Transcriptomic Profiling of Psoriatic Arthritis and Psoriasis Skin Lesions Reveals Shared and Distinct Gene Expression Signatures

M. Gage, T. Pralle, J. Vargo, D. Bolton Jr

2025-05-03

Document Description

This document includes the applicable code, plots, and written assessments for the differential expression analysis of psoriasis (PsO), psoriatic arthritis (PsA), and ankylosing spondylitis (CTL) samples extracted from the GSE186063 data set on the Gene Expression Omnibus. Each sample contains high-throughput sequencing data of paired-end RNA reads of skin biopsies. Biopsies were taken from a cohort consisting of five patients with a dermatologist-confirmed diagnosis of psoriasis without concurrent psoriatic arthritis, five patients with diagnosed psoriatic arthritis, and five patients with ankylosing spondylitis and no history of psoriasis (reference group). Sample reads (FASTA files) were QC'd, trimmed, aligned to a human reference genome, and used to generate counts in Galaxy before differential expression analysis.

Loading in Data/Merging Counts Files

```r
# Set the working directory via setwd() or via Session->Set Working Directory->Choose Directory

# Reading in the GSE186063 GEO counts files AND assigning column names for each.
counts_298 <- read.table("PsA_298.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsA_298"))
counts_300 <- read.table("PsA_300.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsA_300"))
counts_302 <- read.table("PsA_302.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsA_302"))
counts_333 <- read.table("PsA_333.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsA_333"))
counts_350 <- read.table("PsA_350.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsA_350"))

counts_325 <- read.table("PsO_325.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsO_325"))
counts_327 <- read.table("PsO_327.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsO_327"))
counts_344 <- read.table("PsO_344.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsO_344"))
counts_358 <- read.table("PsO_358.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsO_358"))
```

```r
counts_356 <- read.table("PsO_356.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","PsO_356"))

counts_303 <- read.table("CTL_303.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","CTL_303"))
counts_305 <- read.table("CTL_305.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","CTL_305"))
counts_317 <- read.table("CTL_317.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","CTL_317"))
counts_345 <- read.table("CTL_345.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","CTL_345"))
counts_322 <- read.table("CTL_322.tabular",sep="\t",skip=0,nrows=78932,quote="\"",
                         header=FALSE,col.names=c("Gene_ID","CTL_322"))


# Creating a list of all the counts data
counts_list <- list(counts_298, counts_300, counts_302, counts_333, counts_350,
                    counts_325, counts_327, counts_344, counts_358, counts_356,
                    counts_303, counts_305, counts_317, counts_345, counts_322)

# Merge all counts data frames by the "Gene" column
merged_counts <- Reduce(function(x, y) merge(x, y, by="Gene_ID", all=TRUE), counts_list)

# Including only the sample counts and moving the Gene_ID column to become the row names
counts_all <- merged_counts[,c(2:16)]
row.names(counts_all) <- merged_counts[,1]

# Saving the merged table as a .tsv file
write.table(merged_counts, "merged_counts.tsv", sep="\t", row.names=FALSE, quote=FALSE)
```

DESeq2 Data Preparation

```r
# load libraries
library(DESeq2)
library(pheatmap)
library(EnhancedVolcano)
library(org.Hs.eg.db)

# Read in table of gene annotations
annotation <- read.table("human_genome_annot.txt",header=TRUE,sep="\t",quote="\"")

# Preparing the NON-log2 transformed raw counts for DESeq2 (IDs --> column names)
rnaseqMatrix <- merged_counts[,c(2:16)]
row.names(rnaseqMatrix) <- merged_counts[,1]
head(rnaseqMatrix)

# Define how the samples map to different sample groups
samples <- data.frame(matrix(c("PsA_298", "PsA_300", "PsA_302", "PsA_333", "PsA_350",
                               "PsO_325", "PsO_327", "PsO_344", "PsO_358", "PsO_356",
                               "CTL_303", "CTL_305", "CTL_317", "CTL_345", "CTL_322",
                               "PsA", "PsA", "PsA", "PsA", "PsA",
                               "PsO", "PsO", "PsO", "PsO", "PsO",
                               "CTL", "CTL", "CTL", "CTL", "CTL"),ncol=2))

# Renaming the columns, setting row names, and creating factor groups
names(samples) <- c("ID","Disease")
rownames(samples) <- samples[,1]
samples$Disease <- factor(c("PsA", "PsA", "PsA", "PsA", "PsA",
                            "PsO", "PsO", "PsO", "PsO", "PsO",
                            "CTL", "CTL", "CTL", "CTL", "CTL"))

# Create the DEseq2DataSet object
deseq2Data <- DESeqDataSetFromMatrix(countData = rnaseqMatrix,
                                     colData = samples,
                                     design = ~ Disease)

# Determine how many genes were lowly expressed and decide on a threshold
dim(deseq2Data)
dim(deseq2Data[rowSums(counts(deseq2Data)) > 10, ]) # how many genes have more than 10 reads?
# (this number can be set higher with more sample groups)

# Filter out lowly expressed genes
deseq2Data <- deseq2Data[rowSums(counts(deseq2Data)) > 10, ]
```

DESeq2 Diagnostic Analysis

```
# Pairwise scatterplots of log2 transformed data (THIS CAN TAKE A LONG TIME TO RUN)
pdf("pairwise_scatter.pdf")
pairs(log2(counts(deseq2Data)+1), pch = '.', xlim=c(0,18), ylim=c(0,18))
dev.off()

# Run pipeline for differential expression steps
deseq2Data <- DESeq(deseq2Data)

# rlog transform counts
rld <- rlog(deseq2Data, blind=FALSE)
rlogcounts <- data.frame(assay(deseq2Data))
rownames(rlogcounts) <- rownames(deseq2Data)

# PCA plot
plotPCA(rld, intgroup=c("Disease"))
```

```
# PCA plot data
plotPCA(rld, intgroup=c("Disease"), returnData=TRUE)
```

Developing the DEG List and Associated Statistics

```
# Here, we are comparing each sample group.
# The current code reflects the PsO vs. CTL comparison.
```

```r
# No DEGs were statistically significant after FDR adjustments for PsA vs. PsO

# Pairwise contrast results between two groups.
res_PsO_CTL <- results(deseq2Data, contrast=c("Disease", "PsO", "CTL"))
resOrdered_PsO_CTL <- res_PsO_CTL[order(res_PsO_CTL$pvalue),]

# Get number of differentially expressed data at different significance thresholds
summary(resOrdered_PsO_CTL)
sum(resOrdered_PsO_CTL$padj < 0.05, na.rm=TRUE)
sum(resOrdered_PsO_CTL$pvalue < 0.05, na.rm=TRUE)

# MA plot
pdf("PsO_CTL_MA_plot.pdf")
plotMA(resOrdered_PsO_CTL, ylim = c(-9, 9),  main = "MA Plot: PsO vs. CTL")
dev.off()


#Uncomment if you want to pick on points in the MA plot
#plotMA(res_PsO_CTL)
#idx <- identify(res_PsO_CTL$baseMean, res_PsO_CTL$log2FoldChange)
#rownames(res_PsO_CTL)[idx]


# Save the results as a data frame
results <- data.frame(resOrdered_PsO_CTL)
head(results)

# Annotate results
results.annot <- merge(results,annotation,by.x=0,by.y=1)
results.annot <- results.annot[order(results.annot$padj),]
names(results.annot) <- c("GeneID","baseMean","log2FoldChange","lfcSE","stat","pvalue","padj","Symbol",

head(results.annot)

write.table(results.annot,"annot_gene_list_PsO_CTL.txt",sep="\t",row.names=FALSE)
```

Plots

```r
#*******************************************
# Volcano Plot
#*******************************************

# You will have to edit labels and file name for the treatment group you are working with

ens <- rownames(deseq2Data)
symbols <- mapIds(org.Hs.eg.db, keys = ens,
                  column = c('SYMBOL'), keytype = 'ENSEMBL')
symbols <- symbols[!is.na(symbols)]
symbols <- symbols[match(rownames(deseq2Data), names(symbols))]
rownames(deseq2Data) <- symbols
keep <- !is.na(rownames(deseq2Data))
```

```
deseq2Data <- deseq2Data[keep,]

res <- results(deseq2Data,
               contrast = c("Disease", "PsO", "CTL"))

res <- lfcShrink(deseq2Data,
                 contrast = c("Disease", "PsO", "CTL"), res=res, type = 'normal')

pdf("PsO_CTL_Volcano_plot.pdf")
EnhancedVolcano(res,
                lab = rownames(res),
                x = 'log2FoldChange',
                y = 'pvalue',
                pCutoff = 0.000000001,
                drawConnectors = TRUE,
                widthConnectors = 0.5,
                FCcutoff = 1.0,
                pointSize = 2.0,
                labSize = 3.0,
                title = 'PsO vs CTL Volcano Plot')
dev.off()




#*******************************************
# Heatmap
#*******************************************


# Candidate genes (you can select any genes you want on BioMart. The current selected
# genes are the top 20-30 genes in the DEG lists for PsO and PsA, but any genes may be chosen)
candidates <- read.delim("candidate_genes_mart.txt",header=TRUE)
names(candidates) <- c("ID","Symbol", "Biotype")

# Subset results with data just for candidate genes
table_hits <- rlogcounts[candidates$ID,]

# Replace any NA's with 0
table_hits[is.na(table_hits)] <- 0

# Replace Ensembl Gene IDs with symbols
row.names(table_hits) <- candidates$Symbol

cal_z_score <- function(x){
  (x - mean(x)) / sd(x)
}

table_hits_norm <- t(apply(table_hits, 1, cal_z_score))

pdf("heatmap_candidate_genes.pdf")
pheatmap(table_hits_norm,name = "Row Z-Score",cluster_cols = FALSE)
```

```
dev.off()
```