TSNE

白德凤：中国农业科学院深圳基因组研究所

2024-04-17

### tSNE数据降维及绘图

本节作者：白德凤：中国农业科学院深圳基因组研究所

版本1.0.0，更新日期：2024年4月17日

#### 什么是tSNE (t-Distributed Stochastic Neighbor Embedding)

#### tSNE简介和基本原理

tSNE是数据降维方式中的一种，其由T分布和随机近邻嵌入，其可以讲高维数据降到2-3维。t-SNE把高维度的数据点之间的距离转化为高斯分布概率，在高维度用高斯分布概率表示相似性，在低维度用t分布表示相似性，然后设置一个惩罚函数，实现降低维度但是保留一定局部特征的方式。

#### tSNE实例

美国纪念斯隆凯瑟琳癌症研究中心的Marcel R.M. van den Brink团队2023年发表在Cell上的论文(Nguyen et al., 2023)，研究药物、肠道微生物组和癌症患者死亡之间的关系。我们以此图为例进行讲解。

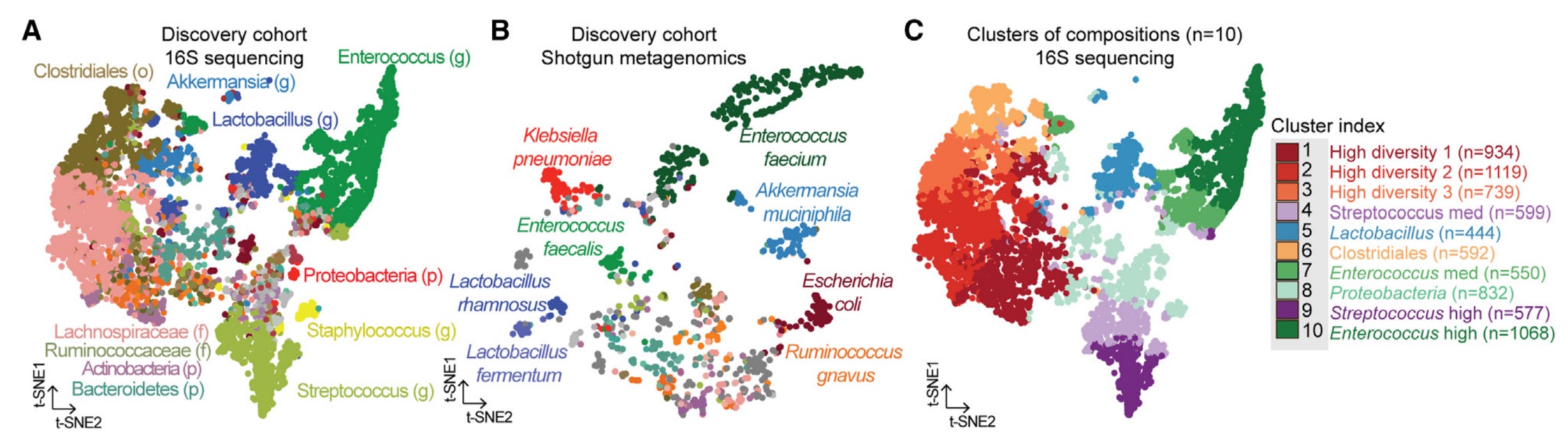


图 2：（A 和 B）通过 tSNE 投影显示 MSKCC 发现队列中肠道微生物群的组成空间。每个点代表一个样本，根据 (A) 16S rRNA（7,454 个样本；778 名患者）或 (B) 宏基因组测序图谱（980 个样本；340 名患者）（p，门；f，科；o，目；g，属）中相对丰度最高的类群着色。样本采集于相对于 HCT 的第 -30 天和第 2,205 天之间。(C)通过 k-means 无监督聚类法划分出十个肠道微生物组聚类群。

Figure 2. (A and B) Compositional space of the intestinal microbiota in the MSKCC discovery cohort visualized by tSNE projection. Each point represents a sample, colored according to the taxon of highest relative abundance based on (A) 16S rRNA (7,454 samples; 778 patients) or (B) shotgun metagenomic sequencing profiles (980 samples; 340 patients) (p, phylum; f, family; o, order; g, genus). Samples were collected between day -30 and 2,205 relative to HCT. (C) Ten clusters of intestinal microbiome compositions are assigned by k-means unsupervised clustering.

结果：对于 16S rRNA 测序样本，我们使用 BrayCurtis b-diversity 差异指数计算了发现队列样本之间的组成差异；对于霰弹枪元基因组测序样本，我们使用 BrayCurtis b-diversity 差异指数计算了物种水平的组成差异，并通过 t-stochastic neighbor embedding（tSNE；图 2A 和 2B）将高维粪便组成数据可视化。我们观察到了微生物组损伤的模式，包括α多样性的丧失和潜在致病菌的富集，如肠球菌属和肠杆菌科（图 2A 和 2B）。具体来说，我们观察到一个样本群，其中最丰富的生物是各种严格厌氧菌，如Ruminococcus gnavus或Erysipelatoclostridium ramosum，以及富含潜在致病性兼性物种的独特样本群，包括粪肠球菌、肺炎克雷伯菌和大肠埃希菌。考虑到在保留细菌群落结构的同时降低维度复杂性的数学挑战，我们对发现集样本的 Bray-Curtis b-diversity 矩阵进行了无监督 k-means 聚类，并确定了 10 个不同的微生物群（图 2C）。

We computed the compositional differences among discovery cohort samples using BrayCurtis b-diversity dissimilarity indices at the genus level for 16S rRNA-sequenced samples or at the species level for shotgun metagenomic sequenced samples and visualized the highdimensional stool composition data via t-stochastic neighbor embedding (tSNE; Figures 2A and 2B). We observed patterns of microbiome injuries, including the loss of alpha diversity and enrichment of potentially pathogenic bacteria such as Enterococcus and Enterobacteriaceae (Figures 2A and 2B).A subset of 980 samples with shotgun metagenomic profiling also showed similar patterns of microbiome injuries during allo-HCT (Figure 2B). Specifically, we observed a cluster of samples whose most abundant organisms were various strict anaerobes such as Ruminococcus gnavus or Erysipelatoclostridium ramosum, as well as distinct clusters enriched for potentially pathogenic facultative species including Enterococcus faecium, Klebsiella pneumoniae, and Escherichia coli. Given the mathematical challenge of reducing dimensionality complexity while preserving bacterial community structure, we performed unsupervised k-means clustering on the Bray-Curtis b-diversity matrix of samples in the discovery set and identified ten distinct microbiota clusters (Figure 2C).

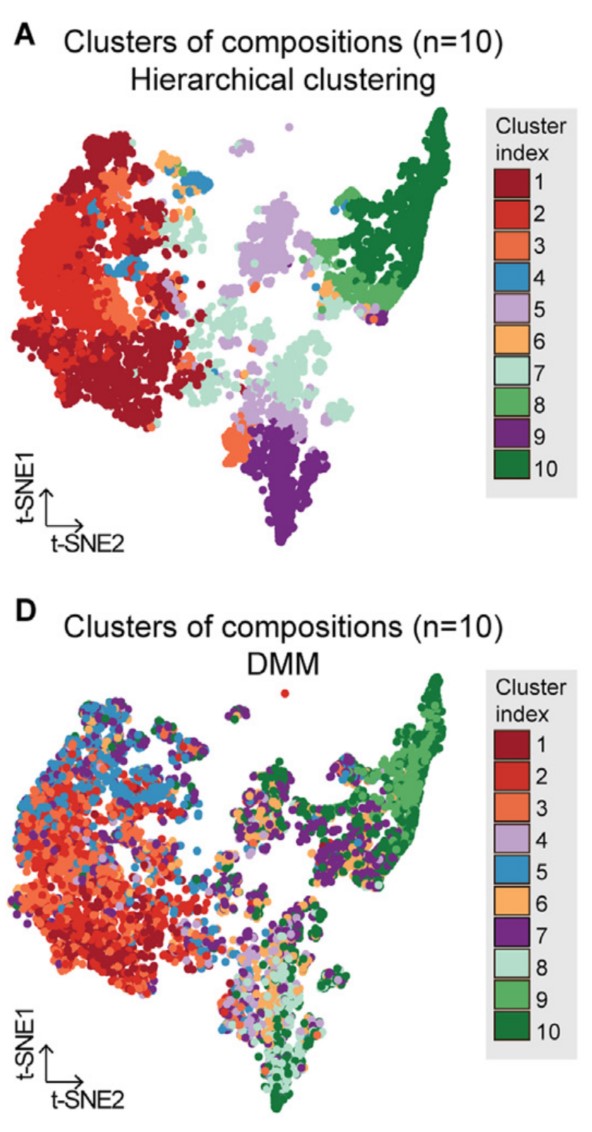


Figure S1. (A and D) Compositional space of the intestinal microbiota visualized by tSNE projection in the MSKCC discovery cohort. Each dot represents a sample, colored according to clusters assigned by (A) hierarchical clustering or (D) Dirichlet multinomial mixture (DMM) model.

图 S1. (A和D）通过tSNE投影观察MSKCC发现队列中的肠道微生物群的组成空间。每个点代表一个样本，根据(A)分层聚类或(D)Dirichlet 多叉混合物（DMM）模型分配的聚类着色。

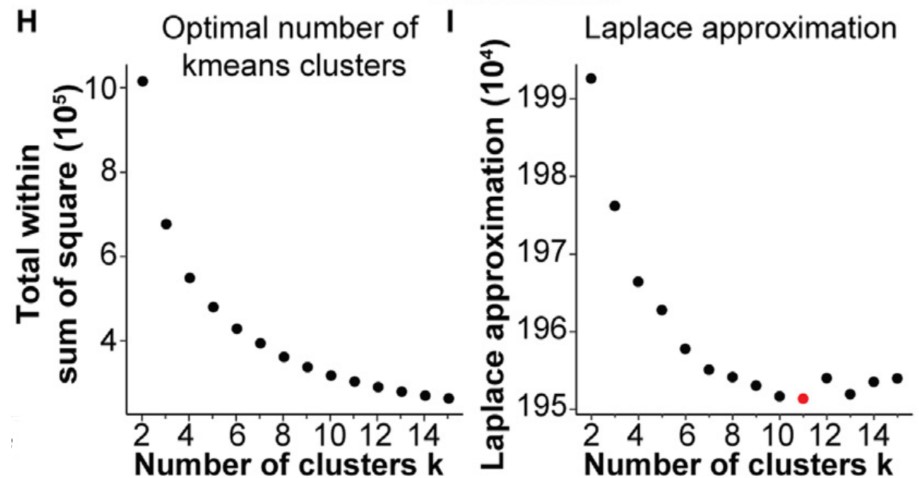
 Figure S1. (H) Optimal number of k-means clusters was estimated from the curve of within-cluster sum of square distances from each point to its cluster centroid. (I) Optimal number of clusters identified by DMM was estimated by the smallest Laplace approximation metric.

图 S1. (H)根据每个点到其聚类中心点的聚类内平方距离之和曲线估算 k-means 聚类的最佳数目。(I) 通过最小拉普拉斯近似度量估算出 DMM 确定的最佳聚类数目。

#### tSNE绘图实战

#### tSNE投影肠道微生物群的组成空间图

##### 导入需要的软件包

##### 导入数据

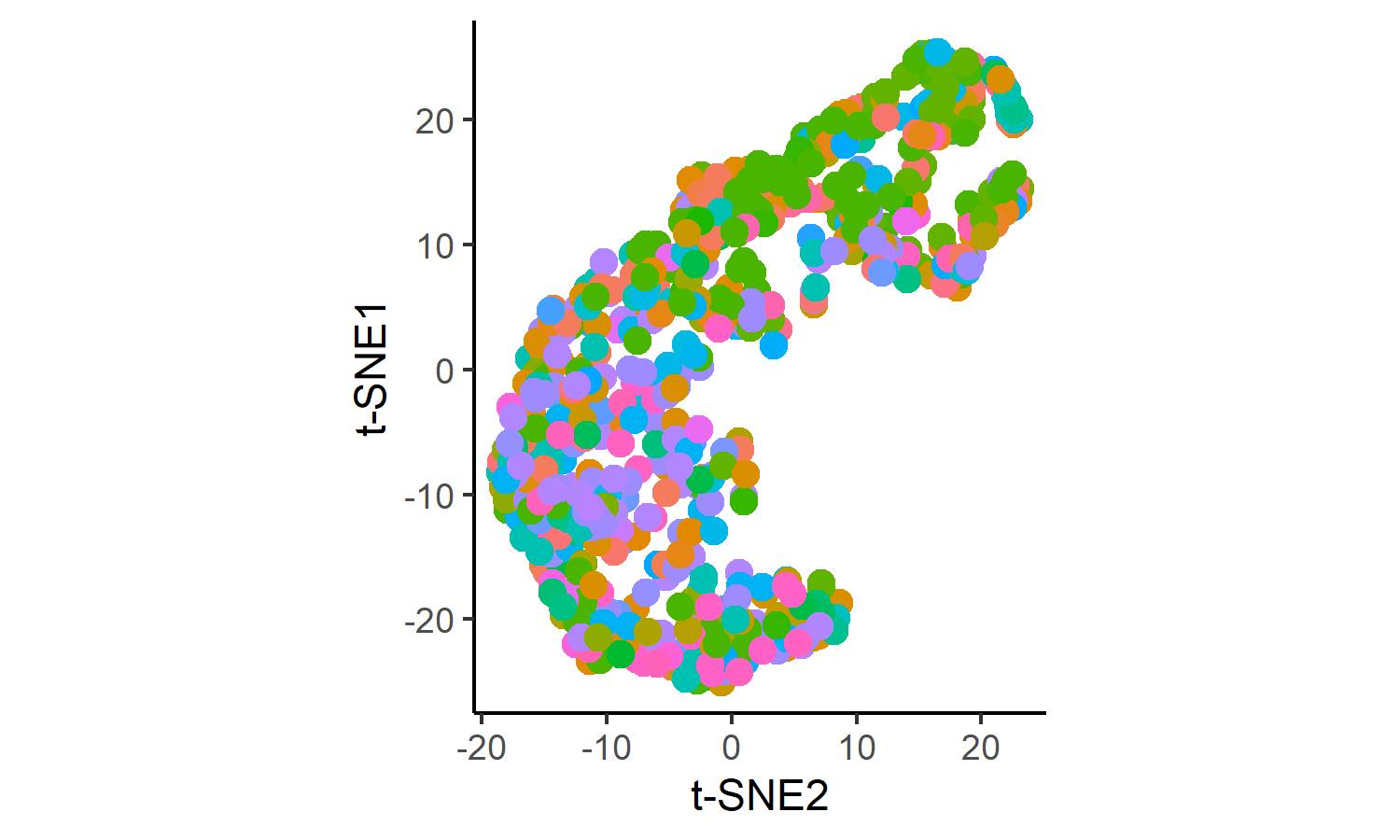
# 整体：健康组+疾病组  
#丰度数据表和metadata  
design = read.table("data/design.txt", header=T, row.names=1, sep="\t")  
design$group = design$groupID  
otufile = "data/sum\_s2.txt"  
otutab = read.table(paste(otufile, sep=""), header=T, row.names=1, sep="\t", comment.char="")   
  
#健康组  
design2 = read.table("data/design\_sh.txt", header=T, row.names=1, sep="\t")  
design2$group = design2$groupID  
otufile2 = "data/sum\_sh.txt"  
otutab2 = read.table(paste(otufile2, sep=""), header=T, row.names=1, sep="\t", comment.char="")   
  
#疾病组  
design3 = read.table("data/design\_sp.txt", header=T, row.names=1, sep="\t")  
design3$group = design3$groupID  
otufile3 = "data/sum\_sp.txt"  
otutab3 = read.table(paste(otufile3, sep=""), header=T, row.names=1, sep="\t", comment.char="")

##### 鉴定最丰富的属

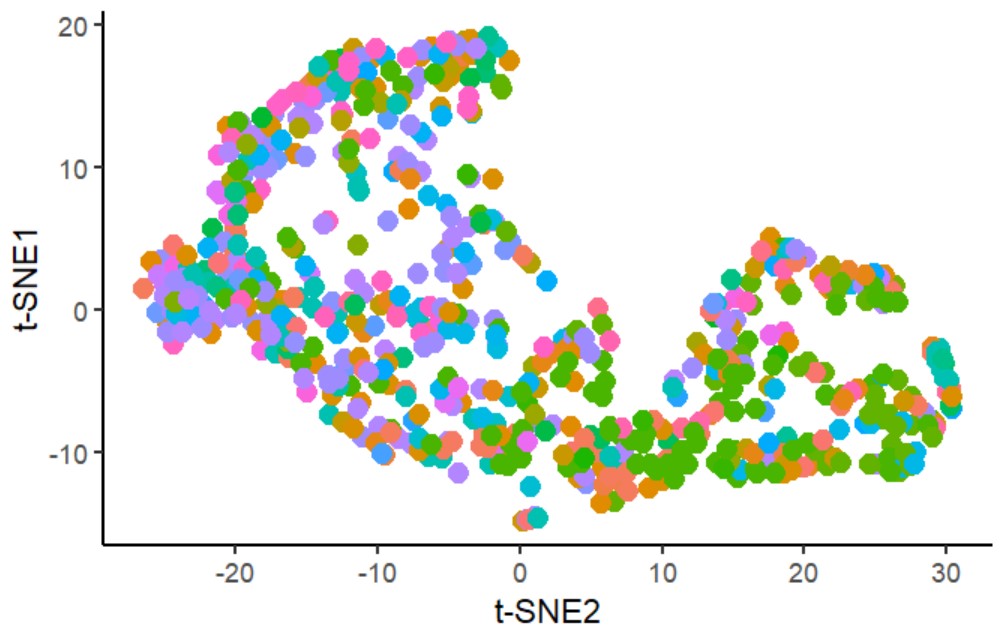
library(dplyr)  
library(Rtsne)  
#整体  
#计算得到每个样本中每个微生物属的相对丰富  
otutab <- t(otutab)  
otutab <- otutab[-1, ]  
sample = rownames(otutab)  
otutab <- cbind(sample, otutab)  
otutab <- as.data.frame(otutab)  
genus\_abundance\_tbl = otutab %>%  
 filter(sample %in% design$Samples)  
#得到每个样本优势微生物属的数据  
dominant\_genus = colnames(genus\_abundance\_tbl[,-1])[apply(genus\_abundance\_tbl[,-1], 1, which.max)]  
#dominant\_genus  
# 得到每个样本优势微生物属的数据  
design$dominant\_genus = dominant\_genus  
#计算tSNE坐标  
#读入bray-curtail Beta diversity Matrix  
bc\_data = read.table(paste("data/bray\_curtis.txt",sep=""), header=T, row.names=1, sep="\t", comment.char="")   
#bc\_data  
#利用Rtsne计算tSNE坐标  
tsne\_out <- Rtsne(bc\_data, pca=FALSE, perplexity=45, theta=0.1, max\_iter = 3000)  
#str(tsne\_out)  
#Y中存储了画图坐标  
stnes = tsne\_out$Y  
#stnes  
  
#健康组  
otutab2 <- t(otutab2)  
otutab2 <- otutab2[-1, ]  
sample2 = rownames(otutab2)  
otutab2 <- cbind(sample2, otutab2)  
otutab2 <- as.data.frame(otutab2)  
genus\_abundance\_tbl2 = otutab2 %>%  
 filter(sample2 %in% design2$Samples)  
#得到每个样本优势微生物属的数据  
dominant\_genus2 = colnames(genus\_abundance\_tbl2[,-1])[apply(genus\_abundance\_tbl2[,-1], 1, which.max)]  
#dominant\_genus2  
# 得到每个样本优势微生物属的数据  
design2$dominant\_genus2 = dominant\_genus2  
#计算tSNE坐标  
#读入bray-curtail Beta diversity Matrix  
bc\_data2 = read.table(paste("data/bray\_curtis\_sh.txt",sep=""), header=T, row.names=1, sep="\t", comment.char="")   
#bc\_data2  
#利用Rtsne计算tSNE坐标  
tsne\_out2 <- Rtsne(bc\_data2, pca=FALSE, perplexity=45, theta=0.1, max\_iter = 3000)  
#str(tsne\_out2)  
#Y中存储了画图坐标  
stnes2 = tsne\_out2$Y  
#stnes2  
  
#疾病组  
otutab3 <- t(otutab3)  
otutab3 <- otutab3[-1, ]  
sample3 = rownames(otutab3)  
otutab3 <- cbind(sample3, otutab3)  
otutab3 <- as.data.frame(otutab3)  
genus\_abundance\_tbl3 = otutab3 %>%  
 filter(sample3 %in% design3$Samples)  
#得到每个样本优势微生物属的数据  
dominant\_genus3 = colnames(genus\_abundance\_tbl3[,-1])[apply(genus\_abundance\_tbl3[,-1], 1, which.max)]  
#dominant\_genus3  
# 得到每个样本优势微生物属的数据  
design3$dominant\_genus3 = dominant\_genus3  
#计算tSNE坐标  
#读入bray-curtail Beta diversity Matrix  
bc\_data3 = read.table(paste("data/bray\_curtis\_sp.txt",sep=""), header=T, row.names=1, sep="\t", comment.char="")   
#bc\_data3  
#利用Rtsne计算tSNE坐标  
tsne\_out3 <- Rtsne(bc\_data3, pca=FALSE, perplexity=45, theta=0.1, max\_iter = 3000)  
#str(tsne\_out3)  
#Y中存储了画图坐标  
stnes3 = tsne\_out3$Y  
#stnes3

##### 绘tSNE图

#整体：健康组+疾病组  
p1 <- ggplot(design, aes(x=stnes[,2],y=stnes[,1],col=factor(dominant\_genus))) +  
 geom\_point(size = 3) +  
 #scale\_color\_manual(name = "Legend", values=color\_scheme\_genus\_vector) +  
 #color = ""+  
 theme\_classic() +  
 coord\_fixed() +  
 theme(legend.position = "none")+  
 labs(x = "t-SNE2", y= "t-SNE1")  
#ggsave(paste("data\_npc/tSNE\_all.pdf",".pdf", sep=""), p1, width=89 \* 1.5, height=50 \* 1.5, unit='mm')  
(p1)

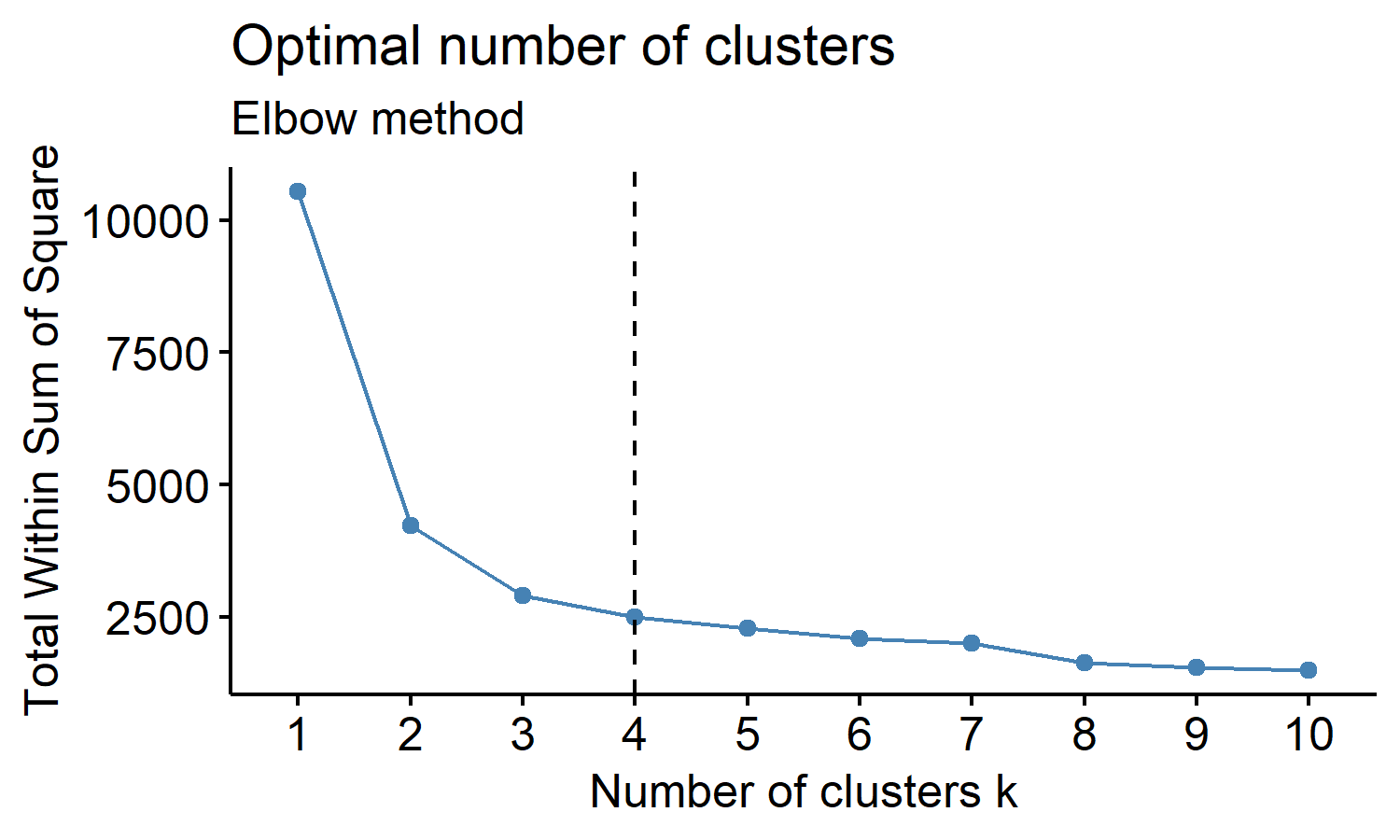
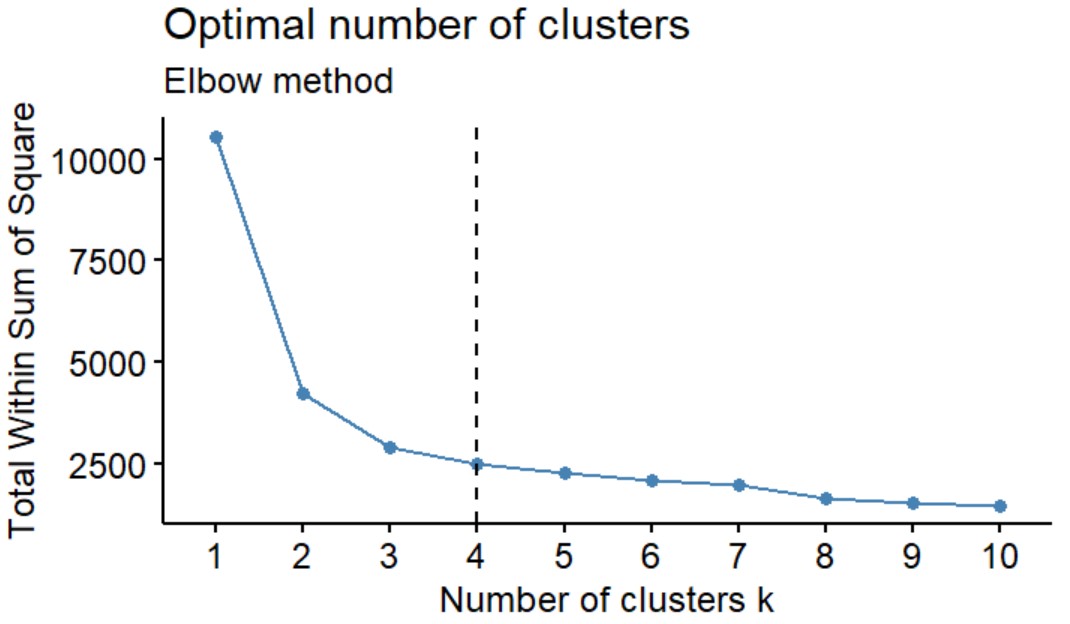


#健康组  
p2 <- ggplot(design2, aes(x=stnes2[,2],y=stnes2[,1],col=factor(dominant\_genus2))) +  
 geom\_point(size = 3) +  
 #scale\_color\_manual(name = "Legend", values=color\_scheme\_genus\_vector) +  
 #color = ""+  
 theme\_classic() +  
 coord\_fixed() +  
 theme(legend.position = "none")+  
 labs(x = "t-SNE2", y= "t-SNE1")  
#ggsave(paste("data\_npc/tSNE\_H.pdf",".pdf", sep=""), p2, width=89 \* 1.5, height=50 \* 1.5, unit='mm')  
#p2  
  
#疾病组  
p3 <- ggplot(design3, aes(x=stnes3[,2],y=stnes3[,1],col=factor(dominant\_genus3))) +  
 geom\_point(size = 3) +  
 #scale\_color\_manual(name = "Legend", values=color\_scheme\_genus\_vector) +  
 #color = ""+  
 theme\_classic() +  
 coord\_fixed() +  
 theme(legend.position = "none")+  
 labs(x = "t-SNE2", y= "t-SNE1")  
#ggsave(paste("data\_npc/tSNE\_P.pdf",".pdf", sep=""), p3, width=89 \* 1.5, height=50 \* 1.5, unit='mm')  
#p3



##### 通过kmeans方法确定最佳聚类数量

#设置工作目录  
#setwd("C:/A202308/result/beta")  
#导入用到的软件包  
library(dplyr)  
library(factoextra)  
  
#读入bray-curtail Beta diversity Matrix  
bc\_data = read.table(paste("data/bray\_curtis.txt",sep=""), header=T, row.names=1, sep="\t", comment.char="")   
#bc\_data  
  
##1.kmeans聚类  
km <- kmeans(bc\_data, 3, nstart = 24)  
#km  
  
#聚类可视化  
# fviz\_cluster(km, data = bc\_data,  
# palette = c("#A50026","#C2A5CF","#3690C0"),  
# ellipse.type = "euclid",  
# star.plot = TRUE,   
# repel = TRUE,  
# ggtheme = theme\_minimal()  
# )  
  
#确定kmeans聚类的最佳聚类数。可以在平方误差综合SSE碎石图上查找弯曲或弯头。弯头在结果图中的位置表明适合kmeans的簇数  
(fviz\_nbclust(bc\_data, kmeans, method = "wss")+  
 geom\_vline(xintercept = 4, linetype = 2)+  
 labs(subtitle = "Elbow method"))

##### DMM模型方法进行聚类群分析

##Dirichlet multinomial mixture (DMM) model实现微生物数据的聚类分析  
#参考http://www.360doc.com/content/22/0326/22/73394596\_1023477764.shtml  
#安装和导入DMM模型聚类分析需要的软件包  
library(BiocManager)  
#BiocManager::install("microbiome")  
#BiocManager::install("DirichletMultinomial")  
library(dplyr)  
library(microbiome)  
library(DirichletMultinomial)  
library(reshape2)  
library(magrittr)  
library(readxl)  
#suppress(library(readxl))  
  
# 构建phyloseq数据  
# metadata  
metadata <- data.frame(read\_xlsx('data/dietswap\_data.xlsx', sheet='samtable'))  
rownames(metadata) <- metadata$sample  
# taxtable  
tax <- data.frame(read\_xlsx('data/dietswap\_data.xlsx', sheet='taxtable'))  
# otutable  
otu <- data.frame(read\_xlsx('data/dietswap\_data.xlsx', sheet='otutable'))  
# order same as metadata  
#tax <- tax[,c('clade\_name',metadata$sample)]  
otu <- otu[,c('clade\_name',metadata$sample)]  
otu <- as.data.frame(otu)  
# make table sample x feature  
rownames(tax) <- tax$clade\_name  
tax <- data.frame(tax[,-1], check.names=FALSE)  
rownames(otu) <- otu$clade\_name  
otu <- data.frame(otu[,-1], check.names=FALSE)  
# compile count data into phyloseq objects for species  
dietswap\_data <- phyloseq(otu\_table(as.matrix(otu), taxa\_are\_rows=TRUE),  
 sample\_data(metadata),  
 tax\_table(as.matrix(tax)))  
pseq <- dietswap\_data  
  
#示例数据  
# data("dietswap")  
# #这个示例数据包含52个微生物，222个样本，样本信息文件有8列  
# pseq<-dietswap  
# # #pseq@otu\_table  
# write.csv(pseq@otu\_table, "data/dietswap\_otutable.csv")  
# write.csv(pseq@tax\_table, "data/dietswap\_taxtable.csv")  
# samdata <- as.data.frame(pseq@sam\_data)  
# write.csv(samdata, "data/dietswap\_samtable.csv")  
  
#只挑选出核心群进行分析  
pseq.comp<-microbiome::transform(pseq, "compositional")  
#核心微生物群的指标应该是在50%的样本中相对丰度为0.1%  
taxa<-core\_members(pseq.comp, detection = 0.1/100, prevalence = 50/100)  
#删除不想要的OTU或群  
pseq<-prune\_taxa(taxa, pseq)  
#数据中有三个分析组，按照这三个组分为三个部分的预设进行建模  
map = sample\_data(pseq)  
#head(map)  
map$group %>% unique()

[1] "DI" "HE" "ED"

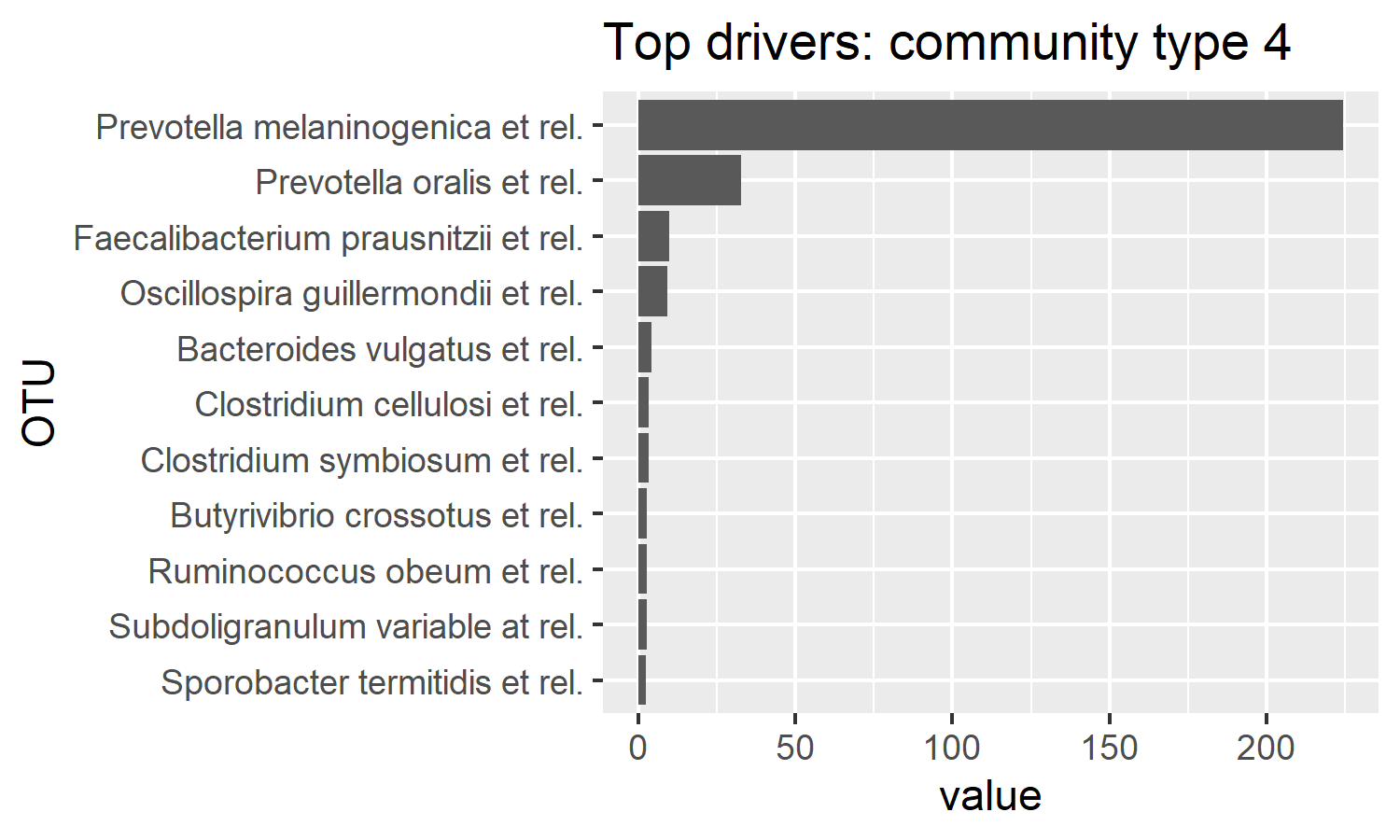
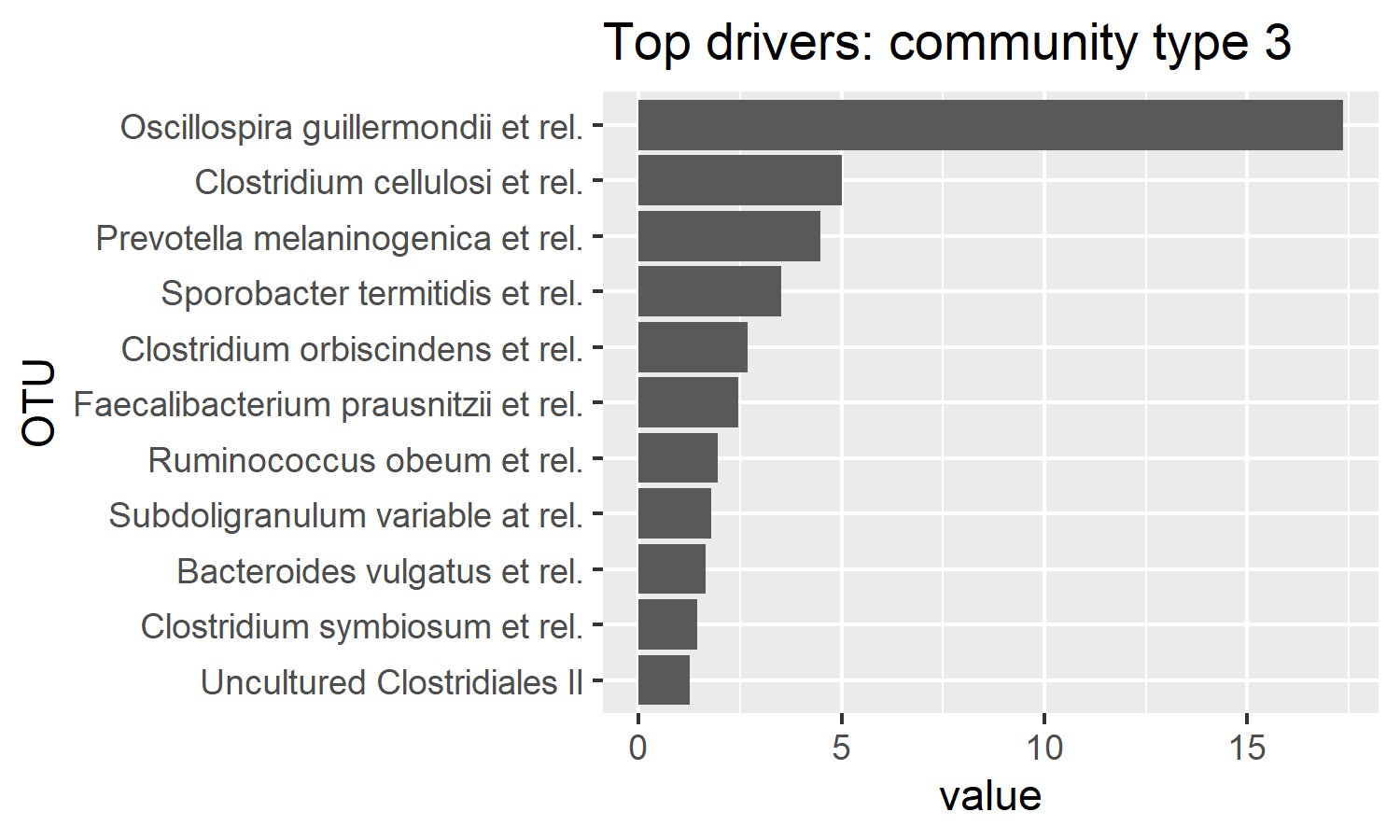
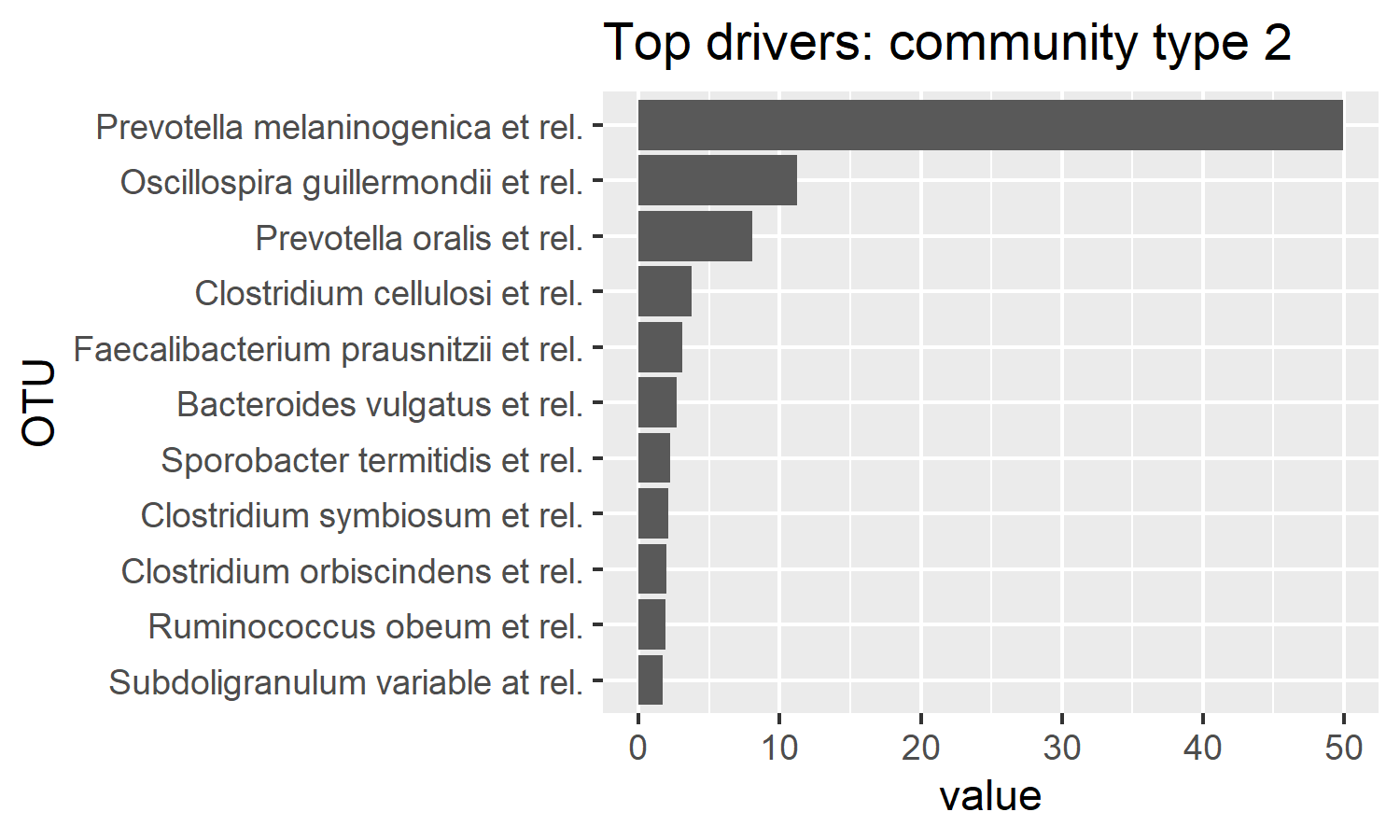
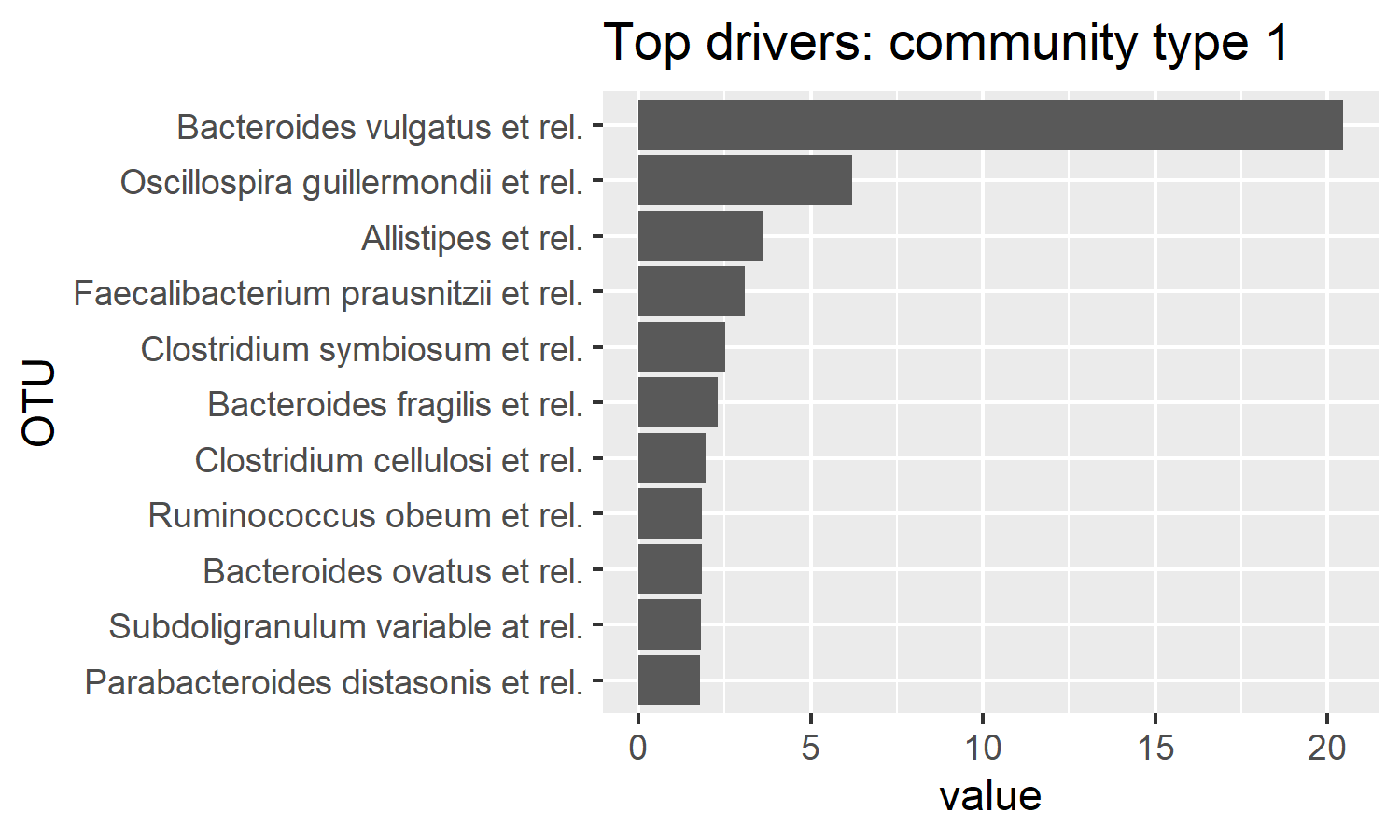
#222个样本中核心物种的丰富度，行为种名，列为样本对应的丰富度  
dat<-abundances(pseq)  
#将核心群的丰度数据转置且变为矩阵，行为样本数，列为不同的种名  
count<-as.matrix(t(dat))  
#lapply函数调用函数dmn，对样本计数矩阵拟合多项式模型  
fit<-lapply(1:4, dmn, count=count, verbose=TRUE)

Soft kmeans  
 Expectation Maximization setup  
 Expectation Maximization  
 Hessian  
 Soft kmeans  
 iteration 10 change 0.000023  
 Expectation Maximization setup  
 Expectation Maximization  
 Hessian  
 Soft kmeans  
 iteration 10 change 0.000214  
 Expectation Maximization setup  
 Expectation Maximization  
 iteration 10 change 0.000067  
 Hessian  
 Soft kmeans  
 iteration 10 change 0.007448  
 iteration 20 change 0.006604  
 iteration 30 change 0.006360  
 iteration 40 change 0.000126  
 iteration 50 change 0.000002  
 Expectation Maximization setup  
 Expectation Maximization  
 iteration 10 change 0.414550  
 iteration 20 change 0.014948  
 Hessian

#群落类型数设为3，得到三个dmn  
#判断拟合结果  
#laplace获取拟合的模型的参数  
lplc<-sapply(fit, laplace)  
aic<-sapply(fit, AIC)  
#用最大似然法估计参数时，AIC、BIC越小拟合越好  
bic<-sapply(fit,BIC)  
#选择最佳模型  
#unlist对所有参数列表的模型挑选参数最小的  
best<-fit[[which.min(unlist(lplc))]]  
#获取最佳拟合模型的参数pi和theta值  
mixturewt(best)

pi theta  
1 0.3243262 80.23526  
2 0.2647695 119.45009  
3 0.2338796 71.38629  
4 0.1770247 337.95754

#保存最佳模型  
ass<-apply(mixture(best),1,which.max)  
write.csv(ass, file="results/DMM\_3clusters\_L6.csv")  
  
for(k in seq(ncol(fitted(best)))){  
 #melt是数据整合函数，将宽格式转化为长格式  
 d<-melt(fitted(best))  
 #数据重塑后列名设置  
 colnames(d)<-c("OTU","cluster","value")  
 d<-subset(d, cluster==k) %>%  
 #按照assignment strength升序排列  
 arrange(value) %>%  
 #mutate函数在数据框中添加新变量，unique函数去除重复值提取唯一值，并将对应OTU转换为因子，只展示最重要的  
 mutate(OTU = factor(OTU, levels = unique(OTU))) %>%  
 #过滤OTUvalue值大于四分位数的  
 filter(abs(value)>quantile(abs(value), 0.8))  
 p<-ggplot(d, aes(x=OTU, y= value))+  
 geom\_bar(stat = "identity")+  
 #坐标轴转换  
 coord\_flip()+  
 labs(title = paste("Top drivers: community type", k))  
 print(p)  
}



#如何先择最佳的k值  
data(fit)  
lplc <- sapply(fit, laplace)  
#plot(lplc, type="b")  
fit[[which.min(lplc)]]

class: DMN   
k: 4   
samples x taxa: 278 x 130   
Laplace: 38781.1 BIC: 40425.31 AIC: 39476.69

lplc2 <- as.data.frame(lplc)  
lplc2$cluster <- rownames(lplc2)  
p01\_DMM <- ggplot(data = lplc2, aes(x=cluster,y=lplc, group = 1))+  
 geom\_point()+  
 geom\_line(color = "lightblue")+  
 xlab("Number of clusters k")+   
 ylab("Laplace approximation")+   
 theme\_classic() + #去掉背景灰色  
 geom\_vline(xintercept = 4, colour='black', lwd=0.36, linetype="dashed")+  
 theme(panel.grid.major=element\_line(colour=NA),  
 panel.background = element\_rect(fill = "transparent",colour = NA),  
 plot.background = element\_rect(fill = "transparent",colour = NA),  
 plot.title = element\_text(hjust = 0.5,size = 15),  
 panel.grid.minor = element\_blank(),   
 text = element\_text(family = "sans"), #设置中文字体的显示  
 axis.text.x = element\_text(hjust = 0.5,size = 10),   
 axis.text.y = element\_text(hjust = 0.5,size = 10),  
 axis.title.y = element\_text(size = 15),   
 axis.title.x = element\_text(size = 15),   
 legend.text = element\_text(size = 15),  
 legend.position = c(.92,.72), #更改图例的位置，放至图内部的右上角  
 legend.box.background = element\_rect(color="black"))#+ #图例添加边框线  
 #scale\_x\_continuous(limits = c(2000,2014),breaks = seq(2000,2014,1)) #更改横坐标刻度值  
(p01\_DMM)

