

# Vignette: NIBMG survival analysis

*Samanwoy Mukhopadhyay*

*28 June 2019*

## Contents

<b>1</b>	<b>General summary</b>	<b>1</b>
<b>2</b>	<b>Getting the data</b>	<b>1</b>
2.1	Installation of the Data Package <b>ssnibmgsurv</b>	1
2.2	Installation of the Data Package <b>nibmgsurvmeta</b>	1
2.3	Running the analysis : Setting working directory	2
<b>3</b>	<b>Case control analysis Getting genes and pathways DE in sepsis</b>	<b>2</b>
3.1	Genome-level changes in gene expression	4
3.2	Pathway analysis results for case vs control design	5
<b>4</b>	<b>Survival analysis</b>	<b>7</b>
4.1	Survival analysis using Published transcriptome (from GEO) data on sepsis.	7
4.2	Pathway analysis of NIBMG cohort for survival	10
4.3	Down-regulation of NF- $\kappa$ B signalling pathway genes in non-survivors	10
4.4	Relative gene expression of the targets of NF- $\kappa$ B	12
4.5	M1 macrophage specific gene expression associated with survival	13
4.6	Down-regulation of Antigen processing and presentation signalling pathway genes in Non-survivors	15
4.7	Down-regulation of T cell receptor signalling pathway genes in Non-survivors	16
4.8	Key Biological Modules altered in Survivor	17
<b>5</b>	<b>Analyzing the heterogeneity of NIBMG sepsis cases</b>	<b>19</b>

## 1 General summary

Sepsis is one of the leading diseases associated with high mortality and morbidity, due to the systemic nature of this illness, transcriptomic technology is particularly suited to investigation of molecular underpinning of survival from sepsis episodes. We adopted an analysis approach that combined published transcriptome data and data generated in our laboratory from Indian sepsis patients leading to the discovery of key immune pathways to be altered in non-survivors compared to survivors. This is the first clinical transcriptomic study on sepsis from India, showing that non-survival is associated with down-regulated adaptive immune pathways and significant M2-specific immune-suppression, possibly regulated by NF- $\kappa$ B signalling. 3 Biological process related to sepsis was found to be significantly altered in non-survivors. A patient specific analysis reveals up-regulation of coagulation and inflammation but a strong down-regulation of immunosuppression modules in Indian sepsis patients.

## 2 Getting the data

Availability of data and materials Data and R code are available at the following link (<https://figshare.com/>) (search for the project “**ss\_survival\_nibmg**”). The whole data can be downloaded as a single zip file (“**ss\_survival\_nibmg\_doc.zip**”). Upon uncompressing the zip, instructions for running the code and generating the figures of this manuscript are available in the file howto.pdf.

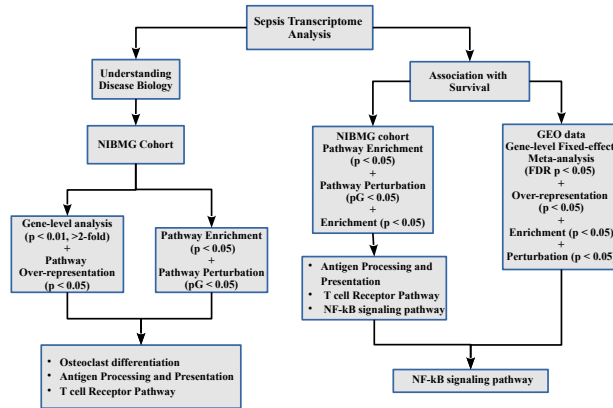


Figure 1: Analysis flow with results: The left arm describes the analysis flow for Sepsis Cases vs healthy controls whereas the right arm describes the analysis for the Survivor vs Non-survivor analysis

## 2.1 Installation of the Data Package ssnibmgsurv

Download the file **ssnibmgsurv\_1.0.tar.gz** from <https://figshare.com/> (search for **ssnibmgsurv**) Change the directory to where you saved the file. Start R. At the R prompt, issue the following command:

```
install.packages(pkgs="ssnibmgsurv_1.0.tar.gz", repos=NULL)
# Now the data package ssnibmg is installed on your computer.
# Check with the following command:
library("ssnibmgsurv")
```

## 2.2 Installation of the Data Package nibmgsurvmeta

Download the file **nibmgsurvmeta\_1.0.tar.gz** from <https://figshare.com/> (search for **nibmgsurvmeta**) Change the directory to where you saved the file. Start R. At the R prompt, issue the following command:

```
install.packages(pkgs="nibmgsurvmeta_1.0.tar.gz", repos=NULL)
# Now the data package ssnibmg is installed on your computer.
# Check with the following command:
library("nibmgsurvmeta")
```

Now both the data packages are installed in your computer; let's focus on starting the analysis.

## 2.3 Running the analysis : Setting working directory

It is assumed that you have access to a folder **ss\_survival\_nibmg\_doc** Start R and set the working directory to **ss\_survival\_nibmg\_doc**. Run the code chunks of vignette as it follows.

# 3 Case control analysis Getting genes and pathways DE in sepsis

```
# Clear the workspace and close any graphics window if open
rm(list=ls())
graphics.off()
```

```

# loading the priliminary libraries datapackages
source("Rcode/prelim.R")

source("Rcode/getData.R")
# A age and gender matched case vs control analysis
matched.12 <- c("C11","C8","C1","C7","C17","C10","C21","C18","C20","C4","C9",
               "C12","42D1","1D1","8D1","50D1","60D1","90D1","62D1","70D1",
               "19D1","32D1","14D1","61D1")
esetm <- eset[, matched.12]
rttm <- rowttests(esetm, "Group")
lfcdm<-rttm$dm
pdm <- p.adjust(rttm$p.value, method="BH")
names(pdm) <- rownames(rttm)
names(lfcdm) <- rownames(rttm)
egs.all <- featureNames(esetm)
# Removing some variables that are not to be used for further analysis
rm(snames, ptids)

# Draw a volcano plot to show that 24% of the genome
# are perturbed in sepsis (FDR p < 0.05)
# sepsis and matched control samples are used here
source("Rcode/makeplot_1_volcano_for_genomic_storm.R")
box()

```

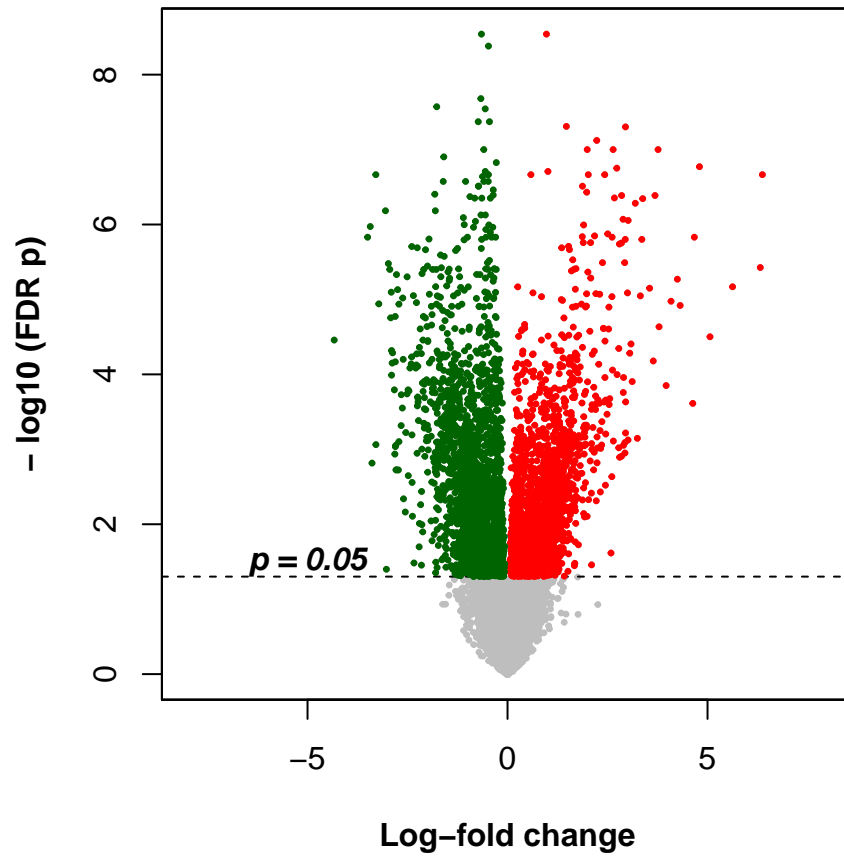


Figure 2: Volcano plot showing 24 percent of the genome perturbed in sepsis compared to healthy control (FDR  $p < 0.05$ ). This establishes large scale change in gene expression in sepsis, and possible multiple pathways being perturbed.

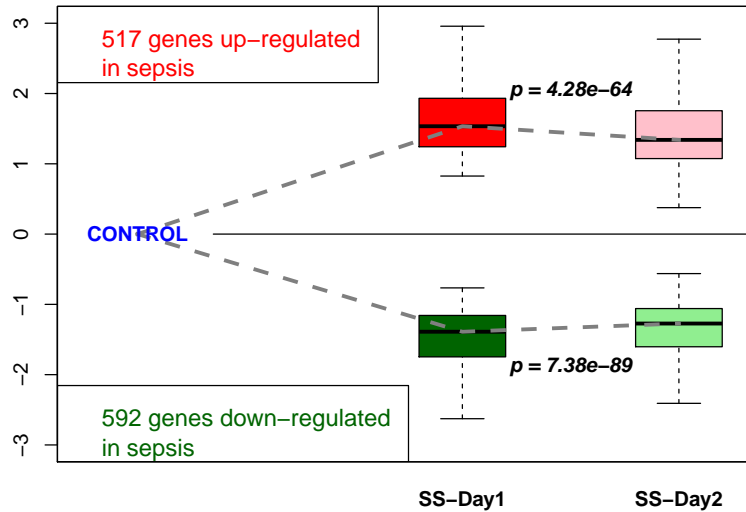


Figure 3: Temporal change of DE genes (FDR  $p < 0.05$ , 2 fold-change or more), there is a non-random trend toward the baseline with time (p-values from paired t-tests are provided in the legend). This is consistent with earlier findings from patients with trauma (USA Gluegrant project)

### 3.1 Genome-level changes in gene expression

Differential gene expression analysis revealed 1109 genes to be altered in sepsis patients compared to age and gender matched healthy controls. Volcano plot (Figure 2) showing 24% of the genome perturbed in sepsis compared to healthy control (FDR  $p < 0.05$ ). This establishes large scale change in gene expression in sepsis, and possible multiple pathways being perturbed.

```
# Detect the highly significant DE genes of sepsis -
# FDR p < 0.01; 2-fold
# Draw a temporal box-plot to show temporal
# changes in control vs cases
upg <- egs.all[which(pdm<0.01 & lfcdm>1)]
downg <- egs.all[which(pdm<0.01 & lfcdm<(-1))]
deg.d1<-union(upg,downg)

# to be used later for survival vs nonsurvival
source("Rcode/makeplot_2_temporal_plot_for_DEgns.R")
box()

# draw trajectory of DE genes survivor versus non-survivor
source("Rcode/drawDEtrajectorySurvival.R")
box()
```

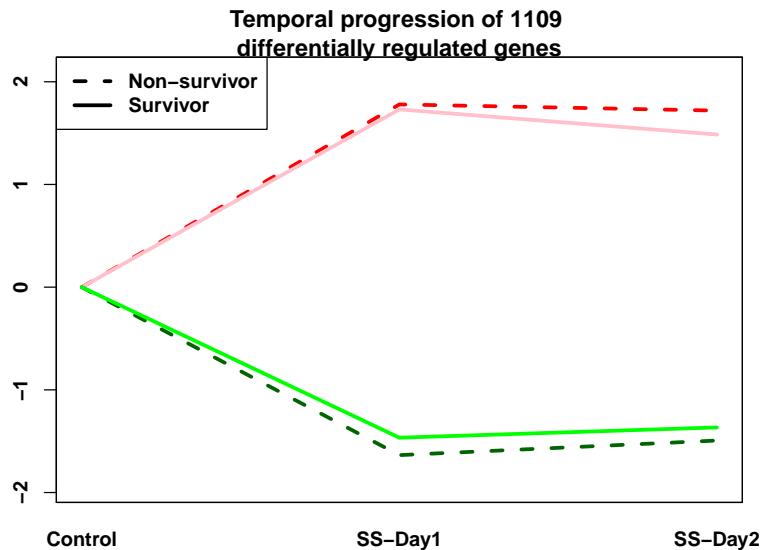


Figure 4: Temporal change of DE genes (FDR  $p < 0.05$ , 2 fold-change or more), there is a non-random trend toward the baseline with time (p-values from paired t-tests are provided in the legend). The delayed return to baseline is associated with non-recovery from sepsis.

### 3.2 Pathway analysis results for case vs control design

```
# Pathway Analysis: ORA, GSEA, SPIA
# ORA - Over representation analysis
pORAup <- getORApvals(upg, egs.all)[,"p"]
pORAup <- p.adjust(pORAup, method="BH")
pORAdown <- getORApvals(downg, egs.all)[,"p"]
pORAdown <- p.adjust(pORAdown, method="BH")

NIBMG.ORA.disease <- cbind(pORAup, pORAdown)
colnames(NIBMG.ORA.disease) <- paste(colnames(NIBMG.ORA.disease), "NIBMG.disease", sep="_")

# GSEA, SPIA
# Warning running this code will take long time (approx 10 minutes each)
source("Rcode/run_GSEA_sepsis_vs_control.R") # GSEA

##
## Loading GSEA permutation t.test result from file ...
## done!

source("Rcode/run_SPIA_sepsis_vs_control.R") # SPIA

## Loading SPIA
## result from file ... done!

# Combine result from 3 pathway analyses and print the Down and Up pathways#
pathsDown = intersect(intersect(names(which(pGSEAdown < 0.01)),
                                names(which(pORAdown < 0.01))), names(which(pG < 0.01)))
pathways.list[ paste0("path:", pathsDown)]

##
## path:hsa03013
## "RNA transport - Homo sapiens (human)"
## path:hsa04612
```

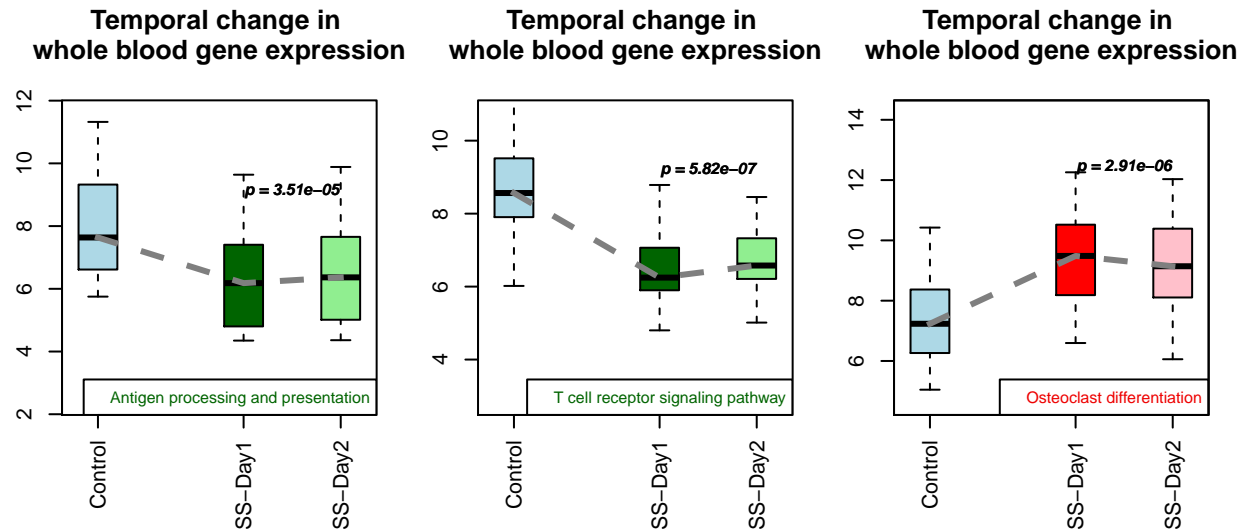


Figure 5: Temporal boxplot of 3 key pathways altered in Sepsis by combined pathway analysis.

```
## "Antigen processing and presentation - Homo sapiens (human)"
##                               path:hsa04660
## "T cell receptor signaling pathway - Homo sapiens (human)"
##                               path:hsa05332
##                               "Graft-versus-host disease - Homo sapiens (human)"

# Removing some variables that are not to be used for further analysis
rm(pathsDown)

pathsUp = intersect(intersect(names(which(pGSEaup < 0.01)),
                             names(which(pORAup < 0.01))), names(which(pG < 0.01)))
pathways.list[ paste0("path:", pathsUp)]

##                               path:hsa04380
## "Osteoclast differentiation - Homo sapiens (human)"
##                               path:hsa05133
## "Pertussis - Homo sapiens (human)"
##                               path:hsa05150
## "Staphylococcus aureus infection - Homo sapiens (human)"
##                               path:hsa05202
## "Transcriptional misregulation in cancer - Homo sapiens (human)"
##                               path:hsa05322
## "Systemic lupus erythematosus - Homo sapiens (human)"

# Removing some variables that are not to be used for further analysis
rm(pathsUp)

par(mfrow=c(1,3))
# Two pathways are down-regulated in sepsis
box.plot.KEGG(id="hsa04612", direction="down") # Antigen processing and presentation
box.plot.KEGG(id="hsa04660", direction="down") # T cell receptor signaling
# Up-regulated pathway: Osteoclast Differentiation
box.plot.KEGG(id="hsa04380", direction="up") # Osteoclast Differentiation
box()
```

## 4 Survival analysis

### 4.1 Survival analysis using Published transcriptome (from GEO) data on sepsis.

```
### Section B: Survival analysis
### Getting genes and pathways associated with survival
### uses NIBMG and published data sets
#####
rm(list=ls())
graphics.off()

#####
# Preliminaries
source("Rcode/prelim.R")

# Get the expression set
source("Rcode/getData.R")
library("nibmgsurvmeta")
data(ss.list) # eight data sets = 4 adult + r child
data(ss.surv.list) # 8 datasets without control samples
studies = read.table(file="metadata/studies.txt", header=TRUE, sep="\t")
study.ids = as.character(studies$study.id)
study.type = as.character(studies$age)
names(study.type) = study.ids
rm(studies)

#####
# perform differential expression analysis for survivors
# and non survivors day 1 in NIBMG data
#####

#####
# Day 1 Nsurv vs Surv with p adjusted
rtt1 <- rowttests(eset.s, factor(eset.s$Outcome))
sel1 <- rownames(rtt1)[which(rtt1$p.value < 0.01)]
sel1.nibmg.lfc <- rtt1[sel1,]

#####
# Hierarchical clustering of LFC of 8 studies
source("Rcode/run_hclust.R")
box()

# Pathway analysis in GEO data
# ORA: Over-representation Analysis
source("Rcode/run_ORA_surv_nonsurv_analysis.R")
# Perform Permutation GSEA analysis
source("Rcode/run_GSEA_surv_nonsurv_analysis.R")

## Loading gsea_child data from file ... done!
## File exists ...
## Done!...
## Loading gsea_adult data from file ... done!
## File exists ...
## Done!...
```



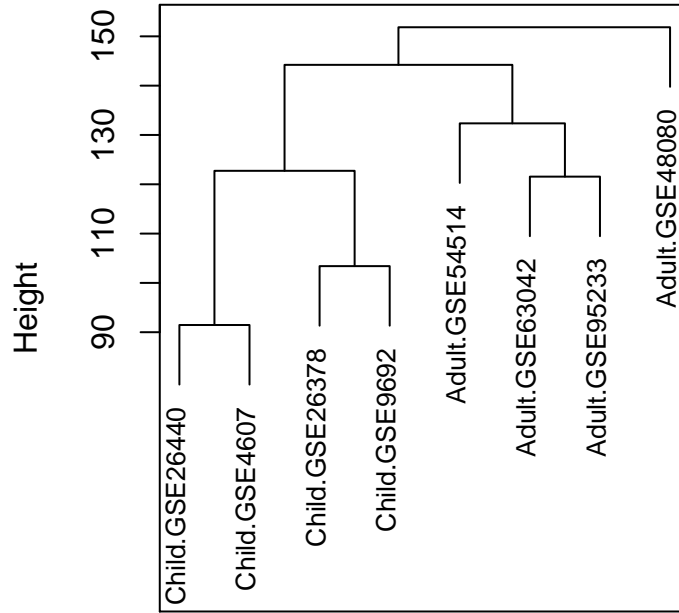


Figure 6: Survival analysis with eight published data sets: human adults and children with sepsis; hierarchical clustering with log-fold change in gene expression led to evidence of developmental age-specific differential perturbation; i.e., separate clusters for adult and child data sets. In view of this difference, further analysis was confined to Adult data sets when combined with NIBMG data.

```
#####
# twoway evidence plot adult
source("Rcode/run_SPIA_surv_nonsurv_analysis_adult.R")

res <- spia.res[[1]]
resall.adult <- data.frame(rep(names(ss.adult)[1], nrow(res)),
                           res$Name, res$ID, res$NDE,
                           res$pNDE, res$tA, res$pPERT,
                           res$pG, res$pGFdr, res$Status)

col.nm <- as.character(sapply(strsplit(colnames(resall.adult), "res."), "[[", 2))
col.nm[1] <- "Study"

colnames(resall.adult) <- col.nm

for(i in 2:length(ss.adult)) {
  res <- spia.res[[i]]
  resedited <- data.frame(rep(names(ss.adult)[i],
                              nrow(res)), res$Name, res$ID, res$NDE, res$pNDE,
                              res$tA, res$pPERT, res$pG, res$pGFdr, res$Status)

  colnames(resedited) <- col.nm
  resall.adult <- rbind(resall.adult, resedited, deparse.level=0)
}

#####
# Calculate Fisher's product of p-values of
# perturbation for all pathways
```

```

#
keggs <- as.character(unique(resall.adult$ID))

# Create empty vector for capturing fisher product of pG
pPERT.Fp.adult <- vector(mode="numeric", length=length(keggs))
names(pPERT.Fp.adult) <- keggs

for(id in keggs) {
  pvec <- resall.adult[resall.adult$ID==id, "pPERT"]
  pPERT.Fp.adult[id] <- Fisher.test(pvec)["p.value"]
}
rm(pvec)

# Get the two p values for combining
# FDR correct the Perturbation value
pb <- pPERT.Fp.adult
pb <- p.adjust(pb, "fdr")
ph <- as.numeric(pNDE.paths.ad[paste("hsa", names(pb), sep=""), "p"])
names(ph) <- names(pb)

# Use a floor value for p
ph[ph < 1e-07] <- 1e-07
pb[pb < 1e-07] <- 1e-07

pGmeta.adult <- combfunc(pb, ph, "fisher")

# Capture the fisher product of pPERT pG Meta p values
# into a dataframe
fisher.prod.spia.adult <- as.data.frame(cbind(paste("hsa", names(pPERT.Fp.adult), sep=""),
                                             as.numeric(ph), as.numeric(pPERT.Fp.adult),
                                             as.numeric(pGmeta.adult)))
colnames(fisher.prod.spia.adult) <- c("paths", "pNDE.adult", "pPERT.adult", "pG.meta.adult")

#####
# Following code is derived from SPIA::plotP
# The pG threshold is the p-value 0.05 corrected for the number of
# pathways being considered
tr= 0.05
#tr<- 0.05/length(pb)

# plot neg.log.p_PERT against neg.log.p_NDE
plot(-log(ph), -log(pb), col="gray80",
     xlim = c(0, max(c(-log(ph), -log(pb)) +1, na.rm = TRUE)),
     ylim = c(0, max(c(-log(ph), -log(pb)) +1, na.rm = TRUE)),
     pch = 19, main = "Two-way evidence plot : Adult Sepsis", cex = 1.5,
     xlab = "Evidence of Over-representation, -log(p_ORA)",
     ylab = "Evidence of Perturbation, -log(p_PERT)")

# For selected pathways for visualisation: NLR, NFkB, Osteoclast
#####
sel.paths.ad <- c("04621", "04064", "04380")
col.vec <- c("red2", "purple2", "darkblue")

```

## Two-way evidence plot : Adult Sepsis

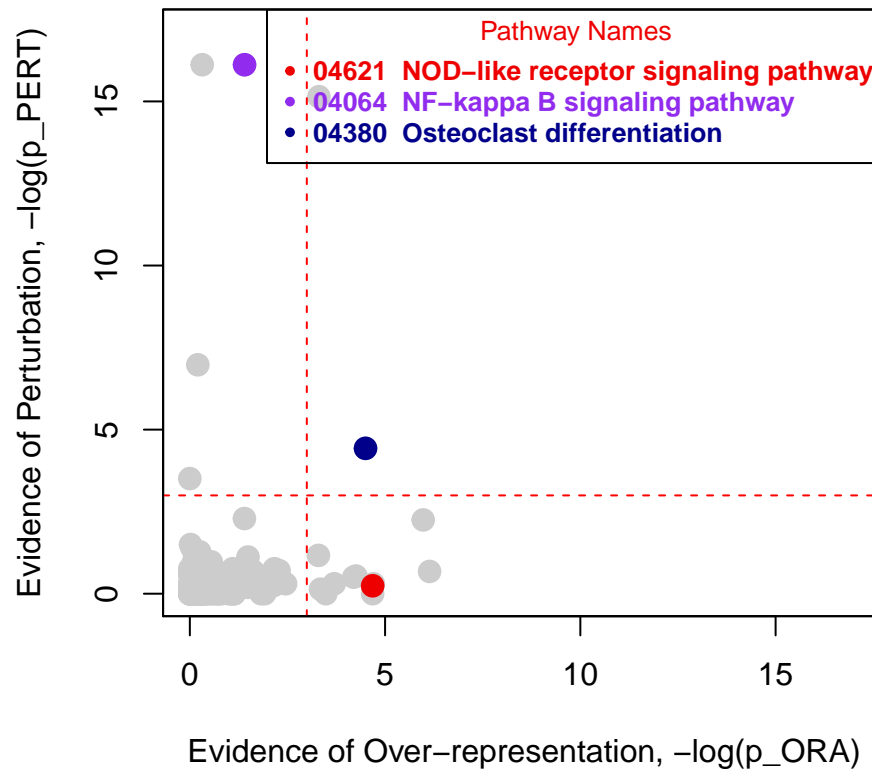


Figure 7: KEGG pathways associated with survival in GEO data; NF-kappaB signalling pathway, Osteoclast differentiation and NOD-like receptor signalling pathway.

```
points(-log(ph)[sel.paths.ad ], -log(pb)[sel.paths.ad ], pch = 19, col = col.vec,
       cex = 1.5)
abline(v = -log(tr), lwd = 1, col = "red", lty = 2)
abline(h = -log(tr), lwd = 1, col = "red", lty = 2)
path.nms <- as.character(sapply(strsplit(pathways.list[paste("path:hsa", sel.paths.ad, sep="")], " -"),
# Add a legend to the plot
legend("topright", title="Pathway Names", legend= paste(sel.paths.ad , path.nms, sep=" "), text.font=
box())
```

### 4.2 Pathway analysis of NIBMG cohort for survival

### 4.3 Down-regulation of NF- $\kappa$ B signalling pathway genes in non-survivors

```
rm(list=ls())
#####
# Preliminaries
source("Rcode/prelim.R")

# Get the expression set
source("Rcode/getData.R")
```

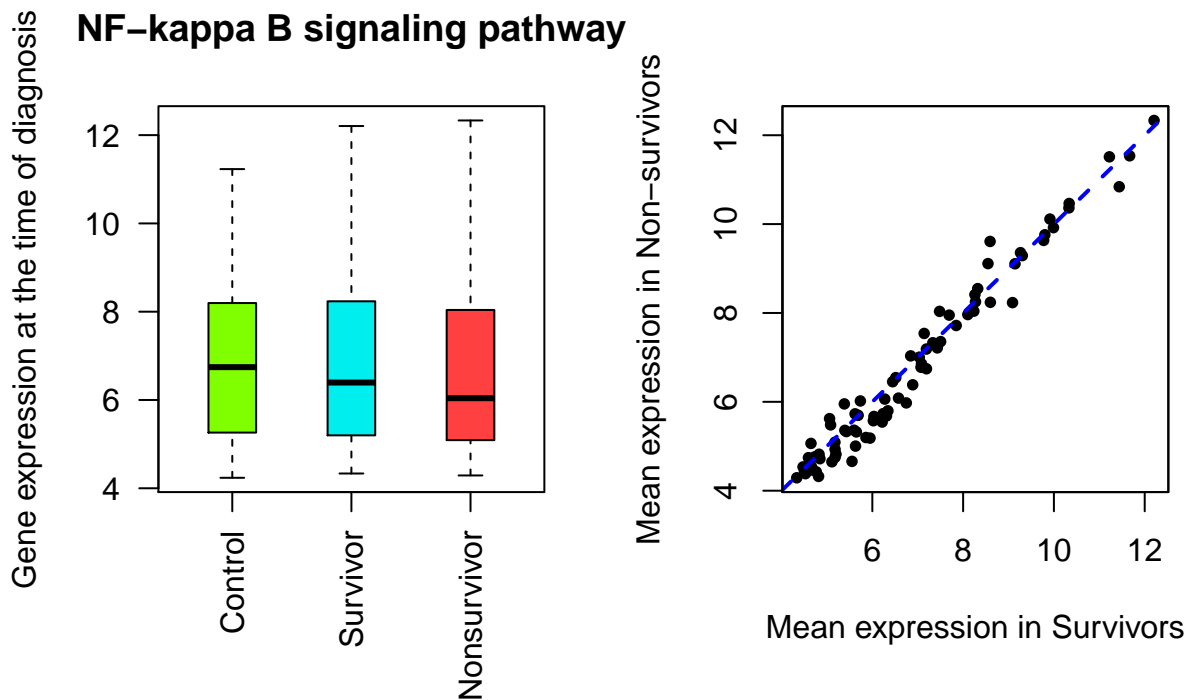


Figure 8: NF-kappa B signalling pathway boxplot and scatterplot

```
# show gene expression trend for NF-kB signaling pathway
getPval(keggid="hsa04064", drawPlot=TRUE, getSigGenes=TRUE)
```

```
## [1] "TNFRSF13C" "CSNK2A2" "ICAM1"
## [4] "LCK" "BCL2A1" "RELB"
## [7] "BCL2L1" "TRAF5" "BCL10"
## [10] "CD40"
```

```
box()
```

```
# Draw the permutation histogram of GSEA for NF-kB signaling pathway
drawPermutHist(keggid="hsa04064", eset=eset.s, fac=factor(as.character(eset.s$Outcome)))
```

```
## p.val zobs
## 0.044100 -3.860523
```

```
box()
```

## NF-kappa B signaling pathway

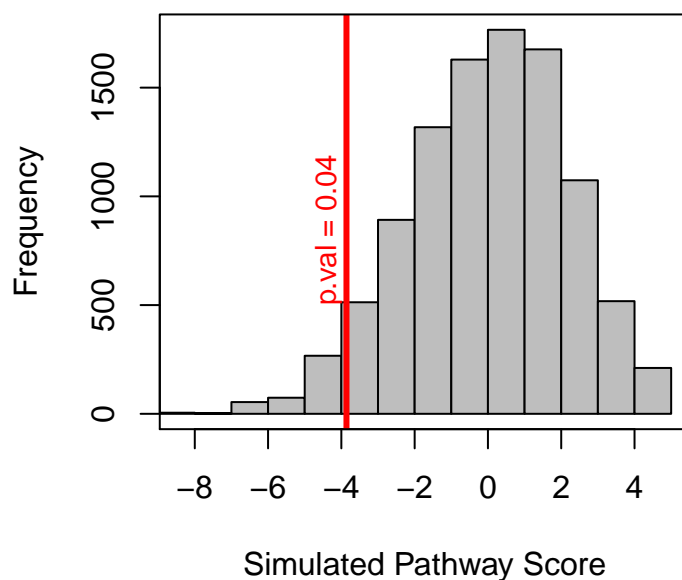


Figure 9: NF-kappa B signalling pathway histogram. Permutation based Gene set enrichment analysis creates a histogram of simulated pathway scores (in gray bars). The red line shows the observed pathway score in nonsurvivors when compared to survivor.

### 4.4 Relative gene expression of the targets of NF- $\kappa$ B

```
#####  
# drawing a histogram for NFkB targets  
#####  
source("Rcode/plotDensityNFkbTargets.R")  
box()
```

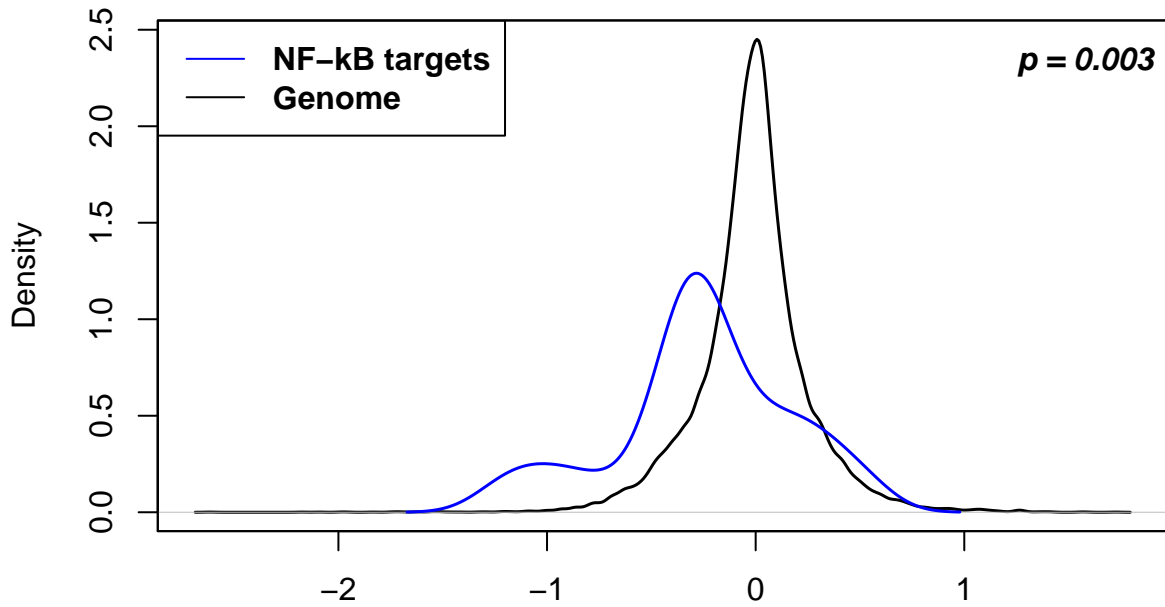


Figure 10: Relative gene expression of the targets of NFkB (i.e., Antigen processing and presentation genes and Immune receptor Genes). The gray peak in the background represents distribution of all genes in the genome. There is significant down-regulation of the targets in the non-survivors (blue line).

#### 4.5 M1 macrophage specific gene expression associated with survival

```
# check the expression of M1 vs M2 markers in NIBMG data
# # read the file containing gene IDs
#tg.gns.old <- scan("Metadata/targets_of_NFkB.txt", what="characters")
# M2 Genes
#####
sel.gns.dat <- read.table("metadata/M1_M2_markers.txt", sep="\t", header=T)
m1.gns <- sel.gns.dat[,1]
m1.gns <- intersect(m1.gns, featureNames(eset))
m2.gns <- sel.gns.dat[,2]
m2.gns <- intersect(m2.gns, featureNames(eset))

# function for plotting macrophage-specific gene expression
plotMgexp = function(type="M1", normalizeByControl=FALSE) {
  if(type=="M1") {
    egs = m1.gns
  } else {
    egs = m2.gns
  }

  # M1 gene expression
  gexp.s = rowMeans(exprs(eset[egs, eset$Outcome=="Surv" & eset$Group=="D1"]))
  gexp.ns = rowMeans(exprs(eset[egs, eset$Outcome=="Nonsurv" & eset$Group=="D1"]))

  if(normalizeByControl==TRUE) {
    gexp.c = rowMeans(exprs(eset[egs, which.ctrl]))
    gexp.s = gexp.s-gexp.c
    gexp.ns = gexp.ns-gexp.c
  }
}
```

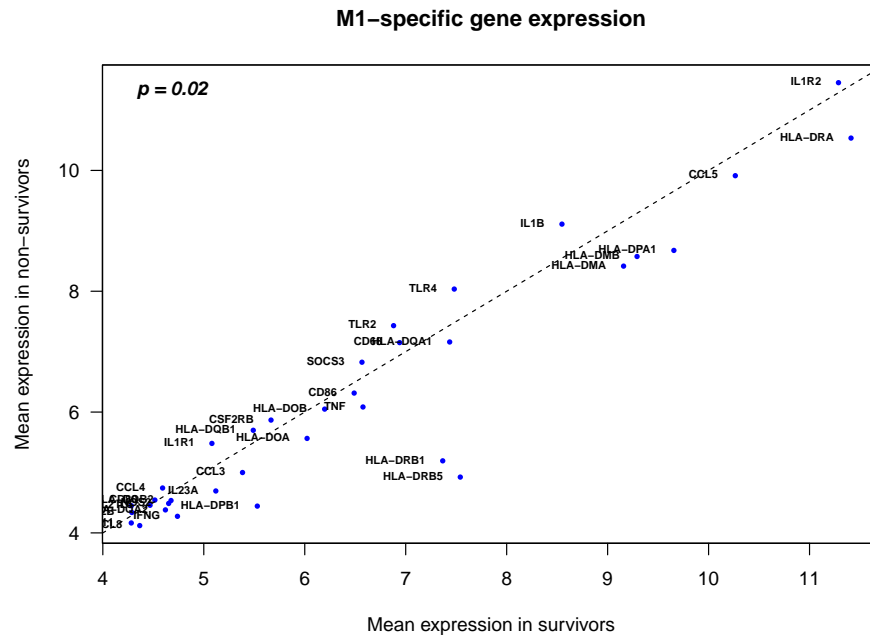


Figure 11: M1 macrophages (classically activated macrophages) are pro-inflammatory, important in host defence against the pathogens, phagocytosis, secretion of pro-inflammatory cytokines and microbicidal molecules. M2 macrophages (alternatively activated macrophages) participate in regulation of resolution of inflammation and repair of damaged tissues. M1-specific gene expression is significantly ( $p = 0.02$ ) associated with survival from sepsis in our data.

```

}

plot(x = gexp.s, y=gexp.ns, las=1, cex=0.62, col= "blue", pch=16,
     ylab = "Mean expression in non-survivors", xlab = "Mean expression in survivors",
     main=paste0(type, "-specific gene expression"))
abline(0,1, lty=2, xlim= c(-3, 10), ylim=c(-3, 10))
gsyms <- as.character(unlist(mget(m1.gns, org.Hs.egSYMBOL)))
#gsyms[intersect(which(gexp.1[,1] < -1), which(gexp.1[,2] < -1) )] <- ""
text(x = gexp.s, y=gexp.ns, labels=gsyms, cex= 0.58, pos=2, offset = 0.75, font=2)

pval <- t.test(gexp.s, gexp.ns, paired = T)$p.value
legend.str <- paste("p = ", formatC(pval, digits=1), sep="")
legend("topleft", legend.str, bty="n", text.font=4)
}
plotMgexp("M1")
box()

```

## 4.6 Down-regulation of Antigen processing and presentation signalling pathway genes in Non-survivors

```
#####
# Survivor versus non-survivor: Visual Search for interesting pathways
#####
getPval("hsa04612", drawPlot=TRUE, getSigGenes=TRUE) # APC

## [1] "HLA-DPA1" "HLA-DPB1" "HLA-DRB1" "HLA-DRB5" "HSPA5"
## [6] "KIR2DL3" "KIR3DL2" "CIITA" "PSME1" "TAP1"
## [11] "TAP2" "CD74"
```

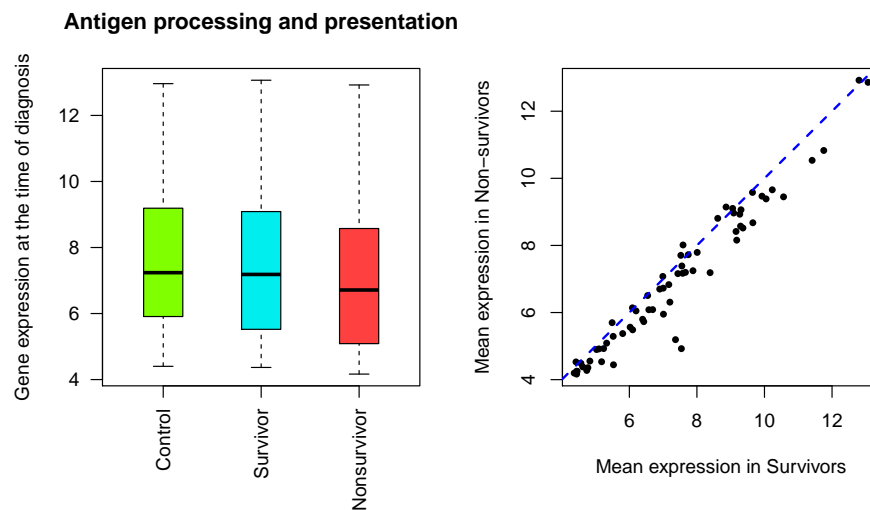


Figure 12: Status of APC pathway in NIBMG data with Box/Scatterplot

```
drawPermutHist("hsa04612", eset=eset.s, fac=factor(as.character(eset.s$Outcome))) # APC

##      p.val      zobs
## 0.015400 -7.349244

box()
```

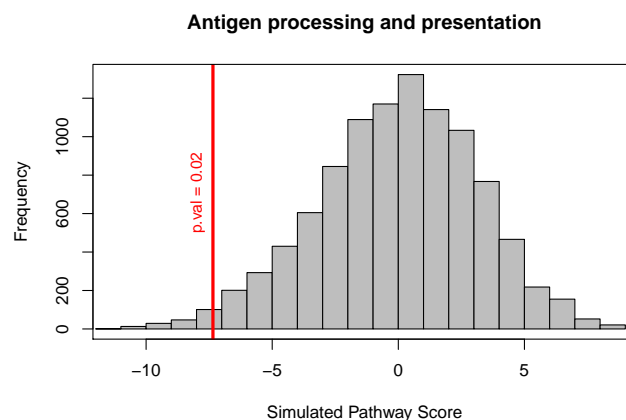


Figure 13: Status of APC pathway in NIBMG data, permutation based Geneset enrichment histogram.



## 4.7 Down-regulation of T cell receptor signalling pathway genes in Non-survivors

```
#####
# Survivor versus non-survivor: Visual Search for interesting pathways
#####
getPval("hsa04660", drawPlot=TRUE, getSigGenes=TRUE) # TCR

## [1] "RASGRP1" "ITK"      "LCK"      "NFKBIE"  "SOS2"
## [6] "BCL10"   "CD3D"     "CD3E"     "CD3G"

box()
```

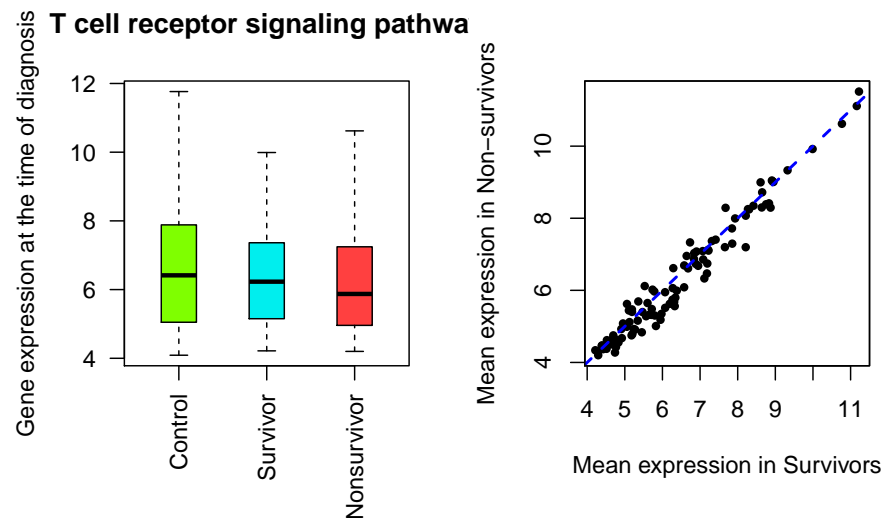


Figure 14: Status of TCR pathway in NIBMG data with Box/Scatterplot

```
drawPermutHist("hsa04660", eset=eset.s, fac=factor(as.character(eset.s$Outcome))) # TCR

##      p.val      zobs
## 0.114400 -2.670645

box()
```

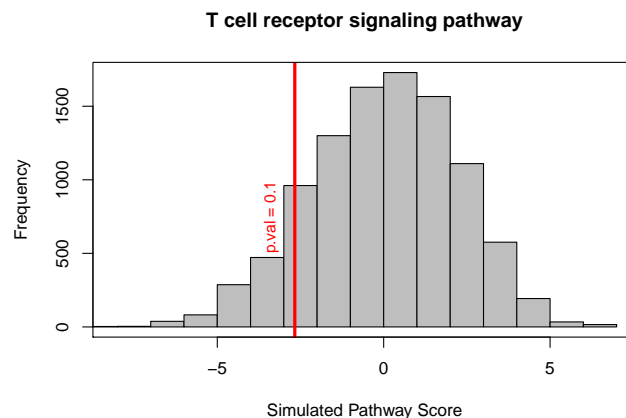


Figure 15: Status of TCR pathway in NIBMG data, permutation based Geneset enrichment histogram.

## 4.8 Key Biological Modules altered in Survivor

Key biological processes known to be associated with Sepsis (from literature) was identified as KEGG pathways. These pathways that represent a common biological processes were then checked by GSEA permutation scores whether they significantly alter ( $p < 0.1$ ) with the outcome. Selected significant pathway genes are then checked for monotonic change i.e. the absolute change (whether a gene is Up or Down) in gene expression goes in a same direction from control > Survivor > Non-survivor. Pathways were merged based on these monotonic genes that are common between them and connects each other. We constructed three modules (immunosuppression, coagulation and inflammation) based on the following criteria. First, we selected genes (nodes) from significantly perturbed pathways, that were also monotonically dysregulated from baseline in survivors and further in non-survivors. [We reasoned that the extent of perturbation of biological process shall increase with severity of disease, and therefore, the genes that are monotonically changing are key nodes of outcome-associated modules.] Then we extended the network of genes based on the connectivity described in literature. Wherever possible, we made the connectivity parsimonious with minimal number of non-significant genes (included only to preserve the continuity of signal flow, in keeping with the directed nature of KEGG graph which is experimentally curated). Three modules were constructed to describe three aspects of sepsis pathophysiology – inflammation, coagulation and immunosuppression. Each module was then considered an independent gene set and tested for significance of transcriptional difference between non-survivors and survivors by permutation-based testing of pathway score.

```
net.dat <- read.table("metadata//network_input.csv", sep="\t", header=T)
net.list = with(net.dat, split(x=Egid, f=Process))

par(mfrow=c(3, 1), mar=c(2, 9, 2, 9))
for(i in 1:3) {
  drawPermutHist(geneids=net.list[[i]],
                 titlestr=names(net.list)[i],
                 fac=eset.s$Outcome,
                 eset=eset.s)
  box()
}
```

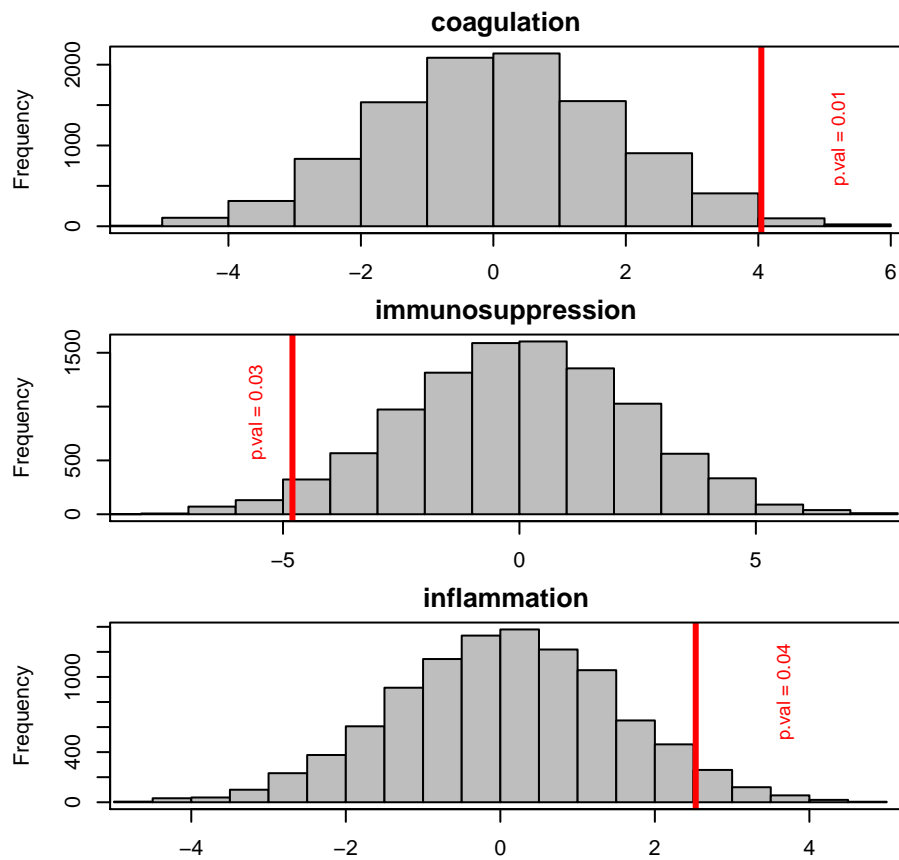


Figure 16: Three Biological processes found to be differentially enriched in Nonsurvivors.

## 5 Analyzing the heterogeneity of NIBMG sepsis cases

```
# get scores for the three modules from NIBMG data
fn = "Results/nibmgModuleScores.rda"
if(file.exists(fn)) {
  cat("Reading NIBMG module scores from file ...")
  load(fn)
  cat(" done!\n")
} else {
  sids = sampleNames(eset)[-c(which.ctrl)]
  ids.con = sampleNames(eset)[which.ctrl]
  modScores = sapply(names(net.list), function(id) {
    egs = as.character(net.list[[id]])
    sapply(sids, function(sid) {
      eset.curr = eset[,c(sid, ids.con)]
      eset.curr$Group = factor(as.character(eset.curr$Group))
      rtt = rowttests(eset.curr, "Group")
      rttstats = rtt[egs, "statistic"]
      z = sum(rttstats)/sqrt(length(egs))
      return(z)
    })
  })
  save(modScores, file=fn)
}

## Reading NIBMG module scores from file ... done!

# format the data
sids = rownames(modScores)
sids.df = do.call(rbind, strsplit(sids, split="D"))
colnames(sids.df) = c("Pt", "Day")
rm(sids)
dat = data.frame(sids.df, sids=rownames(modScores), modScores)

dat.split = with(dat, split(sids, Pt))
dat.split = sapply(dat.split, as.character)

par(mar=c(3,1,2,1))
dat = sapply(dat.split, function(x)
  as.matrix(dat[unlist(x),
    c("coagulation", "immunosuppression", "inflammation")]))
dat1 = dat
for(i in 1:length(dat)) {dat1[[i]] = rbind(dat[[i]], c(0,0,0))}
o = order(sapply(dat, function(x) min(x)))
dat1 = dat1[o]
plotdat = t(do.call(rbind, dat1))
mycols = c("blue", "darkgreen", "red")
b = barplot(plotdat, width=1, ylim=c(-15, 15), beside = T,
  col=mycols,
  border=mycols, las=2, cex.names=0.9)
legend("top", c("Coagulation", "Immunosuppression", "Inflammation"), horiz=T,
  col=mycols,
  border=mycols,
  pch=15, pt.cex= 1.7, bty="n", inset=c(-0.05, 0))
```

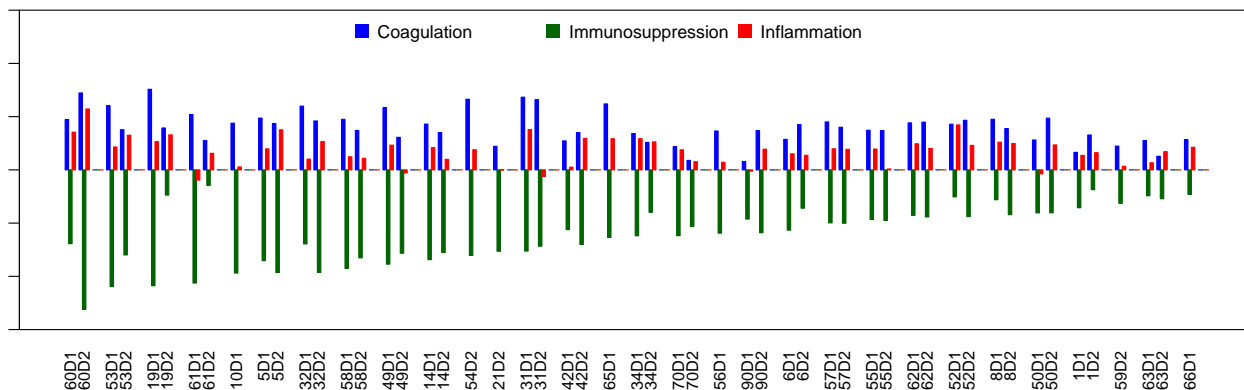


Figure 17: Barplot showing the score of 3 key modules in each sepsis patients

```
box()
```

```
# runEndo.R
# Endotype-specific analysis
# CIA      Sweeny2018    Check for Gene group expression
# changing in the predicted direction
# MARS2     Scicluna2017 Check gene expression ratio for each MARS category
# MARS140   Scicluna2017 Check gene expression correlation with given
# MARS category-mean expression; PCA
# PED       Wong2009     PCA
# SRSDE     Davenport2016 PCA
# SRS7      Davenport2016 PCA
# SRSF6     Burnham2017  PCA

# if endogenes object is absent, get it
if(!exists("endogenes")) {
  source("Rcode/getEndogenes.R")
}
```

```
## Loading endotype-specific gene lists from file ... done!
```

```
# Should the gene expression data be normalized to
# mean control group expression
normalize2con = TRUE
# get the gene expression data for the patients
which.case = c(which.d1, which.d2)
sids = sampleNames(eset[,which.case])
```

```
meanCon = rowMeans(exprs(eset[,which.ctrl]))
gexp = exprs(eset[,sids])
if(normalize2con==TRUE) {
  gexp = gexp-meanCon
}
```

```
#####
# CIA    Check with specific genes direction; derived from Sweeny2018
#####
# ciaGeneScore: a cia internal function for getting gene score
# (based on t-test and log-fold change)
```

```

# input: gene id (eg), patient id (id)
# output: gene score scalar (+1, -1 or 0)
ciaGeneScore = function(eg, id) {
  which.id = which(sampleNames(eset)==id)
  eset.sel = eset[, c(which.ctrl, which.id)]
  rtt = rowttests(eset.sel[eg, ], "Group")
  pval = rtt[eg, "p.value"]
  gscore = 0
  lfc = rtt[eg, "dm"]
  gscore = sign(lfc)
  return(gscore)
}

# ciaDirection2integer: a cia internal function for converting
# gene direction to signed integer
# input: cia category (group)
# output: vector of integers (+1 or -1)
ciaDirection2integer = function(group) {
  attach(endogenes)
  sel = which(cia$Category==group)
  egs = cia$EntrezID[sel]
  result = c(-1,1)[factor(cia$Direction[sel])]
  names(result) = egs
  detach(endogenes)
  return(result)
}

# ciaGroupScore: a cia internal function
# input: cia category id (ciagroup), patient id (id)
ciaGroupScore = function(ciagroup, id) {
  delta = ciaDirection2integer(ciagroup)
  egs = names(delta)
  scores = sapply(egs, ciaGeneScore, id)
  grpscore = sum(scores==delta)*100/length(delta)
  return(grpscore)
}

# generate the patient-specific category scores
nibmg.cia = t(sapply(sort(sids), function(id) {
  sapply(as.character(unique(endogenes$cia$Category)), function(ciagroup) {
    ciaGroupScore(ciagroup, id)
  })
})))

# make binary result for each sample
nibmg.cia.binary = t(apply(nibmg.cia, 1, function(x) {
  res = rep(0, length(x))
  sel = which.max(x)
  res[sel] = 1
  res
})))
colnames(nibmg.cia.binary) = colnames(nibmg.cia)

```

```

nibmg.cia = nibmg.cia.binary

#####
# MARS2 Check ratio
#####
# mars2scores: a mars2 internal function for getting mars2 scores for a given patient id
# input: patient id (id)
# output: a vector of 4 values for 4 MARS categories
mars2scores = function(id) {
  attach(endogenes)
  mnames = paste0("Mars", 1:4)
  mscores = rep(NA, 4)
  names(mscores) = mnames
  for(i in 1:length(mscores)) {
    numr.eg = as.character(mars2$EntrezID[mars2$Category== mnames[i]
                                         & mars2$Math=="Numerator"])
    denr.eg = as.character(mars2$EntrezID[mars2$Category== mnames[i]
                                         & mars2$Math=="Denominator"])
    mscores[i] = 2^(gexp[numr.eg,id]-gexp[denr.eg,id])
  }
  detach(endogenes)
  return(mscores)
}

ratio.mars2 = t(sapply(sort(sids), mars2scores))

nibmg.mars2 = t(apply(ratio.mars2, 1, function(x){
  sel = which.max(x)
  y = rep(0, length(x))
  y[sel] = 1
  return(y)
}))
colnames(nibmg.mars2) = colnames(ratio.mars2)
colnames(nibmg.mars2) = paste0(colnames(nibmg.mars2), ".2gr")

#scan()
#####
# MARS140 check for correlation with our data
#####
# get data for endotype-associated genes; intersect with NIBMG expression set
dat.mars140 = endogenes[["mars140"]]
egs.mars140 = dat.mars140$EntrezID
rownames(dat.mars140) = egs.mars140

egs.mars140 = intersect(egs.mars140, featureNames(eset))
dat.mars140 = dat.mars140[egs.mars140,]

gexp.mars140 = gexp[egs.mars140,]
r.mars140 = t(apply(gexp.mars140, 2, function(x) {
  apply(dat.mars140[, paste0("Mars",1:4)], 2, function(y) {
    rx=cor.test(x, y)
    res=ifelse(rx$p.value<0.01, rx$estimate, 0)
    return(res)
  })
}))

```

```

}))
o = order(rownames(r.mars140))
r.mars140 = r.mars140[o,]

# find the ids without any category result; correlation was insignificant
# for each of the 4 MARS categories
noCategory = names(which(apply(r.mars140, 1, max)==0))

nibmg.mars140 = t(apply(r.mars140, 1, function(x){
  sel = which.max(x)
  y = rep(0, length(x))
  y[sel] = 1
  return(y)
}))
nibmg.mars140[noCategory,] = 0
colnames(nibmg.mars140) = colnames(r.mars140)

cols3 = rainbow(3)
plotdat = as.matrix(data.frame(nibmg.cia, nibmg.mars2, nibmg.mars140))
heatmap.2(plotdat, trace="none", mar=c(8,22), dendrogram = "none",
  ColSideColors = c(rep(cols3[1],3),rep(cols3[2],4),rep(cols3[3],4)),
  Rowv = F, Colv = F, key=FALSE, keysize=0.7,
  col=c("white","darkblue"),
  rowsep=0:nrow(plotdat), colsep=0:ncol(plotdat), cexRow = 0.7,
  sepcolor="black", sepwidth=c(.01,.01))
legend(x=0.68, y=1, legend=c("CIA","MARS with 2-gene ratio","MARS with 140 genes"),
  col=cols3, bty="n", pch=15, horiz=F)
box()

```



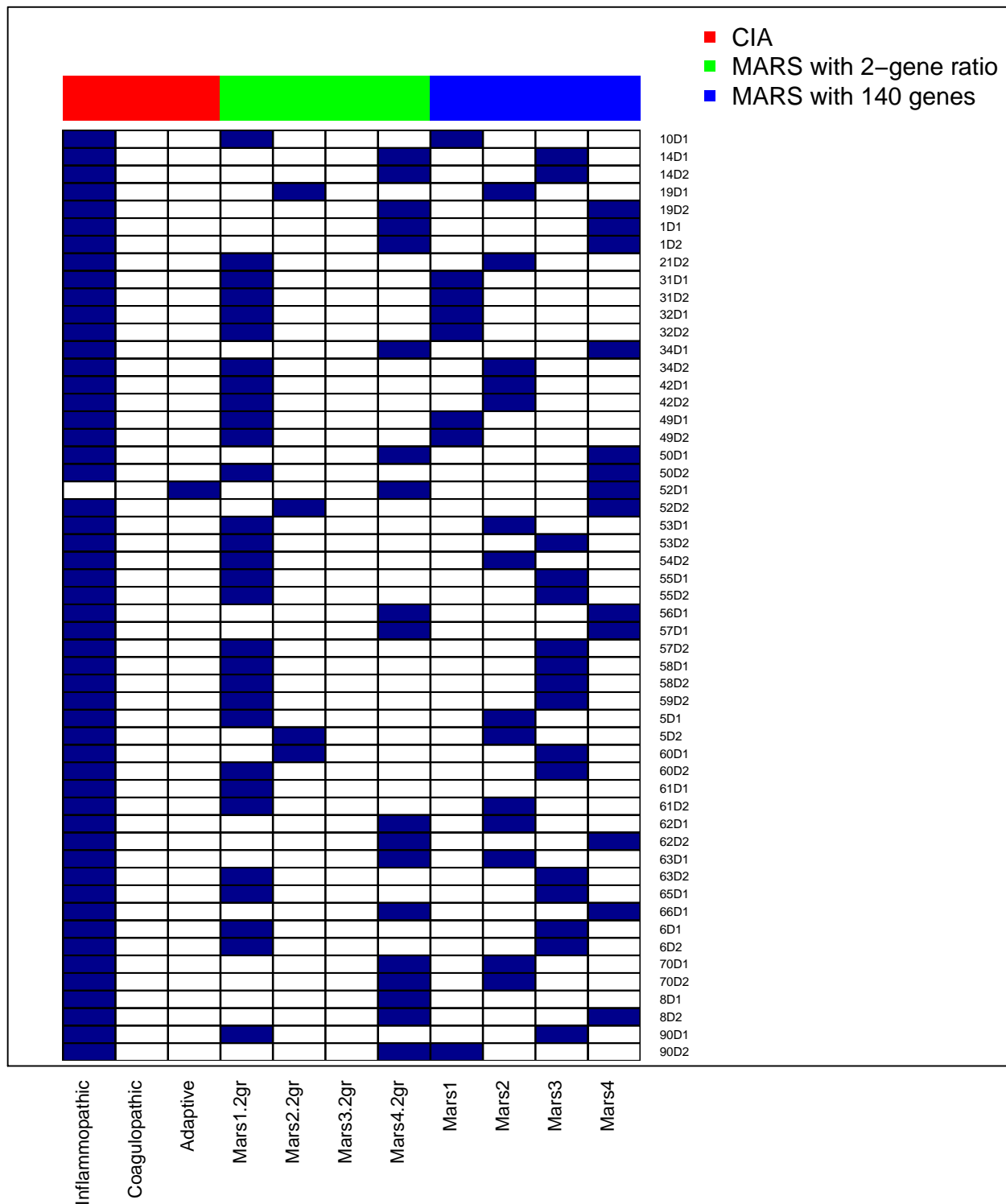


Figure 18: Heterogeneity in sepsis – endotypes :We applied 3 of the known classifiers (others could not be applied due to practical issues) on our samples but did not find a consensus among the results. We did observe that CIA (Sweeny) labeled all our samples (except one) in one category. MARS (SciCluna) classification led to more diverse labeling.