XDAS2020 - Hypergeometric test

Kris Gunsalus

10/29/2020

Testing for gene set overlap

A recent study[1] created organoid models of brain tumor formation using CRISPR to generate mutations in known oncogenes. Differences in gene expression between two tumor models (primitive neuroectodermal tumors and glioblastoma were compared with controls and were found to differ in the sets of genes that were mis-regulated.

The two conditions were generated by MYC oncogene overexpression (MYC^{OE}) (CNS-NET) and knockouts of several genes including $PTEN$ and $P53$ (GBM).

The researchers first showed that the two tumor models seem to show quite different gene expression patterns using **PCA** (principal components analysis) of the top 500 genes with the highest variability compared with controls (Figure 3a,b). We will go over PCA later in the course.

Next, they performed **hypergeometric tests** to ask whether the gene sets that were significantly **up- or down-regulated** in the two models (in comparison with controls) showed less overlap that expected by random chance (Supplementary Figure 3).

If so, this would indicate that the gene programs that are misregulated in the two models are significantly divergent.

Hypergeometric test in R

First, look at the documentation for the hypergeometric distribution in R.

```
# check how the hypergeometric functions work
help(phyper)
```

Up-regulated genes in brain organoids

You want to test whether the two sets overlap by less than expected by chance. How will you set up the problem? The numbers you need are given in **Fig. S3**.

```
## Up-regulation ========================================================= #
## 2450 total genes up-regulated in Clusters 2 & Clusters 3
## out of 4035 genes up-regulated in either experiment (vs CTL)
# C2 = 2155 + 92 = 2247
# C3 = 203 + 92 = 295
```

[1] **Ref**: Genetically engineered cerebral organoids model brain tumor formation. Bian..Knoblich, *Nature Methods* 15,631–639 (2018); **URL**: https://www.nature.com/articles/s41592-018-0070-7 ; **Fig S3**: https://www.nature.com/articles/s41592-018-0070-7/figures/9

```r
# Overlap = 92

t = 92         # overlap
A = 2155 + t   # 2247: size of Cluster 2 (Set A = bigger set)
B = 203 + t    # 295: size of Cluster 3 (Set B = smaller set)
N = 4035       # total genes up-regulated in both experiments

# What is the expected overlap?
ol.exp = A*B / N
ol.exp
## [1] 164.2788
```

Hypergeometric test

Perform a hypergeometric test with the above data. Remember, you are looking for less overlap than expected between the two sets.

```r
## p-value using hypergeometric test ----------------------- #

# if B <= A (smaller than or equal to), the maximum possible size of t is B;
# so the total probability of getting an overlap of size t or LESS is:

# CDF: P(x <= t)
# parameterization for phyper(q, m, n, k, lower.tail = T/F)
#   x = number of "white balls" drawn (t)
#   m = number of "white balls" in urn (let's use A=C2)
#   n = number of "black balls" in urn (everything that is NOT in A)
#   k = number of balls drawn from the urn (let's use B=C3 as the frame of reference)
phyper(t, A, N-A, B, lower.tail=T) # p-value for t
## [1] 1.047283e-18

# PDF: P(x <= t) = sum(P(x = 0:t))
sum(dhyper(0:t, A, N-A, B)) # p-value for t (same as using CDF)
## [1] 1.047283e-18

# the P-value is actually the same either way
phyper(t, B, N-B, A, lower.tail=T)  # p-value for t using CDF
## [1] 1.047283e-18
```

Fisher's Exact Test

The hypergeometric test is equivalant to a one-tailed Fisher's test (but it is actually more efficient in R). Use a Fisher's test to make the same comparison.

```r
## with Fisher's exact test ------------------------------ #

# make a contingency table: the orientation is arbitrary,
# since the test is for more (or less) extreme values
#   - by convention, put the categories "of interest" first

# set up a matrix of values in this order:
#   matrix(c(intersect(A,B), setdiff(B,A),
```

```
#            setdiff(A,B), n - union(A,B) ), nrow=2, byrow = T)

test.data = matrix(c( t,    B-t,
                      A-t,  N-(A+B-t) ), nrow=2, byrow = T)
rownames(test.data) = c("C3","not.C3")
colnames(test.data) = c("C2","not.C2")
test.data
##          C2 not.C2
## C3       92    203
## not.C3 2155   1585

# now do the test

#fisher.test(test.data, alternative="greater") # test for enrichment
fisher.test(test.data, alternative="less")     # test for depletion
##
##  Fisher's Exact Test for Count Data
##
## data:  test.data
## p-value < 2.2e-16
## alternative hypothesis: true odds ratio is less than 1
## 95 percent confidence interval:
##  0.0000000 0.4155848
## sample estimates:
## odds ratio
##  0.3334194
```

Down-regulated genes in brain organoids

Check that you can also get the same p-value as shown in the paper for the down-regulated genes. You can just copy and paste the commands from above, except using the new values from **Fig. S3** for the down-regulated genes.

```
## Down-regulation ========================================================= #
# 1493 genes in B and C out of 2738 total

t = 97       # overlap
A = 995 + t  # size of size of Cluster 2 (Set A = bigger set)
B = 401 + t  # size of size of Cluster 3 (Set B = smaller set)
N = 2738     # total

# What is the expected overlap?
ol.exp = A*B / N
ol.exp
## [1] 198.618


## p-value using hypergeometric test ----------------------- #
phyper(t, A, N-A, B, lower.tail=T)
## [1] 1.033302e-26


## with Fisher's exact test ------------------------------ #
```

```
test.data = matrix(c( t,   B-t,
                      A-t, N-(A+B-t) ), nrow=2, byrow = T)
rownames(test.data) = c("C3","not.C3")
colnames(test.data) = c("C2","not.C2")
test.data
##         C2 not.C2
## C3      97    401
## not.C3 995   1245

fisher.test(test.data, alternative="less")     # test for depletion
##
##  Fisher's Exact Test for Count Data
##
## data:  test.data
## p-value < 2.2e-16
## alternative hypothesis: true odds ratio is less than 1
## 95 percent confidence interval:
##  0.0000000 0.3712296
## sample estimates:
## odds ratio
##  0.3027927
```