#merge the datasets of 55235+77298+12021

library(tidyverse)

library(GEOquery)

library(tidyverse)

library(GEOquery)

library(limma)

library(affy)

library(stringr)

gset = getGEO('GSE55235', destdir=".", AnnotGPL = T, getGPL = T)

class(gset)

gset[[1]]

gset2 = getGEO('GSE12021', destdir=".", AnnotGPL = T, getGPL = T)

class(gset2)

gset2[[1]]

gset3 = getGEO('GSE77298', destdir=".", AnnotGPL = T, getGPL = T)

class(gset3)

gset3[[1]]

plf1<-gset[[1]]@annotation

plf2<-gset2[[1]]@annotation

plf3<-gset3[[1]]@annotation

GPL\_data<- getGEO(filename ="GPL96.annot.gz", AnnotGPL = T)

GPL\_data\_11 <- Table(GPL\_data)

GPL\_data1<- getGEO(filename ="GPL96.annot.gz", AnnotGPL = T)

GPL\_data\_22 <- Table(GPL\_data1)

GPL\_data2<- getGEO(filename ="GPL570.annot.gz", AnnotGPL = T)

GPL\_data\_33 <- Table(GPL\_data2)

raw\_geneid<-fData(gset[[1]])

raw\_geneid<-raw\_geneid[,c(1,3)]

raw\_geneid2<-fData(gset2[[1]])

raw\_geneid2<-raw\_geneid2[,c(1,3)]

raw\_geneid3<-fData(gset3[[1]])

raw\_geneid3<-raw\_geneid3[,c(1,3)]

exp <- exprs(gset[[1]])

probe\_name<-rownames(exp)

exp2 <- exprs(gset2[[1]])

probe\_name2<-rownames(exp2)

exp3 <- exprs(gset3[[1]])

probe\_name3<-rownames(exp3)

raw\_geneid$`Gene symbol`<-data.frame(sapply(raw\_geneid$`Gene symbol`,function(x)unlist(strsplit(x,"///"))[1]),stringsAsFactors=F)[,1]

raw\_geneid2$`Gene symbol`<-data.frame(sapply(raw\_geneid2$`Gene symbol`,function(x)unlist(strsplit(x,"///"))[1]),stringsAsFactors=F)[,1]

raw\_geneid3$`Gene symbol`<-data.frame(sapply(raw\_geneid3$`Gene symbol`,function(x)unlist(strsplit(x,"///"))[1]),stringsAsFactors=F)[,1]

loc<-match(GPL\_data\_11[,1],probe\_name)

probe\_exp<-exp[loc,]

raw\_geneid<-(as.matrix(raw\_geneid[,"Gene symbol"]))

index<-which(!is.na(raw\_geneid))

geneid<-raw\_geneid[index]

exp\_matrix<-probe\_exp[index,]

geneidfactor<-factor(geneid)

gene\_exp\_matrix<-apply(exp\_matrix,2,function(x) tapply(x,geneidfactor,mean))

rownames(gene\_exp\_matrix)<-levels(geneidfactor)

gene\_exp\_matrix<-gene\_exp\_matrix[,-(11:20)]

loc2<-match(GPL\_data\_22[,1],probe\_name2)

probe\_exp2<-exp2[loc2,]

loc2

raw\_geneid2<-(as.matrix(raw\_geneid2[,"Gene symbol"]))

index2<-which(!is.na(raw\_geneid2))

geneid2<-raw\_geneid2[index2]

exp\_matrix2<-probe\_exp2[index2,]

geneidfactor2<-factor(geneid2)

gene\_exp\_matrix2<-apply(exp\_matrix2,2,function(x) tapply(x,geneidfactor2,mean))

rownames(gene\_exp\_matrix2)<-levels(geneidfactor2)

gene\_exp\_matrix2<-gene\_exp\_matrix2[,-c(6,7,13,16:21,25)]

loc3<-match(GPL\_data\_33[,1],probe\_name3)

loc3

probe\_name3

probe\_exp3<-exp3[loc3,]

raw\_geneid3<-(as.matrix(raw\_geneid3[,"Gene symbol"]))

index3<-which(!is.na(raw\_geneid3))

geneid3<-raw\_geneid3[index3]

exp\_matrix3<-probe\_exp3[index3,]

geneidfactor3<-factor(geneid3)

gene\_exp\_matrix3<-apply(exp\_matrix3,2,function(x) tapply(x,geneidfactor3,mean))

rownames(gene\_exp\_matrix3)<-levels(geneidfactor3)

geo\_exp\_1=as.data.frame(gene\_exp\_matrix)

geo\_exp\_2=as.data.frame(gene\_exp\_matrix2)

geo\_exp\_3=as.data.frame(gene\_exp\_matrix3)

sameSample=intersect(rownames(geo\_exp\_1), rownames(geo\_exp\_2))

sameSample=as.data.frame(sameSample)

sameSample=intersect(rownames(geo\_exp\_3), sameSample$sameSample)

gene\_exp1=geo\_exp\_1[sameSample,,drop=F]

gene\_exp2=geo\_exp\_2[sameSample,,drop=F]

gene\_exp3=geo\_exp\_3[sameSample,,drop=F]

bindgeo=cbind(gene\_exp1,gene\_exp2,gene\_exp3)

pdata <- pData(gset[[1]])

pdata <- pdata[-(11:20),]

pdata2 <- pData(gset2[[1]])

pdata2<-pdata2[-c(6,7,13,16:21,25),]

pdata3 <- pData(gset3[[1]])

group\_list <- ifelse(str\_detect(pdata$title,"healthy joint"), "NC",

"RA")

group\_list

group\_list = factor(group\_list,

levels = c("NC","RA"))

group\_list

pdata$group=group\_list

group\_list2 <- ifelse(str\_detect(pdata2$title,"Normal"), "NC",

"RA")

group\_list2

group\_list2 = factor(group\_list2,

levels = c("NC","RA"))

group\_list2

pdata2$group=group\_list2

group\_list3 <- ifelse(str\_detect(pdata3$title,"HC"), "NC",

"RA")

group\_list3

group\_list3 = factor(group\_list3,

levels = c("NC","RA"))

group\_list3

pdata3$group=group\_list3

group1<-(as.matrix(pdata[,"group"]))

row.names(group1)=rownames(pdata)

colnames(group1)="group"

group2<-(as.matrix(pdata2[,"group"]))

row.names(group2)=rownames(pdata2)

colnames(group2)="group"

group3<-(as.matrix(pdata3[,"group"]))

row.names(group3)=rownames(pdata3)

colnames(group3)="group"

talgroup=as.data.frame(rbind(group1,group2,group3))

talgroup\_list=factor(talgroup$group,levels = c("RA","NC"))

write.csv(talgroup,file = "group.csv")

boxplot(bindgeo,outline=T, notch=T,col=talgroup\_list, las=2)

dev.off()

bindgeo\_normal=normalizeBetweenArrays(bindgeo)

boxplot(bindgeo\_normal,outline=T, notch=T,col=talgroup\_list, las=2)

range(bindgeo\_normal)

bindgeo\_normal <- log2(bindgeo\_normal+1)

bindgeo\_normal <-as.data.frame(bindgeo\_normal)

bindgeo\_normal=na.omit(bindgeo\_normal)

write.csv(bindgeo\_normal,file = "bindgeo\_exp.csv")

range(bindgeo\_normal)

dev.off()

#DEGs analysis of train set

expFile="bindgeo\_exp.csv" #?????ļ?

#??ȡ?????ļ????????????ļ?????

expr\_data=read.csv(expFile,sep=",",header=T,row.names = 1,check.names=F)

group<-read.csv("group.csv",header = T,row.names = 1,sep = ",")

design <- model.matrix(~0+factor(group$group))

colnames(design) <- levels(factor(group$group))

rownames(design) <- colnames(expr\_data)

contrast.matrix <- makeContrasts(RA-NC,levels = design)

fit <- lmFit(expr\_data,design)

fit2 <- contrasts.fit(fit, contrast.matrix)

fit2 <- eBayes(fit2)

DEG <- topTable(fit2, coef = 1,n = Inf,sort.by="logFC")

DEG <- na.omit(DEG)

DEG$regulate <- ifelse(DEG$adj.P.Val> 0.05, "unchanged",

ifelse(DEG$logFC > 1, "up-regulated",

ifelse(DEG$logFC < -1, "down-regulated", "unchanged")))

write.table(DEG,"DEG.csv",row.names=T,col.names=T,sep=",")

job <- "test"

write.table(table(DEG$regulate),file = paste0(job,"\_","DEG\_result\_1\_005.txt"),

sep = "\t",quote = F,row.names = T,col.names = T)

write.table(data.frame(gene\_symbol=rownames(DEG),DEG),file = paste0(job,"\_","DEG\_result.txt"),

sep = "\t",quote = F,row.names = F,col.names = T)

DE\_1\_0.05 <- DEG[DEG$adj.P.Val<0.05&abs(DEG$logFC)>1,]

upGene\_1\_0.05 <- DE\_1\_0.05[DE\_1\_0.05$regulate == "up-regulated",]

downGene\_1\_0.05 <- DE\_1\_0.05[DE\_1\_0.05$regulate == "down-regulated",]

write.csv(upGene\_1\_0.05,paste0(job,"\_","upGene\_1\_005.csv"))

write.csv(downGene\_1\_0.05,paste0(job,"\_","downGene\_1\_005.csv"))

#volcano plot of DEGs

library(dplyr) #

library(gt)

data<-read.csv("DEG.csv", header = TRUE, sep = ",", dec = ".", quote = "\"", fill = TRUE, comment.char = "")

data<- data%>%

mutate(expression = case\_when(logFC>= 1 &adj.P.Val< 0.05 ~ "up",

logFC<= -1 &adj.P.Val< 0.05 ~ "down",

TRUE ~ "Unchanged"))

data<-na.omit(data)

library(ggplot2)

library(ggrepel)

volc\_plot<- ggplot(data, aes(logFC, -log10(P.Value))) +xlim(-9,9)+ylim(0,18)+# 将p值进行-log10转化

geom\_hline(yintercept = -log10(0.05), linetype = "dashed", color = "#999999")+

geom\_vline(xintercept = c(-1.2,1.2), linetype = "dashed", color = "#999999")+

geom\_point(aes(color = expression),

size =2.5,

alpha =0.5) +

theme\_bw(base\_size = 12)+

ggsci::scale\_color\_jama() +

theme(panel.grid = element\_blank(),

legend.position = 'right')

volc\_plot

top\_20 <-bind\_rows(

data%>%

filter(expression=='up') %>%

arrange(P.Value, desc(abs(logFC))) %>%

head(10),

data%>%

filter(expression=='down') %>%

arrange(P.Value, desc(abs(logFC))) %>%

head(10)

)

write.table(top\_20,file="top\_20.txt",sep="\t",quote=F,col.names=T)

options(ggrepel.max.overlaps = Inf)

top\_20 %>% gt()

volc\_plot2 <- volc\_plot+

geom\_label\_repel(data = top\_20,

aes(logFC, -log10(P.Value), label = gene),

size = 2.5)

volc\_plot2

ggsave(filename="volplot20.png",

width=10,

height=8,

units="in",

dpi=300)

#heatmap of top 100 DEGs

library(ggplot2)

library(pheatmap)

data<-read.csv("DEG.csv", header = TRUE, sep = ",", dec = ".", quote = "\"", fill = TRUE, comment.char = "")

#data<-read.csv("GEO\_diff.csv", header = TRUE, sep = ",", dec = ".", quote = "\"", fill = TRUE, comment.char = "")

data<- data%>%

mutate(expression = case\_when(logFC>= 1 &adj.P.Val< 0.05 ~ "up",

logFC<= -1 &adj.P.Val< 0.05 ~ "down",

TRUE ~ "Unchanged"))

top\_100 <-bind\_rows(

data%>%

filter(expression=='up') %>%

arrange(P.Value, desc(abs(logFC))) %>%

head(50),

data%>%

filter(expression=='down') %>%

arrange(P.Value, desc(abs(logFC))) %>%

head(50)

)

write.table(top\_100,file="top\_100.csv",sep=",",quote=F,col.names=T,row.names = T)

data=read.csv("top\_100.csv",header=TRUE,row.names=1,check.names = FALSE)

expr\_data<-read.csv("bindgeo\_exp.csv", header = TRUE, sep = ",", dec = ".", quote = "\"", fill = TRUE, comment.char = "",row.names =1)

colnames(data)

rownames(data)

DEG\_gene\_expr100 <- expr\_data[rownames(data),]

pheatmap(DEG\_gene\_expr100)

group <-read.csv(file="group.csv",sep=",",header=TRUE,row.names=1,check.names=FALSE)

p <- pheatmap(DEG\_gene\_expr100,scale="row",

border="white",

cluster\_cols = T,

cluster\_rows = F,

show\_rownames = T,

show\_colnames = F,

annotation\_col = group)

p

dev.off()