

R function scLR

C.-Y. Hsu

Sept 03, 2021

Usage of scLR

A simulated data to compare the products of the expressions of LR pairs between two conditions: TX1 and TX2. 3 samples for each condition. For each sample, there are 1000 genes and 100 cells (5 cell types, 20 cells each). 10 ligand-receptor gene pairs across 5 cell types are compared between TX1 and TX2.

Three inputs are required: countmatrix (gene names required), cellinfo, lrpairs.sample

```
library(scLR)
set.seed(2021)
G <- 1000; n <- 600 # To create a simulated data consisting of 1000 genes and 600 cells
# Data are generated from NB distribution
NB_cell <- function(j) rnbinom(G, size = 0.1, mu = rgamma(G, shape = 2, rate = 2))
countmatrix <- as(sapply(1:n, NB_cell), "sparseMatrix")
# 1000 gene names are taken from lrpairs0 [LR pairs which are commonly compared (built in scLR)]
genenames <- unique(unlist(lrpairs0))
rownames(countmatrix) <- genenames[sample(1:length(genenames),1000)]

# Information for all cells
cellinfo <- data.frame(sampleID = factor(paste0("s", rep(1:6, each=100))),
                      condition = factor(paste0("tx", rep(1:2, each=300))),
                      cellcluster = factor(paste0("cc", rep(rep(1:5, each=20), 6))))

# Names of 10 ligand-receptor pairs which will be compared
lrpairs.sample <- data.frame(lrpairs0[sample(1:200, 10),])
```

Formats of 3 inputs:

```
countmatrix[1:6, 1:10] # format of countmatrix, gene expressions (the first 6 genes and 10 cells)
```

```
## 6 x 10 sparse Matrix of class "dgCMatrix"
##
## TMEM222 15 . 17 . . . 6 27 . .
## RLN2    . . . . . 4 . 89 . .
## ITGB8    1 . . . . . . . . .
## CD96     . . . . 6 . 2 . . .
## FGF3     1 . 6 . . . . . 2 .
## GPC5     . . . . . 4 . . . .
```

```
head(cellinfo, 10) # format of cellinfo, categories of cells (the first 10 cells)
```

```
##   sampleID condition cellcluster
## 1       s1       tx1         cc1
## 2       s1       tx1         cc1
## 3       s1       tx1         cc1
```

```
## 4      s1      tx1      cc1
## 5      s1      tx1      cc1
## 6      s1      tx1      cc1
## 7      s1      tx1      cc1
## 8      s1      tx1      cc1
## 9      s1      tx1      cc1
## 10     s1      tx1      cc1
```

```
head(lrpairs.sample, 10) # format of lrpairs.sample, LR gene pairs to compare.
```

```
##      ligand receptor
## 54     ADM2  CALCRL
## 140    APOB  CALCR
## 157    APOC2  LRP2
## 86     AGRP  SDC3
## 81     AGRN  LRP4
## 41    ADCYAP1 PTH1R
## 46    ADCYAP1 VIPR1
## 56     ADM2  GPR84
## 35     ADAM9  ITGB5
## 130    APOA1  LDLR
```

Output:

```
output <- scLR(countmatrix, cellinfo, lrpairs.sample, low.filter = 1,
               parallel.use = FALSE) # Do parallel computation if parallel.use = TRUE.
```

```
## converting counts to integer mode
## converting counts to integer mode
## converting counts to integer mode
## converting counts to integer mode
## converting counts to integer mode
```

```
head(output$Rs, 10)
```

```
##      lr.cell.name lr.gene.name obs.xy.diff null.diff.sd      pvalue stage
## 1      cc1-cc1 ADCYAP1-PTH1R      4.6792      5.9035 0.42000000      1
## 2      cc1-cc1 ADCYAP1-VIPR1      7.7303      5.8353 0.19500000      1
## 3      cc1-cc1  APOA1-LDLR      1.1413      5.9499 0.82000000      1
## 4      cc1-cc2 ADCYAP1-PTH1R      3.9767      5.6762 0.49500000      1
## 5      cc1-cc2 ADCYAP1-VIPR1      7.4522      5.4610 0.18000000      1
## 6      cc1-cc2  APOA1-LDLR     -1.4156      6.0763 0.85000000      1
## 7      cc1-cc3 ADCYAP1-PTH1R     11.7795      5.4812 0.09177253      2
## 8      cc1-cc3 ADCYAP1-VIPR1      6.2115      5.2035 0.28000000      1
## 9      cc1-cc3  APOA1-LDLR      0.1014      5.9871 0.99000000      1
## 10     cc1-cc4 ADCYAP1-PTH1R      2.0802      5.7900 0.69500000      1
##      adj.p Welch.t.stat Welch.t.sd Welch.t.p      adj.t.p limma.Lg.p limma.Rg.p
## 1 0.8250000      3.2131      1.4563 0.041822 0.5185500 0.1645549 0.88674189
## 2 0.6964286      1.6432      4.7044 0.191738 0.7989083 0.1645549 0.53822020
## 3 0.9739583      0.1955      5.8389 0.859995 0.9846586 0.6960390 0.39640971
## 4 0.8250000      2.1816      1.8229 0.100706 0.5809962 0.1645549 0.53140992
## 5 0.6750000      4.4792      1.6637 0.026672 0.5185500 0.1645549 0.57567834
## 6 0.9739583     -0.2992      4.7320 0.787892 0.9687197 0.6960390 0.76381357
## 7 0.6315789      5.9124      1.9923 0.007903 0.4616625 0.1645549 0.06469086
## 8 0.8076923      2.4446      2.5409 0.075114 0.5525932 0.1645549 0.84351846
## 9 0.9900000      0.0212      4.7774 0.984696 0.9924090 0.6960390 0.64505113
## 10 0.8961864      0.5514      3.7727 0.618230 0.8964952 0.1645549 0.31169701
```

##	limma.Lg.logFC	limma.Rg.logFC	adj.limma.Lg.p	adj.limma.Rg.p
## 1	1.4162	-0.1450	0.9574023	0.9923409
## 2	1.4162	0.6313	0.9574023	0.9794468
## 3	-0.4015	0.8711	0.9847046	0.9750298
## 4	1.4162	-0.6166	0.9574023	0.9794468
## 5	1.4162	0.5500	0.9574023	0.9819266
## 6	-0.4015	-0.3118	0.9847046	0.9867267
## 7	1.4162	1.8081	0.9574023	0.9574023
## 8	1.4162	0.1911	0.9574023	0.9867267
## 9	-0.4015	0.4486	0.9847046	0.9823227
## 10	1.4162	-0.9981	0.9574023	0.9574023

Assume sample s2 does NOT have cell type cc3

```
idx.remove <- which(cellinfo$sampleID=="s2" & cellinfo$cellcluster=="cc3")
cellinfo1 <- cellinfo[-idx.remove,]; cellinfo1 <- droplevels(cellinfo1)
countmatrix1 <- countmatrix[, -idx.remove]
```

Output:

```
output <- scLR(countmatrix1, cellinfo1, lrpairs.sample, low.filter = 1, parallel.use = FALSE)
```

```
## converting counts to integer mode
## converting counts to integer mode
## converting counts to integer mode
## converting counts to integer mode
## converting counts to integer mode
```

```
head(output$Rs, 10)
```

	lr.cell.name	lr.gene.name	obs.xy.diff	null.diff.sd	pvalue	stage	adj.p
## 1	cc1-cc1	ADCYAP1-PTH1R	4.6792	5.9035	0.420	1	0.8437500
## 2	cc1-cc1	ADCYAP1-VIPR1	7.7303	5.8353	0.195	1	0.7687500
## 3	cc1-cc1	APOA1-LDLR	1.1413	5.9499	0.820	1	0.9583333
## 4	cc1-cc2	ADCYAP1-PTH1R	3.9767	5.6762	0.495	1	0.8437500
## 5	cc1-cc2	ADCYAP1-VIPR1	7.4522	5.4610	0.180	1	0.7687500
## 6	cc1-cc2	APOA1-LDLR	-1.4156	6.0763	0.850	1	0.9583333
## 7	cc1-cc3	ADCYAP1-PTH1R	11.8642	6.1917	0.060	1	0.6923077
## 8	cc1-cc3	ADCYAP1-VIPR1	5.2517	5.8012	0.375	1	0.8437500
## 9	cc1-cc3	APOA1-LDLR	-0.9382	6.7981	0.840	1	0.9583333
## 10	cc1-cc4	ADCYAP1-PTH1R	2.0802	5.7331	0.740	1	0.9568966
	Welch.t.stat	Welch.t.sd	Welch.t.p	adj.t.p	limma.Lg.p	limma.Rg.p	
## 1	3.2131	1.4563	0.041822	0.5788159	0.1645549	0.88674189	
## 2	1.6432	4.7044	0.191738	0.7568605	0.1645549	0.53822020	
## 3	0.1955	5.8389	0.859995	0.9594812	0.6960390	0.39640971	
## 4	2.1816	1.8229	0.100706	0.5809962	0.1645549	0.53140992	
## 5	4.4792	1.6637	0.026672	0.5788159	0.1645549	0.57567834	
## 6	-0.2992	4.7320	0.787892	0.9594812	0.6960390	0.76381357	
## 7	4.9511	2.3963	0.020676	0.5788159	0.1645549	0.04813945	
## 8	1.9352	2.7138	0.153205	0.6719292	0.1645549	0.90797168	
## 9	-0.1228	7.6432	0.921102	0.9594812	0.6960390	0.98812853	
## 10	0.5514	3.7727	0.618230	0.9140735	0.1645549	0.31169701	
	limma.Lg.logFC	limma.Rg.logFC	adj.limma.Lg.p	adj.limma.Rg.p			
## 1	1.4162	-0.1450	0.9709894	0.9892382			
## 2	1.4162	0.6313	0.9709894	0.9811766			
## 3	-0.4015	0.8711	0.9820304	0.9811123			
## 4	1.4162	-0.6166	0.9709894	0.9811766			
## 5	1.4162	0.5500	0.9709894	0.9820100			
## 6	-0.4015	-0.3118	0.9820304	0.9820304			
## 7	1.4162	2.1427	0.9709894	0.9571765			
## 8	1.4162	0.1231	0.9709894	0.9901449			
## 9	-0.4015	-0.0155	0.9820304	0.9987378			
## 10	1.4162	-0.9981	0.9709894	0.9709894			