Supplementary mETHOD

# unsupervised consensus clustering

library(ConsensusClusterPlus)

#reading methylation or gene expression matrix

d<-read.table(file.choose(),sep="\t",header=T)

mads=apply(d,1,mad)

n<-length(d[,1])\*0.1

d=d[rev(order(mads))[1:n],]

d = sweep(d,1, apply(d,1,median,na.rm=T))

#the clustering algorithm can be set by a user-specified clustering algorithm

results=ConsensusClusterPlus(d,maxK=6,reps=1000,pItem=0.9,pFeature=0.9,title="SKCM",clusterAlg="km",distance=" euclidean ",seed=1262118388.71279,plot="png")

icl = calcICL(results,title="SKCM",plot="png")

write.table(results[[3]][["consensusClass"]],"k3\_sampleClassfication",sep="\t",col.names=FALSE)

write.table(results[[4]][["consensusClass"]],"k4\_sampleClassfication",sep="\t",col.names=FALSE)

# integrative clustering analysis

library(mogsa)

#reading gene expression matrix

mRNA<-read.table(file.choose(),sep="\t",header=T)

#reading miRNA expression matrix

miRNA<-read.table(file.choose(),sep="\t",header=T)

#reading methylation matrix

Methy<-read.table(file.choose(),sep="\t",header=T)

Mylist<-list(mRNA,miRNA,Methy)

moa <- mbpca(Mylist, ncomp = 20, k = "all", method = "globalScore", option = "lambda1",center=TRUE, scale=FALSE, moa = TRUE, svd.solver = "fast", maxiter = 1000)

r <- bootMbpca(moa, mc.cores = 2, B=100, replace = FALSE, resample = "sample")

moas <- mbpca(Mylist, ncomp = 6, k = 0.1, method = "globalScore", option = "lambda1",center=TRUE, scale=FALSE, moa = TRUE, svd.solver = "fast", maxiter = 1000)

scrs <- moaScore(moas)

gap <- moGap(moas, K.max = 12, cluster = "hcl")

hcl <- hclust(dist(scrs))

cls <- cutree(hcl, k=6)

write.table(cls," clusters ",sep="\t")

# differentially gene expression analysis

library( DESeq )

#reading read counts matrix

countsTable<-read.table(file.choose(),header=TRUE)

condition<-factor(c(rep("B",55),rep("A",301)))

cds <- newCountDataSet( countsTable, condition )

cds <- estimateSizeFactors( cds )

cds <- estimateDispersions( cds, sharingMode = "gene-est-only")

res <- nbinomTest( cds, "B", "A")

write.table(res," DeseqResult1.txt",sep="\t",row.names=FALSE)

countsTable<-read.table(file.choose(),header=TRUE)

condition<-factor(c(rep("B",55),rep("A",91)))

cds <- newCountDataSet( countsTable, condition )

cds <- estimateSizeFactors( cds )

cds <- estimateDispersions( cds, sharingMode = "gene-est-only")

res <- nbinomTest( cds, "B", "A")

write.table(res," DeseqResult2.txt ",sep="\t",row.names=FALSE)

countsTable<-read.table(file.choose(),header=TRUE)

condition<-factor(c(rep("B",301),rep("A",91)))

cds <- newCountDataSet( countsTable, condition )

cds <- estimateSizeFactors( cds )

cds <- estimateDispersions( cds, sharingMode = "gene-est-only")

res <- nbinomTest( cds, "B", "A")

write.table(res," DeseqResult3.txt ",sep="\t",row.names=FALSE)

#Immune scores and stromal scores analysis

library(estimate)

#reading GEO data gene expression matrix

skcm<-"Gene\_Exp\_Matrix"

filterCommonGenes(input.f=skcm,output.f="EntrezGene\_NormalizedExp\_Matrix.gct",id="EntrezID")

estimateScore("EntrezGene\_NormalizedExp\_Matrix.gct","estimate\_score.gct",platform="affymetrix")

# Differential methylation analysis

library(samr)

#reading methylation matrix

x1 = read.table(file.choose(),header=T,sep="\t")

x<- x1[1:length(x1[,1]),2:length(x1[1,])]

y <- c(rep(1,55), rep(2,301))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=as.character(x1[,1]),logged2=TRUE)

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

delta.table <- samr.compute.delta.table(samr.obj,min.foldchange=0.1)

del<- 0

siggenes.table<-samr.compute.siggenes.table(samr.obj,del,data,delta.table,all.genes=TRUE)

write.table(siggenes.table$genes.up," Up1\_Methy",sep="\t",row.names=FALSE)

write.table(siggenes.table$genes.lo," Down1\_Methy",sep="\t",row.names=FALSE)

x1 = read.table(file.choose(),header=T,sep="\t")

x<- x1[1:length(x1[,1]),2:length(x1[1,])]

y <- c(rep(1,55), rep(2,91))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=as.character(x1[,1]),logged2=TRUE)

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

delta.table <- samr.compute.delta.table(samr.obj,min.foldchange=0.1)

del<- 0

siggenes.table<-samr.compute.siggenes.table(samr.obj,del,data,delta.table,all.genes=TRUE)

write.table(siggenes.table$genes.up," Up2\_Methy",sep="\t",row.names=FALSE)

write.table(siggenes.table$genes.lo," Down2\_Methy",sep="\t",row.names=FALSE)

x1 = read.table(file.choose(),header=T,sep="\t")

x<-x1[1:length(x1[,1]),2:length(x1[1,])]

y <- c(rep(1,301), rep(2,91))

data=list(x=x,y=y, geneid=as.character(1:nrow(x)),genenames=as.character(x1[,1]),logged2=TRUE)

samr.obj<-samr(data, resp.type="Two class unpaired", nperms=100)

delta.table <- samr.compute.delta.table(samr.obj,min.foldchange=0.1)

del<- 0

siggenes.table<-samr.compute.siggenes.table(samr.obj,del,data,delta.table,all.genes=TRUE)

write.table(siggenes.table$genes.up," Up3\_Methy",sep="\t",row.names=FALSE)

write.table(siggenes.table$genes.lo," Down3\_Methy",sep="\t",row.names=FALSE)

#signature analysis

suppressPackageStartupMessages(library("deconstructSigs"))

#reading mutation data

sample.mut.ref<-read.table(file.choose(),sep="\t",header=T)

sigs.input <- mut.to.sigs.input(mut.ref = sample.mut.ref, sample.id = "Sample",

chr = "chr",pos = "pos",ref = "ref",alt = "alt")

sample\_better = whichSignatures(tumor.ref = sigs.input,signatures.ref = signatures.cosmic,

sample.id = "Better", contexts.needed = TRUE,tri.counts.method = 'exome')

sample\_poor = whichSignatures(tumor.ref = sigs.input,signatures.ref = signatures.cosmic,

sample.id = "Poor", contexts.needed = TRUE,tri.counts.method = 'exome')

sample\_intermediate = whichSignatures(tumor.ref = sigs.input,signatures.ref = signatures.cosmic,

sample.id = "Intermediate", contexts.needed = TRUE,tri.counts.method = 'exome')

plotSignatures(sample\_better)

plotSignatures(sample\_poor)

plotSignatures(sample\_intermediate)