HPO富集分析

参考： R package DOSE (DOSE:::enricher_internal)

1 原始HPO关联数据的整理

```
#Format: diseaseId<tab>gene-symbol<tab>gene-id(entrez)<tab>HPO-ID<tab>HPO-term-name
OMIM:600920     SCARF2  91179   HP:0000767      Pectus excavatum
OMIM:600920     SCARF2  91179   HP:0003042      Elbow dislocation
OMIM:600920     SCARF2  91179   HP:0005280      Depressed nasal bridge
```

```
for i in {1..3}; do
  grep -v '^#' HPO_diseases_to_genes_to_phenotypes.txt | cut -f$i | uniq -c | gawk '{print $1}' >_${i}
done
cmp _1 _2 # diff
cmp _2 _3 # same
```

```
0@DESKTOP-RDP4CUK:/mnt/c/Users/0/Desktop/BioNLP/2019-03-25/data
$ grep -v '^#' HPO_diseases_to_genes_to_phenotypes.txt | wc -l
78875
0@DESKTOP-RDP4CUK:/mnt/c/Users/0/Desktop/BioNLP/2019-03-25/data
$ grep -v '^#' HPO_diseases_to_genes_to_phenotypes.txt | cut -f3,4 | sort -u | wc -l
70699
0@DESKTOP-RDP4CUK:/mnt/c/Users/0/Desktop/BioNLP/2019-03-25/data
$ grep -v '^#' HPO_diseases_to_genes_to_phenotypes.txt | cut -f2- | sort -u | wc -l
70699
0@DESKTOP-RDP4CUK:/mnt/c/Users/0/Desktop/BioNLP/2019-03-25/data
$ (printf "%s\t%s\t%s\t%s\n" 'gene-symbol' 'gene-id(entrez)' 'HPO-ID' 'HPO-term-name';
> grep -v '^#' HPO_diseases_to_genes_to_phenotypes.txt | cut -f2- | sort -u) > UNIQ_HPO_GP.txt
```

2 HPO关联数据读取

```
setwd('/Users/O/Desktop/BioNLP/2019-03-25/data')
hpo_data <- read.delim('UNIQ_HPO_GP.txt', sep = '\t', header = TRUE, stringsAsFactors = FALSE)
colnames(hpo_data) <- c("gene-symbol", "gene-id(entrez)", "HPO-ID", "HPO-term-name")
hpo_data[["gene-id(entrez)"]] <- as.character(hpo_data[["gene-id(entrez)"]])
save(hpo_data, file = "hpo_data.rda")
GENE  <- hpo_data[["gene-id(entrez)"]]
HPOID <- hpo_data[["HPO-ID"]]
G2P <- sapply(unique(GENE),  function(x) HPOID[GENE == x])
P2G <- sapply(unique(HPOID), function(x) GENE[HPOID == x])
save(G2P, file = 'G2P.rda')
save(P2G, file = 'P2G.rda')
```

3 富集分析的实现

3.1 P value计算

基因数目	全部基因（GENE）	待查询的基因（gene）
in hpoid	M	Q
not in hpoid	N - M	k - Q
total	N	k

$$P = 1 - \sum_{i=0}^{Q-1} \frac{choose(M, i) * choose(N - M, k - i)}{choose(N, k)}$$

```
GetPValue <- function(Q, M, N, k)
    1 - sum(sapply(0:(Q - 1), function(i) choose(M, i) * choose(N - M, k - i) / choose(N, k)))
```

参考：http://www.bioconductor.org/packages/release/bioc/vignettes/DOSE/inst/doc/enrichmentAnalysis.html

3.2 富集分析函数

```r
BasicEnrichHPO <- function(gene, G2P, P2G, HPOID2NAME, minCount = 0, maxCount = Inf) {
    gene <- intersect(gene, names(G2P))
    qG2P <- G2P[gene]
    .qG <- rep(gene, times = sapply(qG2P, length))
    .qP <- unlist(qG2P)
    hpoid <- unique(.qP)
    qP2G <- sapply(hpoid, function(x) .qG[.qP == x])
    gP2G <- P2G[hpoid]
    vQ <- sapply(qP2G, length)
    vM <- sapply(gP2G, length)
    #### 过滤步骤示例（对某些术语进行过滤） ####
    valid_index <- (vM >= minCount & vM <= maxCount)
    vQ <- vQ[valid_index]
    vM <- vM[valid_index]
    N <- length(G2P)
    k <- length(gene)
    pvalues <- sapply(1:length(hpoid), function(i) GetPValue(vQ[i], vM[i], N, k))
    res <- data.frame(
        "P_ID" = hpoid,
        "P_DOC" = HPOID2NAME[hpoid],
        "GeneRatio" = paste(vQ, k, sep = '/'),
        "BgRatio" = paste(vM, N, sep = '/'),
        "pvalues" = pvalues,
        "G_ID" = sapply(qP2G, function(s) do.call("paste", as.list(c(s, sep = '')))),
        "Count" = vQ,
        stringsAsFactors = FALSE
    )
    res[order(res$pvalues), ]
}
```

3.3 数据分析过程

```
load('hpo_data.rda')
load('G2P.rda')
load('P2G.rda')
load('/_winsoft/R/R-3.5.1/library/DOSE/data/geneList.rda')
gene <- intersect(names(geneList)[abs(geneList) > 1.5], names(G2P))

HPOID2NAME <- unique(hpo_data[["HPO-term-name"]])
names(HPOID2NAME) <- unique(hpo_data[["HPO-ID"]])

res <- BasicEnrichHPO(gene, G2P, P2G, HPOID2NAME)
```

数据来源：DOSE包，geneList.rda