 5)

ggplot2的各细节不再详细说明，最终输出结果如下。



## 参考文献

DanielBorcard, FranoisGillet, PierreLegendre, et al. 数量生态学:R语言的应用（赖江山 译）. 高等教育出版社, 2014.