DirichletMultinomial for Clustering and Classification of Microbiome Data

Martin Morgan

Modified: 6 March 2012. Compiled: May 2, 2019

This document illustrates the main features of the DirichletMultinomial package, and in the process replicates key tables and figures from [1].

We start by loading the package, in addition to the packages lattice (for visualization) and parallel (for use of multiple cores during cross-validation).

> library(DirichletMultinomial)
> library(lattice)
> library(xtable)
> library(parallel)

We set the width of R output to 70 characters, and the number of floating point digits displayed to two. The full flag is set to FALSE, so that cached values are used instead of re-computing during production of this vignette. The package defines a set of standard colors; we use .qualitative during visualization. dev.off is redefined to return without displaying results.

> options(width=70, digits=2)
> full <- FALSE
> .qualitative <- DirichletMultinomial:::qualitative
> dev.off <- function(...) invisible(grDevices::dev.off(...))

1 Data

The data used in [1] is included in the package. We read the data in to a matrix count of samples × taxa.

> fl <- system.file(package="DirichletMultinomial", "extdata", + "Twins.csv")
> count <- t(as.matrix(read.csv(fl, row.names=1)))
> count[1:5, 1:3]

<table>
<thead>
<tr>
<th>Acetanaerobacterium</th>
<th>Acetivibrio</th>
<th>Acetobacterium</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS1.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Figure 1: Density of taxa, across samples

```r
> cnts <- log10(colSums(count))
> pdf("taxon-counts.pdf")
> densityplot(cnts, xlim=range(cnts),
+ xlab="Taxon representation (log 10 count)")
> dev.off()
```

2 Clustering

The `dmn` function fits a Dirichlet-Multinomial model, taking as input the count data and a parameter $k$ representing the number of Dirichlet components to model. Here we fit the count data to values of $k$ from 1 to 7, displaying the result for $k = 4$. A sense of the model return value is provided by the documentation for the R object `fit`, class `DMN`.

```r
> if (full) {
+   fit <- mclapply(1:7, dmn, count=count, verbose=TRUE)
```
The return value can be queried for measures of fit (Laplace, AIC, BIC); these are plotted for different \( k \) in Figure 2. The best fit is for \( k = 4 \) distinct Dirichlet components.

```r
> lplc <- sapply(fit, laplace)
> pdf("min-laplace.pdf")
> plot(lplc, type="b", xlab="Number of Dirichlet Components",
+ ylab="Model Fit")
> dev.off()
> (best <- fit[[which.min(lplc)]])
```

In addition to `laplace` goodness of fit can be assessed with the `AIC` and `BIC` functions.

The `mixturewt` function reports the weight \( \pi \) and homogeneity \( \theta \) (large values are more homogeneous) of the fitted model. `mixture` returns a matrix of sample x estimated Dirichlet components; the argument `assign` returns a vector of length equal to the number of samples indicating the component with maximum value.

```r
> mixturewt(best)

pi theta
1  0.31  52
2  0.17  19
3  0.30  53
4  0.22  30
```

```r
> head(mixture(best), 3)

TS1.2 1.0e+00 2.1e-11 8.6e-06 3.3e-08
TS10.2 3.8e-08 3.3e-04 1.0e+00 2.8e-10
TS100.2 7.2e-09 8.8e-01 8.0e-13 1.2e-01
```
The fitted function describes the contribution of each taxonomic group (each point in the panels of Figure 3) to the Dirichlet components; the diagonal nature of the points in a panel suggest that the Dirichlet components are correlated, perhaps reflecting overall numerical abundance.

```r
> pdf("fitted.pdf")
> splom(log(fitted(best)))
> dev.off()
```

The posterior mean difference between the best and single-component Dirichlet multinomial model measures how each component differs from the population average; the sum is a measure of total difference from the mean.

```r
> p0 <- fitted(fit[[1]], scale=TRUE) # scale by theta
> p4 <- fitted(best, scale=TRUE)
> colnames(p4) <- paste("m", 1:4, sep="")
> (meandiff <- colSums(abs(p4 - as.vector(p0))))

m1  m2  m3  m4
0.26 0.47 0.51 0.34

> sum(meandiff)

[1] 1.6
```

Table 1 summarizes taxonomic contributions to each Dirichlet component.
Figure 3: Taxa fitted to Dirichlet components 1-4.

```r
> diff <- rowSums(abs(p4 - as.vector(p0)))
> o <- order(diff, decreasing=TRUE)
> cdiff <- cumsum(diff[o]) / sum(diff)
> df <- head(cbind(Mean=p0[o], p4[o,], diff=diff[o], cdiff), 10)
```

Table 1: Taxonomic contributions (10 largest) to Dirichlet components.

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>m1</th>
<th>m2</th>
<th>m3</th>
<th>m4</th>
<th>diff</th>
<th>cdiff</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bacteroides</td>
<td>0.17</td>
<td>0.23</td>
<td>0.08</td>
<td>0.39</td>
<td>0.07</td>
<td>0.46</td>
<td>0.29</td>
</tr>
<tr>
<td>Uknown</td>
<td>0.31</td>
<td>0.34</td>
<td>0.45</td>
<td>0.22</td>
<td>0.29</td>
<td>0.27</td>
<td>0.46</td>
</tr>
<tr>
<td>Faecalibacterium</td>
<td>0.10</td>
<td>0.09</td>
<td>0.04</td>
<td>0.14</td>
<td>0.14</td>
<td>0.15</td>
<td>0.56</td>
</tr>
<tr>
<td>Prevotella</td>
<td>0.01</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.05</td>
<td>0.06</td>
<td>0.59</td>
</tr>
<tr>
<td>Alistipes</td>
<td>0.02</td>
<td>0.04</td>
<td>0.01</td>
<td>0.02</td>
<td>0.02</td>
<td>0.04</td>
<td>0.62</td>
</tr>
<tr>
<td>Dorea</td>
<td>0.03</td>
<td>0.01</td>
<td>0.04</td>
<td>0.02</td>
<td>0.03</td>
<td>0.04</td>
<td>0.65</td>
</tr>
<tr>
<td>Ruminococcus</td>
<td>0.02</td>
<td>0.04</td>
<td>0.01</td>
<td>0.01</td>
<td>0.02</td>
<td>0.04</td>
<td>0.67</td>
</tr>
<tr>
<td>Oscillibacter</td>
<td>0.03</td>
<td>0.04</td>
<td>0.01</td>
<td>0.02</td>
<td>0.03</td>
<td>0.04</td>
<td>0.70</td>
</tr>
<tr>
<td>Roseburia</td>
<td>0.04</td>
<td>0.02</td>
<td>0.05</td>
<td>0.04</td>
<td>0.04</td>
<td>0.04</td>
<td>0.72</td>
</tr>
<tr>
<td>Subdoligranulum</td>
<td>0.03</td>
<td>0.03</td>
<td>0.02</td>
<td>0.02</td>
<td>0.03</td>
<td>0.03</td>
<td>0.74</td>
</tr>
</tbody>
</table>

Figure 3 shows samples arranged by Dirichlet component, with samples placed into the component for which they had the largest fitted value.

```r
> pdf("heatmap1.pdf")
> heatmapdmn(count, fit[[1]], best, 30)
```
Figure 4: Samples arranged by Dirichlet component. Narrow columns are samples, broader columns component averages. Rows are taxonomic groups. Color represents square-root counts, with dark colors corresponding to larger counts.

```r
> dev.off()
```

### 3 Generative classifier

The following reads in phenotypic information (‘Lean’, ‘Obese’, ‘Overweight’) for each sample.

```r
> fl <- system.file(package="DirichletMultinomial", "extdata", + "TwinStudy.t")
> pheno0 <- scan(fl)
> lvls <- c("Lean", "Obese", "Overwt")
> pheno <- factor(lvls[pheno0 + 1], levels=lvls)
> names(pheno) <- rownames(count)
> table(pheno)
```

```
pheno
   Lean Obese Overwt
  61   193    24
```

Here we subset the count data into sub-counts, one for each phenotype. We retain only the Lean and Obese groups for subsequent analysis.
counts <- lapply(levels(pheno), csubset, count, pheno)
sapply(counts, dim)

[,1] [,2] [,3]
[1,] 61 193 24
[2,] 130 130 130

keep <- c("Lean", "Obese")
count <- count[pheno %in% keep,]
pheno <- factor(pheno[pheno %in% keep], levels=keep)

The `dmngroup` function identifies the best (minimum Laplace score) Dirichlet-multinomial model for each group.

if (full) {
  bestgrp <- dmngroup(count, pheno, k=1:5, verbose=TRUE,
                      mc.preschedule=FALSE)
  save(bestgrp, file=file.path(tempdir(), "bestgrp.rda"))
} else data(bestgrp)

The Lean group is described by a model with one component, the Obese group by a model with three components. Three of the four Dirichlet components of the original single group (`best`) model are represented in the Obese group, the other in the Lean group. The total Laplace score of the two group model is less than of the single-group model, indicating information gain from considering groups separately.

> bestgrp
class: DMNGroup
summary:
  k samples taxa NLE LogDet Laplace BIC AIC
Lean 1 61 130 9066 162 9027 9333 9196
Obese 3 193 130 26770 407 26613 27801 27162

> lapply(bestgrp, mixturewt)

$Lean
  pi theta
1 1 35

$Obese
  pi theta
1 0.53 45
2 0.26 33
3 0.22 18

> c(sapply(bestgrp, laplace),
   `Lean+Obese`=sum(sapply(bestgrp, laplace)),
   `Single`=laplace(best))
The `predict` function assigns samples to classes; the confusion matrix shows that the classifier is moderately effective.

```r
> xtabs(~pheno + predict(bestgrp, count, assign=TRUE))

predict(bestgrp, count, assign = TRUE)

<table>
<thead>
<tr>
<th>pheno</th>
<th>Lean</th>
<th>Obese</th>
<th>Lean+Obese</th>
<th>Single</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lean</td>
<td>38</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Obese</td>
<td>15</td>
<td>178</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

The `cvdmngroup` function performs cross-validation. This is a computationally expensive step.

```r
> if (full) {
+    ## full leave-one-out; expensive!
+    xval <- cvdmngroup(nrow(count), count, c(Lean=1, Obese=3), pheno,
+                        verbose=TRUE, mc.preschedule=FALSE)
+    save(xval, file=file.path(tempdir(), "xval.rda"))
+ } else data(xval)
```

Figure 5 shows an ROC curve for the single and two-group classifier. The single group classifier is performing better than the two-group classifier.

```r
> bst <- roc(pheno[rownames(count)] == "Obese",
+            predict(bestgrp, count)[,"Obese"])
> bst$Label <- "Single"
> two <- roc(pheno[rownames(xval)] == "Obese",
+            xval[,"Obese"])
> two$Label <- "Two group"
> both <- rbind(bst, two)
> pars <- list(superpose.line=list(col=.qualitative[1:2], lwd=2))
> pdf("roc.pdf")
> xyplot(TruePostive ~ FalsePositive, group=Label, both,
+        type="l", par.settings=pars,
+        auto.key=list(lines=TRUE, points=FALSE, x=.6, y=.1),
+        xlab="False Positive", ylab="True Positive")
> dev.off()
```

```r
toLatex(sessionInfo())
```

- R version 3.6.0 (2019-04-26), x86_64-pc-linux-gnu
- Locale: LC_CTYPE=en_US.UTF-8, LC_NUMERIC=C, LC_TIME=en_US.UTF-8, LC_COLLATE=C, LC_MONETARY=en_US.UTF-8, LC_MESSAGES=en_US.UTF-8, LC_PAPER=en_US.UTF-8, LC_NAME=C, LC_ADDRESS=C, LC_TELEPHONE=C, LC_MEASUREMENT=en_US.UTF-8, LC_IDENTIFICATION=C
Figure 5: Receiver-operator curves for the single and two-group classifiers.

- Running under: **Ubuntu 18.04.2 LTS**
- Matrix products: default
- BLAS: `/home/biocbuild/bbs-3.10-bioc/R/lib/libRblas.so`
- LAPACK: `/home/biocbuild/bbs-3.10-bioc/R/lib/libRlapack.so`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: BiocGenerics 0.31.0, DirichletMultinomial 1.27.0, IRanges 2.19.0, S4Vectors 0.23.0, lattice 0.20-38, xtable 1.8-4
- Loaded via a namespace (and not attached): compiler 3.6.0, grid 3.6.0, tools 3.6.0

**References**