

# Pathway analysis and Drug repositioning for Psoriasis based on cogena

Zhilong Jia

2015-09-10

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Downloading the raw data of <a href="#">GSE13355</a></b>	<b>2</b>
<b>3</b>	<b>Differential Expression Analysis</b>	<b>2</b>
<b>4</b>	<b>Co-expression Analysis by cogena</b>	<b>3</b>
<b>5</b>	<b>Pathway Analysis by cogena</b>	<b>4</b>
5.1	The heatmap with co-expression information . . . . .	4
5.2	Table 1: Co-expressed genes are highly connected . . . . .	4
5.3	Figure 2: The result of pathway analysis . . . . .	6
5.4	Table 2 and S1: Make Input for GSEA . . . . .	6
<b>6</b>	<b>Drug repositioning by cogena</b>	<b>8</b>
6.1	Figure 3: Drug repositioning for cluster 3 (A) . . . . .	8
6.2	Figure 4: Drug repositioning for cluster 4 (B) . . . . .	9
6.3	Figure S1: Drug repositioning for cluster 6 (C) . . . . .	10
6.4	Table S2: Output DEGs for CMAP and NFFinder Analysis . . . . .	12
<b>7</b>	<b>System Info</b>	<b>12</b>

## 1 Introduction

*As a reproducible research, this is the whole code and files necessary for the manuscript, **Drug repositioning and drug mode of action discovering based on co-expressed gene-set enrichment analysis.***

packages required:

- knitr
- GEOquery
- affy
- MetaDE
- annotate
- hgu133plus2.db

- ## 2 Downloading the raw data of GSE13355

### 3 Differential Expression Analysis

2

```
#####
# Filter the non-informative and non-expressed genes first.
library(MetaDE)
library(annotate)
library(hgu133plus2.db)

GSE13355.Explist <- list(GSE13355=list(x = exprs(GSE13355rma),
      y = ifelse (GSE13355label$state=="ct", 0, 1),
      symbol = getSYMBOL(rownames(exprs(GSE13355rma)), "hgu133plus2") ))
GSE13355.Explist <- MetaDE.match(GSE13355.Explist, pool.replicate="IQR")
GSE13355.Explist.filtered <- MetaDE.filter(GSE13355.Explist, c(0.2,0.2))
colnames(GSE13355.Explist.filtered$GSE13355$x) <- colnames(exprs(GSE13355rma))

# DEG analysis via limma
DElimma <- function (Expdata, Explabel){

  library(limma)
  Expdesign <- model.matrix(~as.factor(Explabel$rep) + Explabel$state)
  Expfit1 <- lmFit(Expdata, Expdesign)
  Expfit2 <- eBayes(Expfit1)
  dif_Exp <- topTable(Expfit2, coef=tail(colnames(Expdesign), 1), number=Inf)

  return (dif_Exp)
}

GSE13355.limma <- DELimma(GSE13355.Explist.filtered$GSE13355$x, GSE13355label)
GSE13355.DE <- GSE13355.limma[GSE13355.limma$adj.P.Val<=0.05 & abs(GSE13355.limma$logFC)>=1,]
GSE13355.DEG <- rownames(GSE13355.DE)
GSE13355.DEG.expr <- GSE13355.Explist.filtered$GSE13355$x[GSE13355.DEG,]
```

## 4 Co-expression Analysis by cogena

```
# Install cogena if none
# devtools::install_github("zhilongjia/cogena")
library(cogena)
annoGMT <- "c2.cp.kegg.v5.0.symbols.gmt.xz"
annofile <- system.file("extdata", annoGMT, package="cogena")
# nClust <- 2:20
# clMethods <- c("hierarchical", "kmeans", "diana", "fanny", "som", "sota", "pam", "clara", "agnes")
nClust <- 7
clMethods <- c("pam")
ncore <- 7
# Co-expression analysis
genecl_result <- coExp(GSE13355.DEG.expr, nClust=nClust,
      clMethods=clMethods,
      metric="correlation",
      method="complete",
      ncore=ncore,
      verbose=FALSE)
```

## 5 Pathway Analysis by cogena

```
sampleLabel <- GSE13355label$state
names(sampleLabel) <- rownames(GSE13355label)
# cogena analysis (Pathway analysis)
cogena_result <- clEnrich(genecl_result, annofile=annofile, sampleLabel=sampleLabel)

# Summary the results obtained by cogena
summary(cogena_result)
```

```
##
## Clustering Methods:
## pam
##
## The Number of Clusters:
## 7
##
## Metric of Distance Matrix:
## correlation
##
## Agglomeration method for hierarchical clustering (hclust and agnes):
## complete
##
## Gene set:
## c2.cp.kegg.v5.0.symbols.gmt.xz
```

### 5.1 The heatmap with co-expression information

```
# Figure 1
heatmapCluster(cogena_result, "pam", "7", maintitle="Psoriasis")
```

```
## The number of genes in each cluster:
## upDownGene
## 1 2
## 468 238
## cluster_size
## 1 2 3 4 5 6 7
## 257 65 81 130 94 61 18
```

### 5.2 Table 1: Co-expressed genes are highly connected

```
# pPPI function: get the PPI summary information about input genes
pPPI <- function(geneC, string_db){
  example1_mapped <- string_db$map(as.data.frame(geneC), "geneC",
                                   removeUnmappedRows = TRUE, quiet=TRUE)
  hits <- example1_mapped$STRING_id
  net_summary <- string_db$get_summary(unique(hits))
}
```

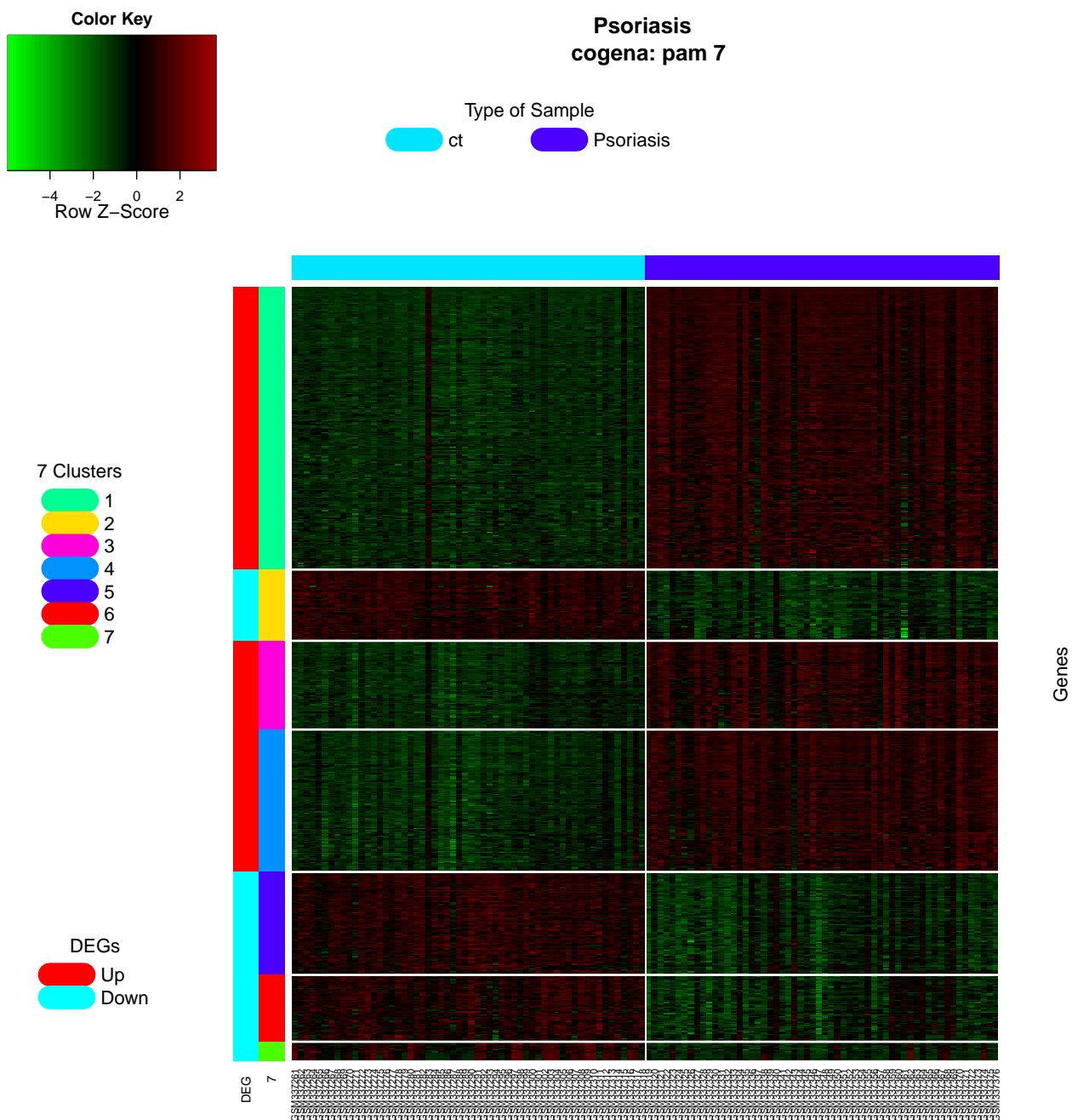


Figure 1: Heatmap with co-expression information

```

    as.numeric( gsub("[^1:9]+\\: |\\)", "", strsplit(net_summary, "\\n|\\(")[[1]] ) )
}

cluster_ppi <- data.frame(protein=numeric(10), interactions=numeric(10),
                          expected_interactions=numeric(10),
                          p_value=numeric(10), stringsAsFactors=FALSE)
rownames(cluster_ppi) <- c(1:7, "Up", "Down", "All_DE")

library(STRINGdb)
suppressWarnings(string_db <- STRINGdb$new(version="10", species=9606,
                                           score_threshold=400,
                                           input_directory=~ /tmp/STRINGdb))

for (i in 1:7) {
  i <- as.character(i)
  cluster_ppi[i,] <- pPPI(geneInCluster(cogena_result, "pam", "7", i), string_db)
}
cluster_ppi["Up",] <- pPPI(rownames(GSE13355.DE[GSE13355.DE$logFC>0,]), string_db)
cluster_ppi["Down",] <- pPPI(rownames(GSE13355.DE[GSE13355.DE$logFC<0,]), string_db)
cluster_ppi["All_DE",] <- pPPI(rownames(GSE13355.DE), string_db)
# Table 1
knitr::kable(cluster_ppi, caption="Summary of interactions within clusters")

```

Table 1: Summary of interactions within clusters

	protein	interactions	expected_interactions	p_value
1	247	302	116	0.0000000
2	62	15	7	0.0078911
3	80	287	24	0.0000000
4	126	500	92	0.0000000
5	90	19	11	0.0348286
6	61	59	16	0.0000000
7	18	3	0	0.0048405
Up	453	1616	633	0.0000000
Down	231	235	112	0.0000000
All_DE	684	2407	1274	0.0000000

### 5.3 Figure 2: The result of pathway analysis

```

# Figure 2
heatmapPEI(cogena_result, "pam", "7", printGS=FALSE, maintitle="Psoriasis")

```

### 5.4 Table 2 and S1: Make Input for GSEA

This is to get the *Table 2 and S1*. The result can be obtained from `../result/GSEA_output`, too. See `gct` and `cls` file format if needed.

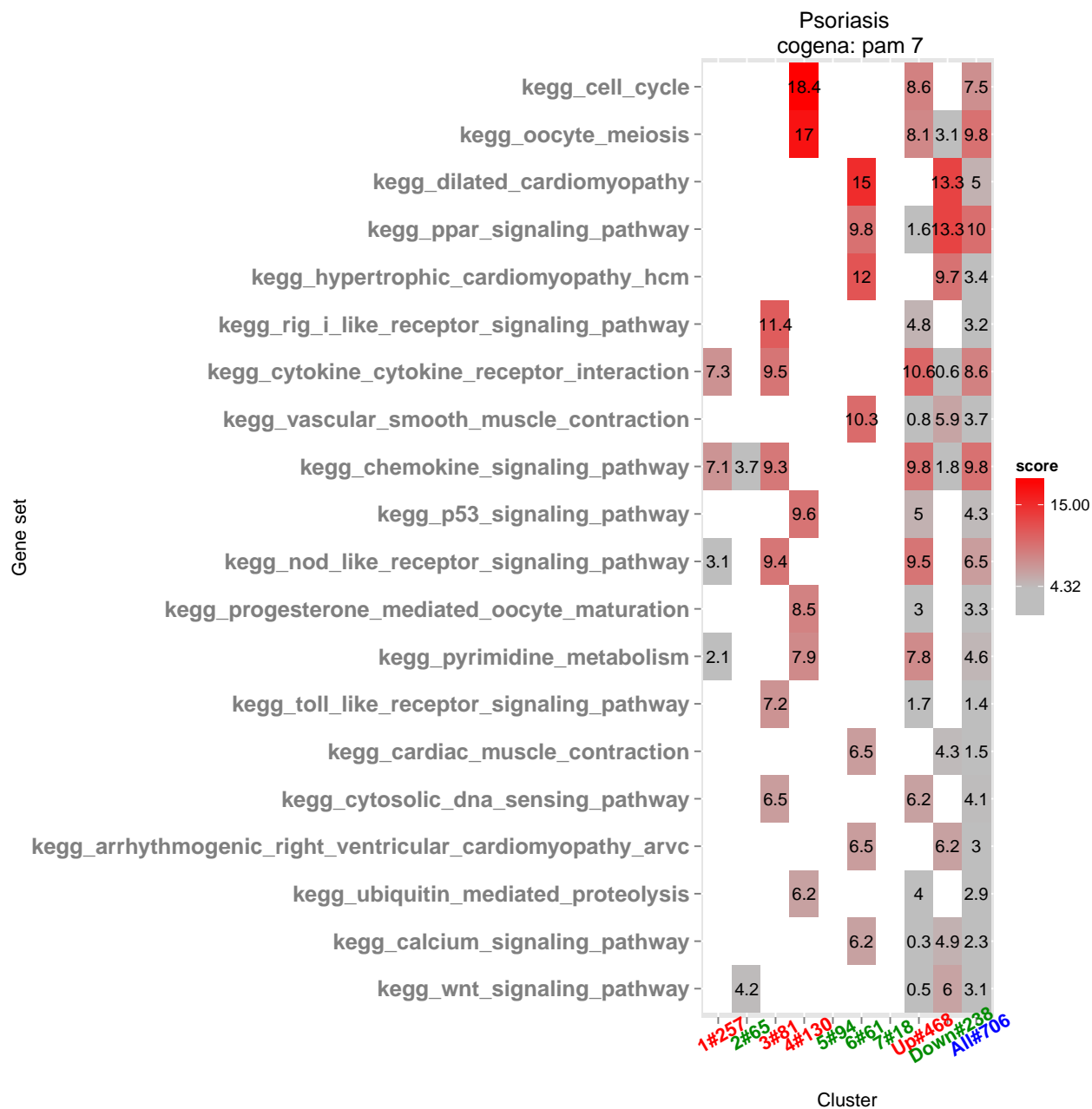


Figure 2: Pathway Analysis

```

expData <- as.data.frame(exprs(GSE13355rma))
expData$DESCRIPTION <- NA
expData <- expData[,c("DESCRIPTION", colnames(expData)[1:116])]

write.table(expData, file="../result/GSEA_input/GSE13355_exp.gct", sep="\t", quote=FALSE)
#####
# Add the following 3 lines at the beginning of GSE13355_exp.gct manually.
#1.2
54675 116
NAME
#####
write.table(t(as.character(GSE13355label$state)),
  file="../result/GSEA_input/GSE13355.cls",quote=FALSE, col.names=FALSE,
  row.names=FALSE)
#####
# Add the following 2 lines at the beginning of GSE13355.cls manually
116 2 1
#ct Psoriasis
#####
# Download gsea2-2.1.0.jar from the GSEA website
# Or from [my github](https://github.com/zhilongjia/geneRanking/blob/master/src/gsea2-2.1.0.jar)
# to the current directory.
#
# GSEA analysis
# java -cp ../gsea2-2.1.0.jar -Xmx512m xtools.gsea.Gsea -res ../result/GSEA_input/GSE13355_exp.gct
# -cls ../result/GSEA_input/GSE13355.cls -gmx ../result/GSEA_input/c2.cp.kegg.v5.0.symbols.gmt
# -collapse true -mode Max_probe -norm meandiv -nperm 1000 -permute phenotype
# -rnd_type no_balance -scoring_scheme weighted -rpt_label GSE13355 -metric Signal2Noise
# -sort real -order descending -chip ../result/GSEA_input/HG_U133_Plus_2.chip
# -include_only_symbols true -make_sets true -median false -num 100 -plot_top_x 20
# -rnd_seed timestamp -save_rnd_lists false -set_max 500 -set_min 15 -zip_report false
# -out ../result/GSEA_output -gui false

```

## 6 Drug repositioning by cogena

```

# Drug repositioning based on CmapDn100 gene set
cmapDn100_cogena_result <- clEnrich_one(genecl_result, "pam", "7",
  annofile=system.file("extdata", "CmapDn100.gmt.xz", package="cogena"),
  sampleLabel=sampleLabel)

# Drug repositioning based on CmapUp100 gene set
cmapUp100_cogena_result <- clEnrich_one(genecl_result, method="pam", nCluster="7",
  annofile=system.file("extdata", "CmapUp100.gmt.xz", package="cogena"),
  sampleLabel=sampleLabel)

```

### 6.1 Figure 3: Drug repositioning for cluster 3 (A)



# Figure 3

```
heatmapPEI(cmapDn100_cogena_result, "pam", "7", printGS=FALSE,
            orderMethod = "3", maintitle="Psoriasis")
```

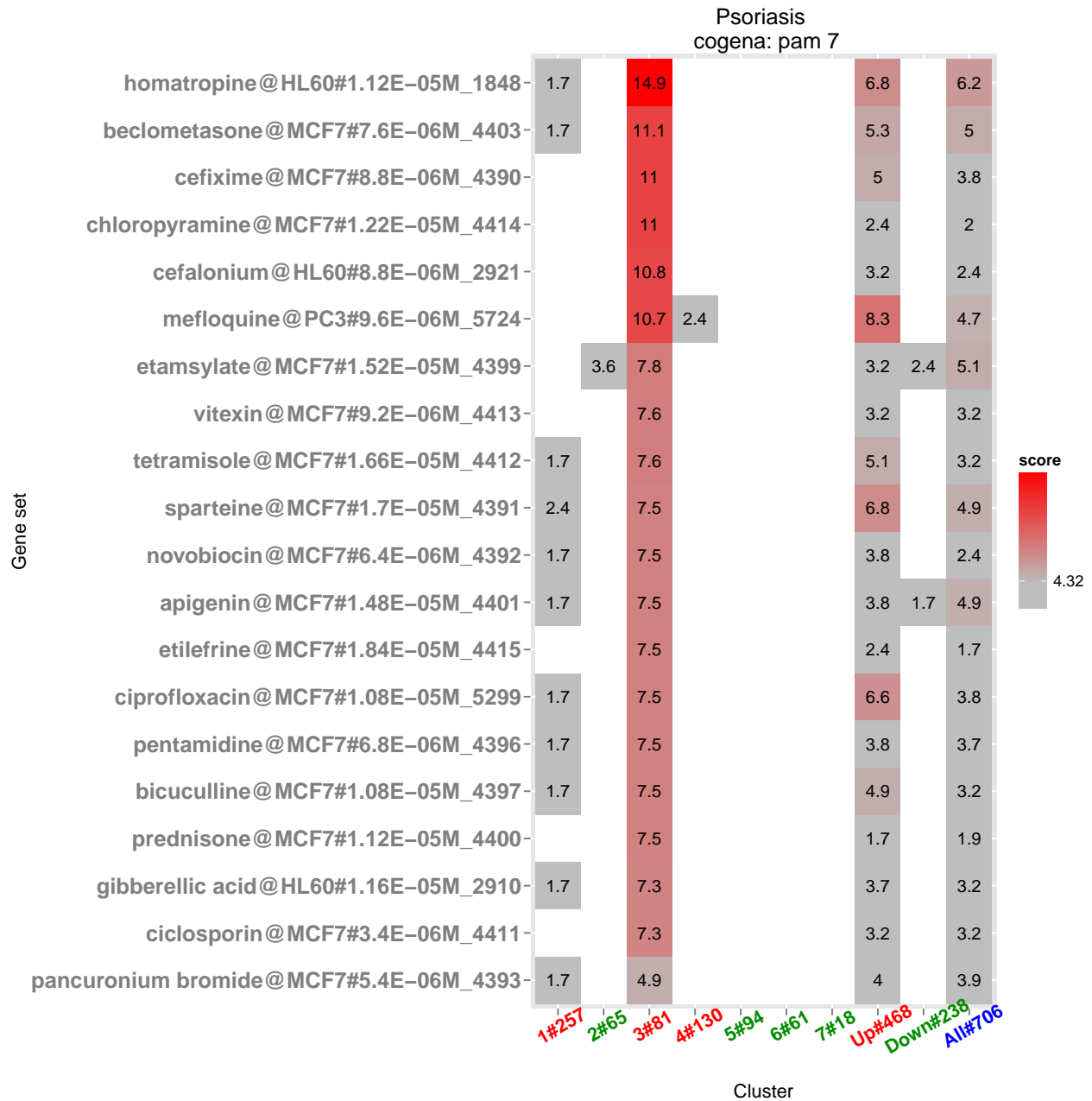


Figure 3: Drug Repositioning for cluster 3

## 6.2 Figure 4: Drug repositioning for cluster 4 (B)

# Figure 4

```
heatmapPEI(cmapDn100_cogena_result, "pam", "7", printGS=FALSE,
```

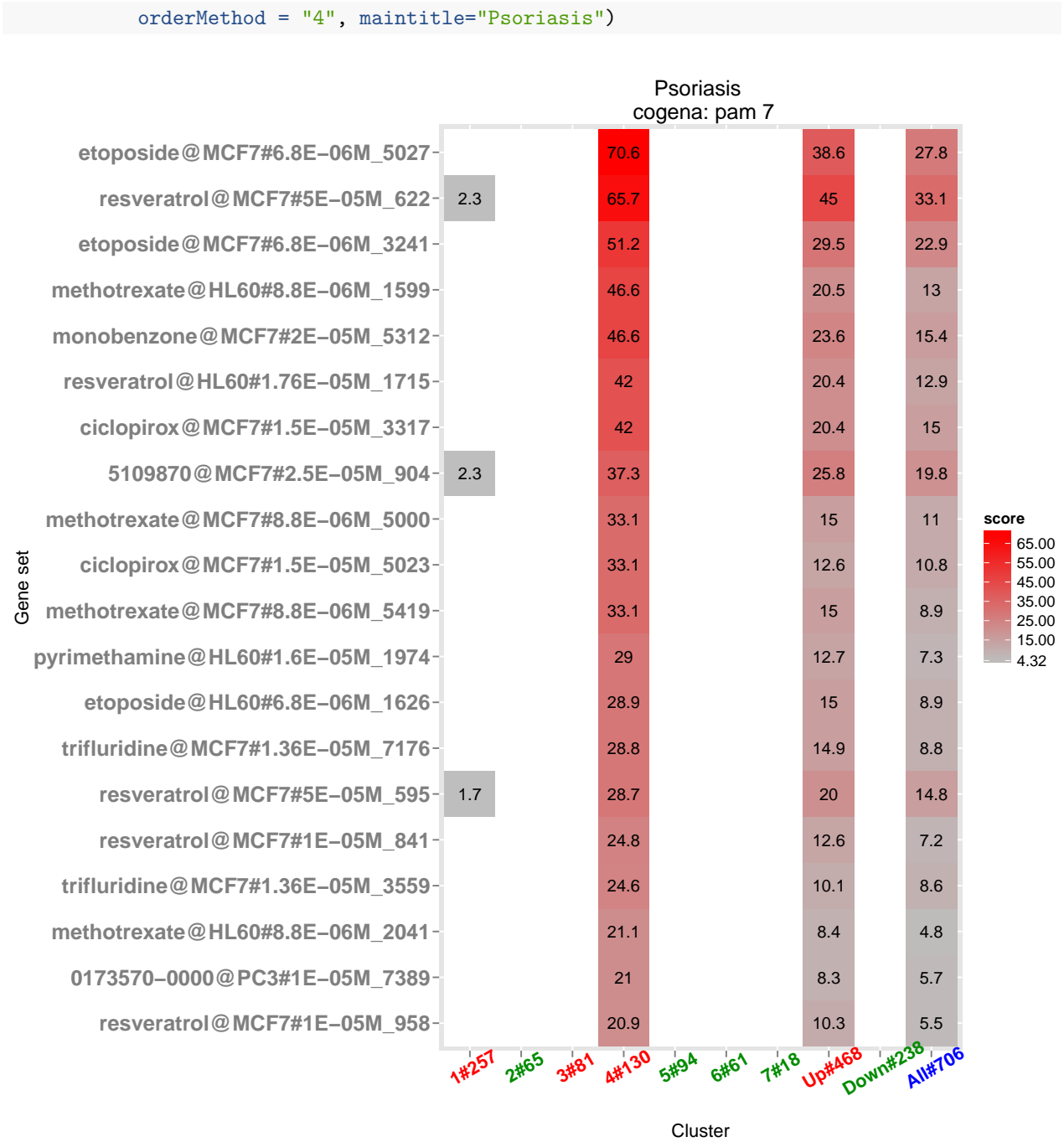


Figure 4: Drug Repositioning for cluster 4

### 6.3 Figure S1: Drug repositioning for cluster 6 (C)

```
# See Figure 5
heatmapPEI(cmapUp100_cogena_result, "pam", "7", printGS=FALSE,
            orderMethod = "6", maintitle="Psoriasis")
```

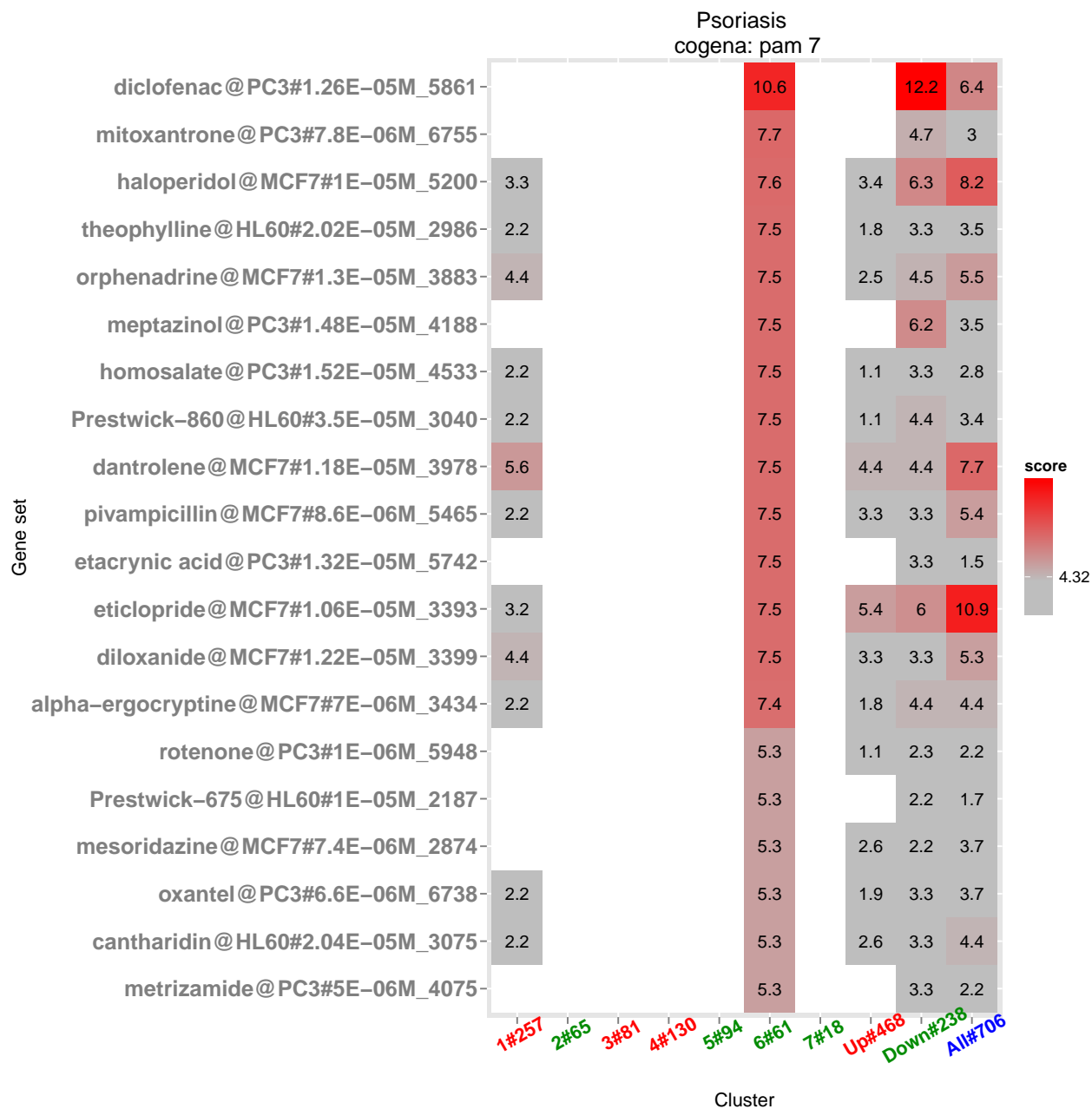


Figure 5: Drug Repositioning for cluster 6

## 6.4 Table S2: Output DEGs for CMAP and NFFinder Analysis

These outputs are used for [CMap](#) and [NFFinder](#) analysis to get the *Table S2*. The results can be obtained from `../result/CMAP_output` and `../result/NFFinder_output`, as well. For NFFinder, the CMap database and Profile matching, “Inverse”, are used.

```
# Convert gene symbols to probes in HGU133a.
symbol2Probe <- function(gs){
  library(hgu133a.db)
  p <- AnnotationDbi::select(hgu133a.db, gs, "PROBEID", "SYMBOL")$PROBEID
  p <- unique(p[which(!is.na(p))])
}

upGene <- rownames(GSE13355.limma[GSE13355.limma$logFC>= 1 & GSE13355.limma$adj.P.Val<=0.05,])
dnGene <- rownames(GSE13355.limma[GSE13355.limma$logFC<= -1 & GSE13355.limma$adj.P.Val<=0.05,])
upProbe <- symbol2Probe(upGene)
dnProbe <- symbol2Probe(dnGene)

# Output files for CMap and NFFinder
write.table(upProbe, file=paste0("../result/CMAP_input/", "GSE13355_Up.grp"),
  quote=F, col.names = F, row.names = F)
write.table(dnProbe, file=paste0("../result/CMAP_input/", "GSE13355_Dn.grp"),
  quote=F, col.names = F, row.names = F)
write.table(upGene, file=paste0("../result/NFFinder_input/", "GSE13355_Up.txt"),
  quote=F, col.names = F, row.names = F)
write.table(dnGene, file=paste0("../result/NFFinder_input/", "GSE13355_Dn.txt"),
  quote=F, col.names = F, row.names = F)
#####
```

## 7 System Info

```
## R version 3.2.0 (2015-04-16)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Debian GNU/Linux jessie/sid
##
## locale:
##  [1] LC_CTYPE=en_GB.UTF-8      LC_NUMERIC=C
##  [3] LC_TIME=en_GB.UTF-8      LC_COLLATE=en_GB.UTF-8
##  [5] LC_MONETARY=en_GB.UTF-8  LC_MESSAGES=en_GB.UTF-8
##  [7] LC_PAPER=en_GB.UTF-8     LC_NAME=C
##  [9] LC_ADDRESS=C             LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_GB.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
##  [1] stats4      tools      parallel    stats      graphics  grDevices  utils
##  [8] datasets   methods    base
##
## other attached packages:
##  [1] STRINGdb_1.8.1      hash_2.2.6          gplots_2.17.0
##  [4] RColorBrewer_1.1-2  plotrix_3.5-12      RCurl_1.95-4.7
##  [7] bitops_1.0-6        igraph_1.0.1        plyr_1.8.3
## [10] sqldf_0.4-10        gsubfn_0.6-6        proto_0.3-10
```

```

## [13] png_0.1-7          cogen_1.2.0          kohonen_2.0.18
## [16] MASS_7.3-43        class_7.3-13         ggplot2_1.0.1
## [19] cluster_2.0.3       limma_3.24.14        hgu133plus2.db_3.1.3
## [22] org.Hs.eg.db_3.1.2  RSQlite_1.0.0        DBI_0.3.1
## [25] annotate_1.46.1     XML_3.98-1.3         AnnotationDbi_1.30.1
## [28] GenomeInfoDb_1.4.1  IRanges_2.2.5        S4Vectors_0.6.3
## [31] MetaDE_1.0.5        combinat_0.0-8        impute_1.42.0
## [34] survival_2.38-3     hgu133plus2cdf_2.16.0 affy_1.46.1
## [37] GEOquery_2.35.4     Biobase_2.28.0       BiocGenerics_0.14.0
##
## loaded via a namespace (and not attached):
## [1] devtools_1.8.0      doParallel_1.0.8      R6_2.1.0
## [4] affyio_1.36.0       KernSmooth_2.23-15    lazyeval_0.1.10
## [7] colorspace_1.2-6    curl_0.9.1            compiler_3.2.0
## [10] git2r_0.10.1        preprocessCore_1.30.0 chron_2.3-47
## [13] biwt_1.0            formatR_1.2           xml2_0.1.1
## [16] caTools_1.17.1      scales_0.2.5          DEoptimR_1.0-3
## [19] mvtnorm_1.0-3       robustbase_0.92-5     stringr_1.0.0
## [22] apcluster_1.4.1     digest_0.6.8          rmarkdown_0.7
## [25] rrcov_1.3-8         htmltools_0.2.6       highr_0.5
## [28] BiocInstaller_1.18.4 mclust_5.0.2          gtools_3.5.0
## [31] dplyr_0.4.2         magrittr_1.5          Matrix_1.2-2
## [34] Rcpp_0.12.0         munsell_0.4.2         stringi_0.5-5
## [37] yaml_2.1.13         zlibbioc_1.14.0       grid_3.2.0
## [40] gdata_2.17.0        lattice_0.20-33       splines_3.2.0
## [43] knitr_1.10.5        tcltk_3.2.0          fastcluster_1.1.16
## [46] reshape2_1.4.1      codetools_0.2-14     evaluate_0.7
## [49] foreach_1.4.2       gtable_0.1.2          amap_0.8-14
## [52] assertthat_0.1      xtable_1.7-4          pcaPP_1.9-60
## [55] iterators_1.0.7     memoise_0.2.1         rversions_1.0.2
## [58] corrplot_0.73

```