

## 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  $4 \times 2 \times 4 \times 3 = 96$ 。

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。

### 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。

#OTU_num	A-low-1-1	A-low-1-2	A-low-1-3	A-low-2-1	A-low-2-2	A-low-2-3	A-low-3-1	A-low-3-2
OTU_1	0.00237668	0.001922952	0.001663678	0.001685284	0.001274766	0.001145129	0.00203098	0.00233347
OTU_2	0.00047534	0.000216062	0.000302487	0.000280881	0.000194456	0.000194456	0.00032409	0.000151243
OTU_3	0.00056176	0.000367305	0.000237668	0.000583368	0.000345699	0.000388912	0.00090746	0.000756217
OTU_4	0.00017285	0.00017285	0.000237668	0.000216062	0.000302487	0.00017285	0.00015124	0.000108031
OTU_5	0.01294212	0.014238489	0.012574811	0.014368127	0.016788022	0.015729317	0.01330942	0.010068492
OTU_6	2.16E-05	0	0	0	0	0	0	0
OTU_7	0	0	2.16E-05	0	2.16E-05	0	2.16E-05	6.48E-05
OTU_8	0.02361558	0.017176933	0.020396258	0.025970659	0.026554026	0.045675518	0.03076724	0.024587862
OTU_9	0.01328782	0.011667351	0.01155932	0.016075017	0.015253981	0.018883823	0.0195104	0.012985329

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）。

	A-low-1-1	A-low-1-2	A-low-1-3	A-low-2-1	A-low-2-2	A-low-2-3	A-low-3-1	A-low-3-2	A-low-3-3
A-low-1-1		0	0.17377632	0.1690224	0.1798914	0.19598763	0.2206195	0.2001797	0.2269505
A-low-1-2	0.173776324		0	0.1971978	0.1687202	0.18952691	0.1994879	0.1830669	0.2107233
A-low-1-3	0.16902244	0.19719784		0	0.2095352	0.21822083	0.2384018	0.2144399	0.1963335
A-low-2-1	0.179891353	0.16872022	0.2095352		0	0.13393324	0.160229	0.1666028	0.2056031
A-low-2-2	0.195987629	0.18952691	0.2182208	0.1339332		0	0.1741863	0.1846442	0.2209433
A-low-2-3	0.220619478	0.19948789	0.2384018	0.160229	0.17418626		0	0.1608551	0.2197549
A-low-3-1	0.200179748	0.18306685	0.2144399	0.1666028	0.18464419	0.1608551		0	0.1810798
A-low-3-2	0.226950534	0.21072325	0.1963335	0.2056031	0.22094329	0.2197549	0.1810798		0
A-low-3-3	0.214958696	0.18879274	0.2284411	0.1856383	0.2075038	0.1847088	0.139335	0.1835209	

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）。

names	site	deal	time	repet
A-low-1-1	A	low	1	1
A-low-1-2	A	low	1	2
A-low-1-3	A	low	1	3
A-low-2-1	A	low	2	1
A-low-2-2	A	low	2	2
A-low-2-3	A	low	2	3
A-low-3-1	A	low	3	1
A-low-3-2	A	low	3	2
A-low-3-3	A	low	3	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)
```



	row.names	X1	X2	X3	X4
1	A-low-1-1	-0.3355314	-0.2026744	0.03315178	-0.009752595
2	A-low-1-2	-0.3318463	-0.2064711	0.0354192	-0.009847764
3	A-low-1-3	-0.335191	-0.2091666	0.02799127	-0.01652384
4	A-low-2-1	-0.3380052	-0.2062402	0.04259958	-0.003326518
5	A-low-2-2	-0.3366984	-0.2090565	0.04773557	-0.004294424
6	A-low-2-3	-0.3377697	-0.2069769	0.05050485	-0.006333475
7	A-low-3-1	-0.3380197	-0.2051847	0.05020658	-0.01078626
8	A-low-3-2	-0.3313182	-0.2072226	0.04580417	-0.01509518
9	A-low-3-3	-0.336938	-0.2063915	0.04566717	-0.009183807
10	A-low-4-1	-0.3228854	-0.1971151	0.03831145	-0.02724653
11	A-low-4-2	-0.3227016	-0.1906099	0.04547855	-0.0258596
12	A-low-4-3	-0.3305753	-0.1981946	0.0516844	-0.02133223
13	A-high-1-1	-0.3020545	-0.1743687	0.02886528	-0.01584207
14	A-high-1-2	-0.3288782	-0.1843949	0.01882587	-0.0131445
15	A-high-1-3	-0.3240959	-0.1822708	0.01105887	-0.008519966
16	A-high-2-1	-0.3310069	-0.1870683	0.01735953	-0.007214164
17	A-high-2-2	-0.3261313	-0.1792848	0.01872475	-0.01308803
18	A-high-2-3	-0.3318569	-0.1894077	0.01697644	-0.01545172
19	A-high-3-1	-0.3361881	-0.1983628	0.0311458	-0.02408011

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。

	row.names	col1	col2
1	OTU_1	0.3682215	-0.1068857
2	OTU_2	-0.02163411	0.2189362
3	OTU_3	-0.08634454	0.2183506
4	OTU_4	0.1119422	0.05710305
5	OTU_5	-0.1239183	-0.05270908
6	OTU_6	0.3581605	-0.06948881
7	OTU_7	0.240102	-0.030197
8	OTU_8	-0.2243261	-0.07709811
9	OTU_9	-0.2263777	-0.07468243
10	OTU_10	0.310923	-0.07238424
11	OTU_11	-0.2430881	-0.121988
12	OTU_12	-0.2055301	-0.07027552
13	OTU_13	0.05142752	0.01325394
14	OTU_14	-0.1921938	-0.03708632
15	OTU_15	-0.05617788	0.06867015
16	OTU_16	-0.2052437	-0.09135235
17	OTU_17	-0.01512712	0.1158625
18	OTU_18	-0.04970522	0.1240672
19	OTU_19	-0.01329707	0.005640777

## 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)

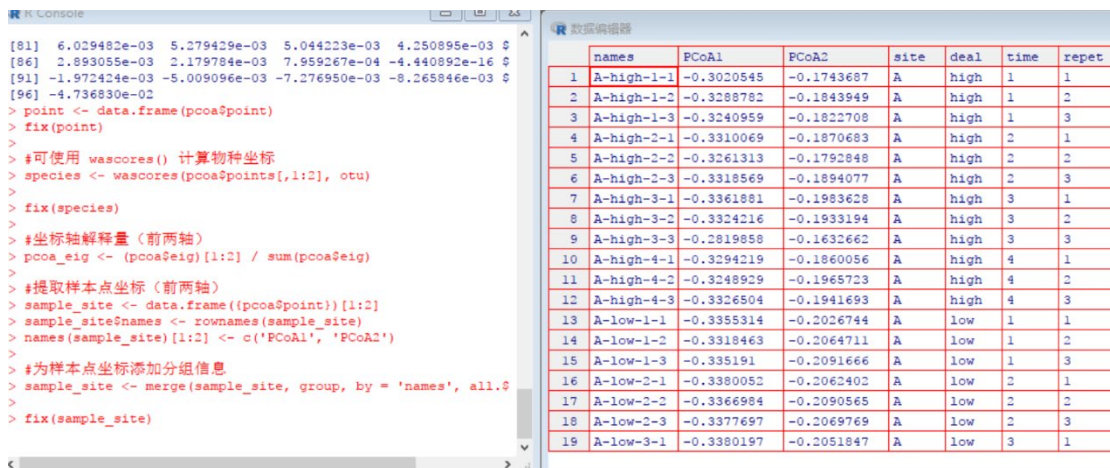
#提取样本点坐标（前两轴）
sample_site <- data.frame(pcoa$point)[1:2]
sample_site$names <- rownames(sample_site)
names(sample_site)[1:2] <- c('PCoA1', 'PCoA2')

#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。



The screenshot shows an R console on the left and a data viewer on the right. The console displays the execution of R code for PCoA analysis, including calculating eigenvalues, extracting coordinates, and merging sample names and group information. The data viewer on the right displays a table with 19 rows and 7 columns: names, PCoA1, PCoA2, site, deal, time, and repet.

	names	PCoA1	PCoA2	site	deal	time	repet
1	A-high-1-1	-0.3020545	-0.1743687	A	high	1	1
2	A-high-1-2	-0.3288782	-0.1843949	A	high	1	2
3	A-high-1-3	-0.3240959	-0.1822708	A	high	1	3
4	A-high-2-1	-0.3310069	-0.1870683	A	high	2	1
5	A-high-2-2	-0.3261313	-0.1792848	A	high	2	2
6	A-high-2-3	-0.3318569	-0.1894077	A	high	2	3
7	A-high-3-1	-0.3361881	-0.1983628	A	high	3	1
8	A-high-3-2	-0.3324216	-0.1933194	A	high	3	2
9	A-high-3-3	-0.2819858	-0.1632662	A	high	3	3
10	A-high-4-1	-0.3294219	-0.1860056	A	high	4	1
11	A-high-4-2	-0.3248929	-0.1965723	A	high	4	2
12	A-high-4-3	-0.3326504	-0.1941693	A	high	4	3
13	A-low-1-1	-0.3355314	-0.2026744	A	low	1	1
14	A-low-1-2	-0.3318463	-0.2064711	A	low	1	2
15	A-low-1-3	-0.335191	-0.2091666	A	low	1	3
16	A-low-2-1	-0.3380052	-0.2062402	A	low	2	1
17	A-low-2-2	-0.3366984	-0.2090565	A	low	2	2
18	A-low-2-3	-0.3377697	-0.2069769	A	low	2	3
19	A-low-3-1	-0.3380197	-0.2051847	A	low	3	1

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 <- dplyr::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。

```

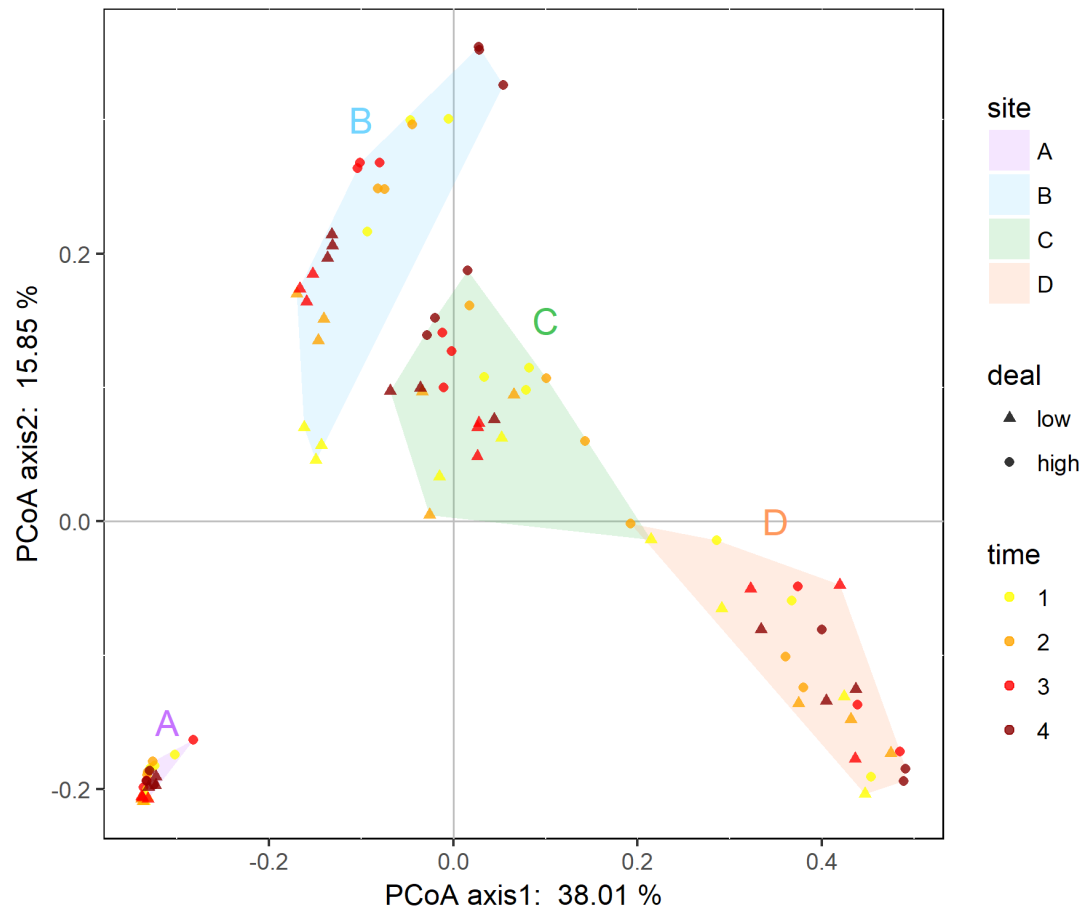
library(ggplot2)

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

Daniel Borcard, François Gillet, Pierre Legendre, 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.