**焦亡评分**

library(GSVA)

library(limma)

library(GSEABase)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\差异分析\\GSE7084A")#设置工作目录

inputFile="geneMatrix.txt" #输入文件

gmtFile="PRG.gmt" #GMT文件

#读取输入文件，并对输入文件处理

rt=read.table(inputFile,sep="\t",header=T,check.names=F)

rt=as.matrix(rt)

rownames(rt)=rt[,1]

exp=rt[,2:ncol(rt)]

dimnames=list(rownames(exp),colnames(exp))

mat=matrix(as.numeric(as.matrix(exp)),nrow=nrow(exp),dimnames=dimnames)

mat=avereps(mat)

mat=mat[rowMeans(mat)>0,]

geneSet=getGmt(gmtFile,

geneIdType=SymbolIdentifier())

#ssgsea分析

ssgseaScore=gsva(mat, geneSet, method='ssgsea', kcdf='Gaussian', abs.ranking=TRUE)

#定义ssGSEA score矫正函数

normalize=function(x){

return((x-min(x))/(max(x)-min(x)))}

#对ssGSEA score进行矫正

ssgseaOut=normalize(ssgseaScore)

ssgseaOut=rbind(id=colnames(ssgseaOut),ssgseaOut)

write.table(ssgseaOut,file="GSEA.txt",sep="\t",quote=F,col.names=F)

**差异分析**

library(ggpubr)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\差异分析\\GSE7084A") #设置工作目录

data=read.table("GSEAx.txt", header=T, sep="\t", check.names=F, row.names=1)

#设置比较组

data$group=factor(data$group, levels=c("Control Abdominal Aorta", "Abdominal Aortic Aneurysm"))

group=levels(factor(data$group))

comp=combn(group, 2)

my\_comparisons=list()

for(i in 1:ncol(comp)){my\_comparisons[[i]]<-comp[,i]}

#对免疫治疗打分进行循环,分别绘制小提琴图

for(i in colnames(data)[1:(ncol(data)-1)]){

rt=data[,c(i, "group")]

colnames(rt)=c("IPS", "group")

gg1=ggboxplot(rt, x="group", y="IPS", fill = "group",

xlab="", ylab=i,

legend.title="",

palette=c("#2E8B57", "#D2691E"), width = .3) + rotate\_x\_text(20)+

stat\_compare\_means(comparisons = my\_comparisons,symnum.args=list(cutpoints = c(0, 0.001, 0.01, 0.05, 1), symbols = c("\*\*\*", "\*\*", "\*", "ns")),label = "p.signif")

pdf(file=paste0(i, ".pdf"), width=5, height=6)

print(gg1)

dev.off()}

library(ggpubr) #引用包

tciaFile="GSEAx.txt" #免疫治疗打分文件

scoreFile="Group.txt" #m6A打分分组文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\差异分析\\GSE57691")

#读取免疫治疗打分文件

ips=read.table(tciaFile, header=T, sep="\t", check.names=F, row.names=1)

#读取m6A打分分组文件

score=read.table(scoreFile, header=T, sep="\t", check.names=F, row.names=1)

#合并数据

sameSample=intersect(row.names(ips), row.names(score))

ips=ips[sameSample, , drop=F]

score=score[sameSample, "Risk", drop=F]

data=cbind(ips, score)

#设置比较组

data$group=factor(data$Risk, levels=c("Control","Abdominal Aortic Aneurysm"))

group=levels(factor(data$group))

comp=combn(group, 2)

my\_comparisons=list()

for(i in 1:ncol(comp)){my\_comparisons[[i]]<-comp[,i]}

#对免疫治疗打分进行循环,分别绘制小提琴图

for(i in colnames(data)[1:(ncol(data)-1)]){

rt=data[,c(i, "group")]

colnames(rt)=c("IPS", "group")

gg1=ggboxplot(rt, x="group", y="IPS", fill = "group",

xlab="", ylab=i,

legend.title="",

palette=c("#2E8B57", "#D2691E"), width = .3) + rotate\_x\_text(20)+

stat\_compare\_means(comparisons = my\_comparisons,symnum.args=list(cutpoints = c(0, 0.001, 0.01, 0.05, 1), symbols = c("\*\*\*", "\*\*", "\*", "ns")),label = "p.signif")

pdf(file=paste0(i, ".pdf"), width=5, height=6)

print(gg1)

dev.off()}

**数据转换**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\GSE7084")

lncRNA<-read.table("geneMatrix-GPL2507.txt",header=T,sep="\t",row.names = 1,check.names = F,stringsAsFactors = F)

lncRNAEXP=log2(lncRNA[,1:ncol(lncRNA)]+1)

lncRNA=cbind(lncRNAEXP)

write.table(cbind(id=rownames(cbind(lncRNA[,1:15])),cbind(lncRNA[,1:15])),"geneMatrix-GPL2507R.txt",sep="\t",quote=F,row.names=F)

**数据合并**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\GSE7084")

inputfile1="geneMatrix-GPL570.txt" #生存时间数据

inputfile2="geneMatrix-GPL2507R.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="geneNames")

write.table(merger\_data,"geneMatrix.txt",sep = "\t",row.names = F,quote = F)

**数据合并**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\差异分析")

inputfile1="geneMatrix.txt" #生存时间数据

inputfile2="PRG.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"PRGExp.txt",sep = "\t",row.names = F,quote = F)

**差异分析**

library(limma)

library(reshape2)

library(ggpubr)

riskFile="Group.txt" #分组

immFile="PRGEx.txt" #基因表达结果文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\差异分析\\GSE7084A")

#设置工作目录

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

#读取风险文件

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

rt=cbind(data[sameSample,,drop=F], risk[sameSample,"Risk",drop=F])

rt=rt[order(rt$Risk, decreasing=T),]

conNum=nrow(rt[rt$Risk=="Control Abdominal Aorta",])

treatNum=nrow(rt[rt$Risk=="Abdominal Aortic Aneurysm",])

##################绘制箱线图##################

#把数据转换成ggplot2输入文件

data=rt

data=melt(data, id.vars=c("Risk"))

colnames(data)=c("Risk", "Immune", "Expression")

#绘制箱线图

group=levels(factor(data$Risk))

data$Risk=factor(data$Risk, levels=c("Control Abdominal Aorta","Abdominal Aortic Aneurysm"))

bioCol=c("#0066FF","#FF0000","#6E568C","#7CC767","#223D6C","#D20A13","#FFD121","#088247","#11AA4D")

bioCol=bioCol[1:length(group)]

boxplot=ggboxplot(data, x="Immune", y="Expression", fill="Risk",

xlab="",

ylab="Relative Expression",

legend.title="Type",

width=0.6,

palette=bioCol)+

rotate\_x\_text(50)+

stat\_compare\_means(aes(group=Risk),symnum.args=list(cutpoints=c(0, 0.001, 0.01, 0.05, 1), symbols=c("\*\*\*", "\*\*", "\*", "ns")), label="p.signif")

#输出图片

pdf(file="PRG.pdf", width=18, height=7)

print(boxplot)

dev.off()

library(limma)

library(reshape2)

library(ggpubr)

riskFile="Group.txt" #分组

immFile="PRGEx.txt" #基因表达结果文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\差异分析\\GSE57691")

#设置工作目录

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

#读取风险文件

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

rt=cbind(data[sameSample,,drop=F], risk[sameSample,"Risk",drop=F])

rt=rt[order(rt$Risk, decreasing=T),]

conNum=nrow(rt[rt$Risk=="Control",])

treatNum=nrow(rt[rt$Risk=="Abdominal Aortic Aneurysm",])

##################绘制箱线图##################

#把数据转换成ggplot2输入文件

data=rt

data=melt(data, id.vars=c("Risk"))

colnames(data)=c("Risk", "Immune", "Expression")

#绘制箱线图

group=levels(factor(data$Risk))

data$Risk=factor(data$Risk, levels=c("Control","Abdominal Aortic Aneurysm"))

bioCol=c("#0066FF","#FF0000","#6E568C","#7CC767","#223D6C","#D20A13","#FFD121","#088247","#11AA4D")

bioCol=bioCol[1:length(group)]

boxplot=ggboxplot(data, x="Immune", y="Expression", fill="Risk",

xlab="",

ylab="Relative Expression",

legend.title="Type",

width=0.6,

palette=bioCol)+

rotate\_x\_text(50)+

stat\_compare\_means(aes(group=Risk),symnum.args=list(cutpoints=c(0, 0.001, 0.01, 0.05, 1), symbols=c("\*\*\*", "\*\*", "\*", "ns")), label="p.signif")

#输出图片

pdf(file="PRG.pdf", width=18, height=7)

print(boxplot)

dev.off()

**PCA分析**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\PCA")

inputfile1="Group.txt" #生存时间数据

inputfile2="PRGEx.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"PCA.txt",sep = "\t",row.names = F,quote = F)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\PCA")

data=read.table("PCA.txt",header=T,sep="\t",row.names=1) #读取表格

data=as.matrix(data) #矩阵转置

data.class <- rownames(data)

data.pca <- prcomp(data, scale. = TRUE) #PCA分析

write.table(predict(data.pca),file="NewTab.xls",quote=F,sep="\t") #输出新表

#pca 2d plot

library(ggplot2)

group=c(rep("Healthy Control",12),rep("Sepsis",111)) #对照组和实验组的样品数目

pcaPredict=predict(data.pca)

PCA = data.frame(PCA1 = pcaPredict[,1], PCA2 = pcaPredict[,2],group=group)

PCA.mean=aggregate(PCA[,1:2],list(group=PCA$group),mean)

#自定义函数

veganCovEllipse<-function (cov, center = c(0, 0), scale = 1, npoints = 100) {

theta <- (0:npoints) \* 2 \* pi/npoints

Circle <- cbind(cos(theta), sin(theta))

t(center + scale \* t(Circle %\*% chol(cov)))

}

df\_ell <- data.frame()

for(g in levels(PCA$group)){

df\_ell <- rbind(df\_ell, cbind(as.data.frame(with(PCA[PCA$group==g,],

veganCovEllipse(cov.wt(cbind(PCA1,PCA2),

wt=rep(1/length(PCA1),length(PCA1)))$cov,

center=c(mean(PCA1),mean(PCA2))))),group=g))

}

pdf(file="PCA.pdf",height=5,width=6)

ggplot(data = PCA, aes(PCA1, PCA2)) + geom\_point(aes(color = group)) +

geom\_path(data=df\_ell, aes(x=PCA1, y=PCA2,colour=group), size=1, linetype=2)+

annotate("text",x=PCA.mean$PCA1,y=PCA.mean$PCA2,label=PCA.mean$group)+

theme\_bw()+

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

dev.off()

**MicroRNA差异分析**

library("limma")

logFoldChange=1

adjustP=0.05

library(limma)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\MicroRNA差异分析\\GSE144431")

rt=read.table("geneMatrix.txt",sep="\t",header=T,check.names=F)

rt=as.matrix(rt)

rownames(rt)=rt[,1]

exp=rt[,2:ncol(rt)]

dimnames=list(rownames(exp),colnames(exp))

rt=matrix(as.numeric(as.matrix(exp)),nrow=nrow(exp),dimnames=dimnames)

#differential

modType=c(rep("normal",4),rep("tumor",4))

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

colnames(design) <- c("con","treat")

fit <- lmFit(rt,design)

cont.matrix<-makeContrasts(treat-con,levels=design)

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

fit2 <- eBayes(fit2)

allDiff=topTable(fit2,adjust='fdr',number=200000)

write.table(allDiff,file="limmaTab.xls",sep="\t",quote=F)

#write table

diffSig <- allDiff[with(allDiff, (abs(logFC)>logFoldChange & adj.P.Val < adjustP )), ]

write.table(diffSig,file="diff.xls",sep="\t",quote=F)

diffUp <- allDiff[with(allDiff, (logFC>logFoldChange & adj.P.Val < adjustP )), ]

write.table(diffUp,file="up.xls",sep="\t",quote=F)

diffDown <- allDiff[with(allDiff, (logFC<(-logFoldChange) & adj.P.Val < adjustP )), ]

write.table(diffDown,file="down.xls",sep="\t",quote=F)

#write expression level of diff gene

hmExp=rt[as.vector(diffSig[,1]),]

diffExp=rbind(id=colnames(hmExp),hmExp)

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

**数据合并**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\MicroRNA差异分析")

inputfile1="GSE62179Diff.txt" #生存时间数据

inputfile2="GSE63541Diff.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"MiRNA.txt",sep = "\t",row.names = F,quote = F)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\MicroRNA差异分析\\精准数据")

inputfile1="PRG.txt" #生存时间数据

inputfile2="Target.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"RNA.txt",sep = "\t",row.names = F,quote = F)

**RF**

set.seed(123)

library(randomForest) #random forests

library(caret) #

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\机器学习\\GSE57691")

data=read.table("PRGEx.txt",sep="\t",header=T,check.names=F, row.names = 1)

rf.pros <- randomForest(Status ~ ., data = data)

rf.pros

plot(rf.pros)

which.min(rf.pros$mse)

rf.pros.2 <- randomForest(Status~ ., data =data, ntree = 230)

rf.pros.2

varImpPlot(rf.pros.2, scale = TRUE,

main = "Variable Importance Plot - IRG")

importance(rf.pros.2)

**LASSO**

library(glmnet)

set.seed(123)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\机器学习\\GSE57691")

#设置工作目录

lncRNA<-read.table("PRGEx.txt",header=T,sep="\t",row.names = 1,check.names = F,stringsAsFactors = F)

#lncRNAEXP=log2(lncRNA[,2:ncol(lncRNA)]+1)

#lncRNA=cbind(lncRNA[1],lncRNAEXP)

v1<-as.matrix(lncRNA[,c(2:ncol(lncRNA))])

v2 <- as.matrix(lncRNA$Status)

myfit <- glmnet(v1, v2, alpha=1,family='binomial')

plot(myfit, xvar = "lambda", label =FALSE)

myfit1 <- cv.glmnet(v1, v2, alpha=1)

plot(myfit1)

abline(v=log(c(myfit1$lambda.min,myfit1$lambda.1se)),lty="dashed")

myfit1$lambda.min

coe <- coef(myfit, s = myfit1$lambda.min)

act\_index <- which(coe != 0)

act\_coe <- coe[act\_index]

row.names(coe)[act\_index]

myfit1$lambda.min

myfit1$lambda.1se

coef(myfit1,s="lambda.min")

coef=coef(myfit, s = myfit1$lambda.min)

index=which(coef != 0)

myfit1$lambda.min

myfit1$lambda.1se

coef(myfit1,s="lambda.min")

LassoGene=row.names(coef)[index]

LassoGene=LassoGene[-1]

write.table(LassoGene, file="LA.txt", sep="\t", quote=F, row.names=F, col.names=F)

library(e1071)

library(kernlab)

library(caret)

set.seed(123)

inputFile="PRGE.txt" #输入文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\GEO Datasets\\机器学习")

#设置工作目录

#读取输入文件

data=read.table(inputFile, header=T, sep="\t", check.names=F, row.names=1)

data=t(data)

group=gsub("(.\*)\\\_(.\*)", "\\2", row.names(data))

#SVM-RFE分析

Profile=rfe(x=data,

y=as.numeric(as.factor(group)),

sizes = c(2,4,6,8, seq(10,40,by=3)),

rfeControl = rfeControl(functions = caretFuncs, method = "cv"),

methods="svmRadial")

#绘制图形

pdf(file="SVM-RFE.pdf", width=6, height=7)

par(las=1)

x = Profile$results$Variables

y = Profile$results$RMSE

plot(x, y, xlab="Variables", ylab="RMSE (Cross-Validation)", col="darkgreen")

lines(x, y, col="darkgreen")

#标注交叉验证误差最小的点

wmin=which.min(y)

wmin.x=x[wmin]

wmin.y=y[wmin]

points(wmin.x, wmin.y, col="blue", pch=16)

text(wmin.x, wmin.y, paste0('N=',wmin.x), pos=2, col=2)

dev.off()

#输出选择的基因

featureGenes=Profile$optVariables

write.table(file="SVM.txt", featureGenes, sep="\t", quote=F, row.names=F, col.names=F)

**UpSetR**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\UpSetR")

#设置工作目录

library(UpSetR)

library(openxlsx)

library(RColorBrewer)

outFile="intersectGenes.txt" #输出交集基因文件

files=dir() #获取目录下所有文件

files=grep("txt$",files,value=T) #提取.txt结尾的文件

geneList=list()

#获取所有txt文件中的基因信息，保存到geneList

for(i in 1:length(files)){

inputFile=files[i]

if(inputFile==outFile){next}

rt=read.table(inputFile,header=F) #读取输入文件

geneNames=as.vector(rt[,1]) #提取基因名称

geneNames=gsub("^ | $","",geneNames) #去掉基因首尾的空格

uniqGene=unique(geneNames) #基因取unique，唯一基因列表

header=unlist(strsplit(inputFile,"\\.|\\-"))

geneList[[header[1]]]=uniqGene

uniqLength=length(uniqGene)

print(paste(header[1],uniqLength,sep=" "))

}

####第二步、取交集作图#####

upsetData=fromList(geneList)

pdf(file="UpSetR.pdf",width=12,height=7)

upset(upsetData,

nsets = length(geneList), #展示多少个数据.

nintersects =50, #展示基因集数目，医学学霸帮为了变成万能代码，就改成这样。

order.by = "freq", #按照数目排序,freq降序，degree升序.

show.numbers = "yes", #柱状图上方是否显示数值

number.angles = 0, #字体角度

point.size = 2, #点的大小

main.bar.color = 'cadetblue4', #y轴柱状图颜色

sets.bar.color=brewer.pal(6,"Set1"),#x轴柱状图的颜色;Set1中只有9个颜色，Set3中有12个颜色，Paired中有12个颜色

matrix.color="#CD5C5C", #交集点颜色

line.size = 1, #线条粗细

mainbar.y.label = "Gene Intersections",

sets.x.label = "Set Size",

text.scale = c(1.5, 1.5, 1.5, 1.5, 1.7, 1.8))

dev.off()

####第三步、输出交集结果#####

intersectGenes=Reduce(intersect,geneList)

write.table(file=outFile,intersectGenes,sep="\t",quote=F,col.names=F,row.names=F)

**ANN**

library(neuralnet)

library(NeuralNetTools)

set.seed(123)

inputFile="ANN.txt"

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\ANN")

data=read.table(inputFile, header=T, sep="\t", check.names=F, row.names=1)

data=as.data.frame(t(data))

group=gsub("(.\*)\\\_(.\*)", "\\2", row.names(data))

data$con=ifelse(group=="con", 1, 0)

data$treat=ifelse(group=="treat", 1, 0)

fit=neuralnet(con+treat~., data, hidden=3)

fit$result.matrix

fit$weight

#plot(fit)

pdf(file="Neural.pdf", width=12, height=8)

plotnet(fit)

dev.off()

net.predict=compute(fit, data)$net.result

net.prediction=c("con", "treat")[apply(net.predict, 1, which.max)]

predict.table=table(group, net.prediction)

predict.table

conAccuracy=predict.table[1,1]/(predict.table[1,1]+predict.table[1,2])

treatAccuracy=predict.table[2,2]/(predict.table[2,1]+predict.table[2,2])

paste0("Con accuracy: ", sprintf("%.3f", conAccuracy))

paste0("Treat accuracy: ", sprintf("%.3f", treatAccuracy))

colnames(net.predict)=c("con", "treat")

outTab=rbind(id=colnames(net.predict), net.predict)

write.table(outTab, file="Neural.txt", sep="\t", quote=F, col.names=F)

**模型评价**

**ROC曲线**

library(rms)

library(foreign)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\模型评价")

data=read.table("dROC.txt",sep="\t",header=T,check.names=F,row.names = 1)

ddist <- datadist(data)

options(datadist='ddist')

modelA <- glm(Status~., data = data, family = binomial(link="logit"))

summary(modelA)

cbind(coef= coef(modelA),confint(modelA))

exp(cbind(OR= coef(modelA),confint(modelA)))

Score<- predict(newdata=data,modelA,"response")

write.table(cbind(id=rownames(cbind(data[,1:9],Score)),cbind(data[,1:9],Score)),"RiskScore.txt",sep="\t",quote=F,row.names=F)

data=read.table("RiskScore.txt",sep="\t",header=T,check.names=F,row.names = 1)

library(pROC)

gmodelA <- roc(Status~Score, data = data,smooth=F)

plot(gmodelA, print.auc=TRUE, print.thres=TRUE,main = "ROC CURVE", col= "blue",print.thres.col="blue",identity.col="blue",

identity.lty=1,identity.lwd=1)

plot(gmodelA, print.auc=TRUE, main = "ROC CURVE", col= "blue",print.thres.col="blue",identity.col="blue",

identity.lty=1,identity.lwd=1)

**Bar**

library(plyr)

library(ggplot2)

library(ggpubr)

scoreFile="Bar.txt" #m6A打分文件

trait="Status" #临床性状

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\模型评价")

#读取输入文件

rt=read.table(scoreFile, header=T, sep="\t", check.names=F, row.names=1)

#定义临床性状的颜色

bioCol=c("#00AFBB","#E7B800","#0066FF","#FF0000","#FF9900","#6E568C","#7CC767","#223D6C","#D20A13","#FFD121","#088247","#11AA4D")

bioCol=bioCol[1:length(unique(rt[,trait]))]

#统计高低评分组病人数目

rt1=rt[,c(trait, "group")]

colnames(rt1)=c("trait", "group")

df=as.data.frame(table(rt1))

#计算高低评分组的百分率

df=ddply(df, .(group), transform, percent = Freq/sum(Freq) \* 100)

#百分比位置

df=ddply(df, .(group), transform, pos = (cumsum(Freq) - 0.5 \* Freq))

df$label=paste0(sprintf("%.0f", df$percent), "%")

df$group=factor(df$group, levels=c("Low", "High"))

#绘制百分率图

p=ggplot(df, aes(x = factor(group), y = percent, fill = trait)) +

geom\_bar(position = position\_stack(), stat = "identity", width = .3) +

scale\_fill\_manual(values=bioCol)+

xlab("PRG score")+ ylab("Percent weight")+ guides(fill=guide\_legend(title=trait))+

geom\_text(aes(label = label), position = position\_stack(vjust = 0.5), size = 4) +

#coord\_flip()+

theme\_bw()

pdf(file="Bar.pdf", width=6, height=5)

print(p)

dev.off()

**DCA**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\模型评价")

source("dca.R")

library(nricens)

library(rms)

library(foreign)

dev=read.table("dROC.txt",sep="\t",header=T,check.names=F,row.names = 1)

modelA <- glm(Status~CASP1+IL18+IL1B+IL6+NLRP1+NLRP2+NLRP3+TNF, data = dev, family = binomial(link="logit"),x=TRUE)

summary(modelA)

dev$PRG\_score<- predict(newdata=dev,modelA,"response")

#Decision Curve Analysis

pdf(file="DCA.pdf", width=6, height=5)

dca(data=dev, outcome="Status", predictors=c("PRG\_score"),smooth="TRUE", probability=c("TRUE"))

dev.off()

**亚组比较**

library(ggpubr)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\模型评价\\GSE57691") #设置工作目录

data=read.table("Clinical.txt", header=T, sep="\t", check.names=F, row.names=1)

#设置比较组

data$group=factor(data$group, levels=c("Small AAA", "Large AAA"))

group=levels(factor(data$group))

comp=combn(group, 2)

my\_comparisons=list()

for(i in 1:ncol(comp)){my\_comparisons[[i]]<-comp[,i]}

#对免疫治疗打分进行循环,分别绘制小提琴图

for(i in colnames(data)[1:(ncol(data)-1)]){

rt=data[,c(i, "group")]

colnames(rt)=c("IPS", "group")

gg1=ggboxplot(rt, x="group", y="IPS", fill = "group",

xlab="", ylab=i,

legend.title="",

palette=c("#0066FF","#FF0000"), width = .3) + rotate\_x\_text(20)+

stat\_compare\_means(comparisons = my\_comparisons,symnum.args=list(cutpoints = c(0, 0.001, 0.01, 0.05, 1), symbols = c("\*\*\*", "\*\*", "\*", "ns")),label = "p.signif")

pdf(file=paste0(i, ".pdf"), width=5, height=6)

print(gg1)

dev.off()}

library(ggpubr)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\模型评价") #设置工作目录

data=read.table("eAAA.txt", header=T, sep="\t", check.names=F, row.names=1)

#设置比较组

data$group=factor(data$group, levels=c("eAAA", "rAAA"))

group=levels(factor(data$group))

comp=combn(group, 2)

my\_comparisons=list()

for(i in 1:ncol(comp)){my\_comparisons[[i]]<-comp[,i]}

#对免疫治疗打分进行循环,分别绘制小提琴图

for(i in colnames(data)[1:(ncol(data)-1)]){

rt=data[,c(i, "group")]

colnames(rt)=c("IPS", "group")

gg1=ggboxplot(rt, x="group", y="IPS", fill = "group",

xlab="", ylab=i,

legend.title="",

palette=c("#0066FF","#FF0000"), width = .3) + rotate\_x\_text(20)+

stat\_compare\_means(comparisons = my\_comparisons,symnum.args=list(cutpoints = c(0, 0.001, 0.01, 0.05, 1), symbols = c("\*\*\*", "\*\*", "\*", "ns")),label = "p.signif")

pdf(file=paste0(i, ".pdf"), width=5, height=6)

print(gg1)

dev.off()}

**临床验证**

**ROC曲线**

library(rms)

library(foreign)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\临床验证")

data=read.table("dROC.txt",sep="\t",header=T,check.names=F,row.names = 1)

ddist <- datadist(data)

options(datadist='ddist')

modelA <- glm(Status~., data = data, family = binomial(link="logit"))

summary(modelA)

cbind(coef= coef(modelA),confint(modelA))

exp(cbind(OR= coef(modelA),confint(modelA)))

Score<- predict(newdata=data,modelA,"response")

write.table(cbind(id=rownames(cbind(data[,1:9],Score)),cbind(data[,1:9],Score)),"RiskScore.txt",sep="\t",quote=F,row.names=F)

data=read.table("RiskScore.txt",sep="\t",header=T,check.names=F,row.names = 1)

library(pROC)

gmodelA <- roc(Status~Score, data = data,smooth=F)

plot(gmodelA, print.auc=TRUE, print.thres=TRUE,main = "ROC CURVE", col= "blue",print.thres.col="blue",identity.col="blue",

identity.lty=1,identity.lwd=1)

plot(gmodelA, print.auc=TRUE, main = "ROC CURVE", col= "blue",print.thres.col="blue",identity.col="blue",

identity.lty=1,identity.lwd=1)

**相关分析**

library(reshape2)

library(ggpubr)

library(ggExtra)

library(pheatmap)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\临床验证")

rt=read.table("AAA.txt", header=T, sep="\t", check.names=F, row.names=1)

#读取输入文件

#绘制基因与免疫细胞相关性的散点图

x=as.numeric(rt[,"Score"])

y=as.numeric(rt[,"Aneurysm diameter"])

df1=as.data.frame(cbind(x,y))

p1=ggplot(df1, aes(x, y)) +

xlab("PRG score") +

ylab("Aneurysm diameter") +

geom\_point() + geom\_smooth(method="lm",formula = y ~ x) + theme\_bw()+

stat\_cor(method = 'spearman', aes(x =x, y =y))

p2=ggMarginal(p1, type="density", xparams=list(fill = "#00AFBB"), yparams=list(fill = "#E7B800"))

p2

**亚组比较**

library(ggpubr)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\临床验证")

data=read.table("PRG.txt", header=T, sep="\t", check.names=F, row.names=1)

#设置比较组

data$group=factor(data$group, levels=c("Low", "High"))

group=levels(factor(data$group))

comp=combn(group, 2)

my\_comparisons=list()

for(i in 1:ncol(comp)){my\_comparisons[[i]]<-comp[,i]}

#对免疫治疗打分进行循环,分别绘制小提琴图

for(i in colnames(data)[1:(ncol(data)-1)]){

rt=data[,c(i, "group")]

colnames(rt)=c("IPS", "group")

gg1=ggboxplot(rt, x="group", y="IPS", fill = "group",

xlab="", ylab=i,

legend.title="PRG",

palette=c("#0066FF","#FF0000"), width = .3) + rotate\_x\_text(20)+

stat\_compare\_means(comparisons = my\_comparisons,symnum.args=list(cutpoints = c(0, 0.001, 0.01, 0.05, 1), symbols = c("\*\*\*", "\*\*", "\*", "ns")),label = "p.signif")

pdf(file=paste0(i, ".pdf"), width=5.5, height=5.5)

print(gg1)

dev.off()}

**免疫细胞**

**ssSGEA**

library(GSVA)

library(limma)

library(GSEABase)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\免疫细胞")

#设置工作目录

inputFile="geneMatrix.txt" #输入文件

gmtFile="ImmuneCell.gmt" #GMT文件

#读取输入文件，并对输入文件处理

rt=read.table(inputFile,sep="\t",header=T,check.names=F)

rt=as.matrix(rt)

rownames(rt)=rt[,1]

exp=rt[,2:ncol(rt)]

dimnames=list(rownames(exp),colnames(exp))

mat=matrix(as.numeric(as.matrix(exp)),nrow=nrow(exp),dimnames=dimnames)

mat=avereps(mat)

mat=mat[rowMeans(mat)>0,]

geneSet=getGmt(gmtFile,

geneIdType=SymbolIdentifier())

#ssgsea分析

ssgseaScore=gsva(mat, geneSet, method='ssgsea', kcdf='Gaussian', abs.ranking=TRUE)

#定义ssGSEA score矫正函数

normalize=function(x){

return((x-min(x))/(max(x)-min(x)))}

#对ssGSEA score进行矫正

ssgseaOut=normalize(ssgseaScore)

ssgseaOut=rbind(id=colnames(ssgseaOut),ssgseaOut)

write.table(ssgseaOut,file="ssGSEA.txt",sep="\t",quote=F,col.names=F)

**数据合并**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\免疫细胞")

inputfile1="id.txt"

inputfile2="CIBERSORT.txt"

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"CIBERSORTx.txt",sep = "\t",row.names = F,quote = F)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\免疫细胞")

inputfile1="id.txt"

inputfile2="ssGSEAx.txt"

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"ssGSEA.txt",sep = "\t",row.names = F,quote = F)

**CIBERSORTx数据比较**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\免疫细胞")

library(vioplot) #引用包

library(limma)

rt=read.table("CIBERSORTx.txt",sep="\t",header=T,row.names=1,check.names=F) #读取输入文件

normal=10 #正常样品数目

tumor=49 #肿瘤样品数目

pdf("CIBERSORT.pdf",height=8,width=15) #保存图片的文件名称

par(las=1,mar=c(10,6,3,3))

x=c(1:ncol(rt))

y=c(1:ncol(rt))

plot(x,y,

xlim=c(0,63),ylim=c(min(rt),max(rt)+0.02),

main="",xlab="", ylab="Fraction",

pch=21,

col="white",

xaxt="n")

#对每个免疫细胞循环，绘制vioplot，正常用绿色表示，肿瘤用红色表示

for(i in 1:ncol(rt)){

normalData=rt[1:normal,i]

tumorData=rt[(normal+1):(normal+tumor),i]

vioplot(normalData,at=3\*(i-1),lty=1,add = T,col ='#2E8B57')

vioplot(tumorData,at=3\*(i-1)+1,lty=1,add = T,col = '#D2691E')

wilcoxTest=wilcox.test(normalData,tumorData)

p=round(wilcoxTest$p.value,3)

mx=max(c(normalData,tumorData))

lines(c(x=3\*(i-1)+0.2,x=3\*(i-1)+0.8),c(mx,mx))

text(x=3\*(i-1)+0.5,y=mx+0.02,labels=ifelse(p<0.001,paste0("p<0.001"),paste0("p=",p)),cex = 0.8)

legend("topleft",

c("Control Abdominal Aorta", "Abdominal Aortic Aneurysm"),

lwd=3,bty="n",cex=0.8,

col=c("#2E8B57", "#D2691E"))

text(seq(1,64,3),-0.05,xpd = NA,labels=colnames(rt),cex = 0.8,srt = 45,pos=2)

}

dev.off()

**ssGSEA数据比较**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\免疫细胞")

library(vioplot) #引用包

library(limma)

rt=read.table("ssGSEA.txt",sep="\t",header=T,row.names=1,check.names=F) #读取输入文件

normal=10 #正常样品数目

tumor=49 #肿瘤样品数目

pdf("ssGSEA.pdf",height=8.5,width=17) #保存图片的文件名称

par(las=1,mar=c(10,6,3,3))

x=c(1:ncol(rt))

y=c(1:ncol(rt))

plot(x,y,

xlim=c(0,76),ylim=c(min(rt),max(rt)+0.02),

main="",xlab="", ylab="Fraction",

pch=21,

col="white",

xaxt="n")

#对每个免疫细胞循环，绘制vioplot，正常用绿色表示，肿瘤用红色表示

for(i in 1:ncol(rt)){

normalData=rt[1:normal,i]

tumorData=rt[(normal+1):(normal+tumor),i]

vioplot(normalData,at=3\*(i-1),lty=1,add = T,col ='#2E8B57')

vioplot(tumorData,at=3\*(i-1)+1,lty=1,add = T,col = '#D20A13')

wilcoxTest=wilcox.test(normalData,tumorData)

p=round(wilcoxTest$p.value,3)

mx=max(c(normalData,tumorData))

lines(c(x=3\*(i-1)+0.2,x=3\*(i-1)+0.8),c(mx,mx))

text(x=3\*(i-1)+0.5,y=mx+0.02,labels=ifelse(p<0.001,paste0("p<0.001"),paste0("p=",p)),cex = 0.8)

legend("topleft",

c("Control Abdominal Aorta", "Abdominal Aortic Aneurysm"), lwd=3,bty="n",cex=0.8,

col=c("#2E8B57", "#D20A13"))

text(seq(1,76,3),-0.09,xpd = NA,labels=colnames(rt),cex = 0.8,srt = 45,pos=2)

}

dev.off()

**Cor ICI**

library(limma)

library(reshape2)

library(ggplot2)

immFile="ssGSEAx.txt" #免疫细胞浸润结果文件

riskFile="Risk.txt" #风险文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\ICI Cor")

#设置工作目录

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

data=data[sameSample,,drop=F]

risk=risk[sameSample,,drop=F]

#相关性分析

outTab=data.frame()

for(immune in colnames(data)){

for(gene in colnames(risk)){

x=as.numeric(data[,immune])

y=as.numeric(risk[,gene])

corT=cor.test(x,y,method="spearman")

cor=corT$estimate

pvalue=corT$p.value

text=ifelse(pvalue<0.001,"\*\*\*",ifelse(pvalue<0.01,"\*\*",ifelse(pvalue<0.05,"\*","")))

outTab=rbind(outTab,cbind(Gene=gene, Immune=immune, cor=cor, text, pvalue))

}

}

write.table(outTab,file="Cor.txt",sep="\t",row.names=F,quote=F)

#绘制相关性热图

#outTab=read.table("Cor.txt",sep="\t",header=T,check.names=F)

outTab$cor=as.numeric(outTab$cor)

pdf(file="Cor.pdf", width=14, height=6)

ggplot(outTab, aes(Immune, Gene)) +

geom\_tile(aes(fill = cor), colour = "grey", size = 1)+

scale\_fill\_gradient2(low = "#5C5DAF", mid = "white", high = "#EA2E2D") +

geom\_text(aes(label=text),col ="black",size = 3) +

theme\_minimal() + #去掉背景

theme(axis.title.x=element\_blank(), axis.ticks.x=element\_blank(), axis.title.y=element\_blank(),

axis.text.x = element\_text(angle = 60, hjust = 1, size = 10, face = "bold"), #x轴字体

axis.text.y = element\_text(size = 10, face = "bold")) + #y轴字体

labs(fill =paste0("\*\*\* p<0.001","\n", "\*\* p<0.01","\n", " \* p<0.05","\n", "\n","Correlation")) + #设置图例

scale\_x\_discrete(position = "bottom") #X轴名称显示位置

dev.off()

library(limma)

library(reshape2)

library(ggplot2)

immFile="CIBERSORT.txt" #免疫细胞浸润结果文件

riskFile="Risk.txt" #风险文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\ICI Cor")

#设置工作目录

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

data=data[sameSample,,drop=F]

risk=risk[sameSample,,drop=F]

#相关性分析

outTab=data.frame()

for(immune in colnames(data)){

for(gene in colnames(risk)){

x=as.numeric(data[,immune])

y=as.numeric(risk[,gene])

corT=cor.test(x,y,method="spearman")

cor=corT$estimate

pvalue=corT$p.value

text=ifelse(pvalue<0.001,"\*\*\*",ifelse(pvalue<0.01,"\*\*",ifelse(pvalue<0.05,"\*","")))

outTab=rbind(outTab,cbind(Gene=gene, Immune=immune, cor=cor, text, pvalue))

}

}

write.table(outTab,file="Cor.txt",sep="\t",row.names=F,quote=F)

#绘制相关性热图

#outTab=read.table("Cor.txt",sep="\t",header=T,check.names=F)

outTab$cor=as.numeric(outTab$cor)

pdf(file="Cor.pdf", width=14, height=6)

ggplot(outTab, aes(Immune, Gene)) +

geom\_tile(aes(fill = cor), colour = "grey", size = 1)+

scale\_fill\_gradient2(low = "#5C5DAF", mid = "white", high = "#EA2E2D") +

geom\_text(aes(label=text),col ="black",size = 3) +

theme\_minimal() + #去掉背景

theme(axis.title.x=element\_blank(), axis.ticks.x=element\_blank(), axis.title.y=element\_blank(),

axis.text.x = element\_text(angle = 60, hjust = 1, size = 10, face = "bold"), #x轴字体

axis.text.y = element\_text(size = 10, face = "bold")) + #y轴字体

labs(fill =paste0("\*\*\* p<0.001","\n", "\*\* p<0.01","\n", " \* p<0.05","\n", "\n","Correlation")) + #设置图例

scale\_x\_discrete(position = "bottom") #X轴名称显示位置

dev.off()

**相关分析**

library(reshape2)

library(ggpubr)

library(ggExtra)

library(pheatmap)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\ICI Cor")

rt=read.table("M2.txt", header=T, sep="\t", check.names=F, row.names=1)

#读取输入文件

#绘制基因与免疫细胞相关性的散点图

x=as.numeric(rt[,"PRG score"])

y=as.numeric(rt[,"M2 Macrophages"])

df1=as.data.frame(cbind(x,y))

p1=ggplot(df1, aes(x, y)) +

xlab("PRG score") +

ylab("M2 Macrophages") +

geom\_point() + geom\_smooth(method="lm",formula = y ~ x) + theme\_bw()+

stat\_cor(method = 'spearman', aes(x =x, y =y))

p2=ggMarginal(p1, type="density", xparams=list(fill = "#00AFBB"), yparams=list(fill = "#E7B800"))

p2

**GSVA**

**IRP**

library(GSVA)

library(limma)

library(GSEABase)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA\\IRP")

#设置工作目录

inputFile="geneMatrix.txt" #输入文件

gmtFile="Immune.gmt" #GMT文件

#读取输入文件，并对输入文件处理

rt=read.table(inputFile,sep="\t",header=T,check.names=F)

rt=as.matrix(rt)

rownames(rt)=rt[,1]

exp=rt[,2:ncol(rt)]

dimnames=list(rownames(exp),colnames(exp))

mat=matrix(as.numeric(as.matrix(exp)),nrow=nrow(exp),dimnames=dimnames)

mat=avereps(mat)

mat=mat[rowMeans(mat)>0,]

geneSet=getGmt(gmtFile,

geneIdType=SymbolIdentifier())

#ssgsea分析

ssgseaScore=gsva(mat, geneSet, method='ssgsea', kcdf='Gaussian', abs.ranking=TRUE)

#定义ssGSEA score矫正函数

normalize=function(x){

return((x-min(x))/(max(x)-min(x)))}

#对ssGSEA score进行矫正

ssgseaOut=normalize(ssgseaScore)

ssgseaOut=rbind(id=colnames(ssgseaOut),ssgseaOut)

write.table(ssgseaOut,file="IRP.txt",sep="\t",quote=F,col.names=F)

**KEGG**

library(GSVA)

library(limma)

library(GSEABase)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA\\KEGG") #设置工作目录

inputFile="geneMatrix.txt"

gmtFile="KEGG.gmt" #GMT文件

#读取输入文件，并对输入文件处理

rt=read.table(inputFile,sep="\t",header=T,check.names=F)

rt=as.matrix(rt)

rownames(rt)=rt[,1]

exp=rt[,2:ncol(rt)]

dimnames=list(rownames(exp),colnames(exp))

mat=matrix(as.numeric(as.matrix(exp)),nrow=nrow(exp),dimnames=dimnames)

mat=avereps(mat)

mat=mat[rowMeans(mat)>0,]

geneSet=getGmt(gmtFile,

geneIdType=SymbolIdentifier())

#ssgsea分析

ssgseaScore=gsva(mat, geneSet, method='ssgsea', kcdf='Gaussian', abs.ranking=TRUE)

#定义ssGSEA score矫正函数

normalize=function(x){

return((x-min(x))/(max(x)-min(x)))}

#对ssGSEA score进行矫正

ssgseaOut=normalize(ssgseaScore)

ssgseaOut=rbind(id=colnames(ssgseaOut),ssgseaOut)

write.table(ssgseaOut,file="KEGG.txt",sep="\t",quote=F,col.names=F)

**IRP差异分析**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA\\IRP")

inputfile1="id.txt" #生存时间数据

inputfile2="IRPx.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"IRP.txt",sep = "\t",row.names = F,quote = F)

library(limma)

library(reshape2)

library(ggpubr)

riskFile="Group.txt" #分组

immFile="IRP.txt" #基因表达结果文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA\\IRP")

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

#读取风险文件

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

rt=cbind(data[sameSample,,drop=F], risk[sameSample,"Risk",drop=F])

rt=rt[order(rt$Risk, decreasing=T),]

conNum=nrow(rt[rt$Risk=="Control Abdominal Aorta",])

treatNum=nrow(rt[rt$Risk=="Abdominal Aortic Aneurysm",])

##################绘制箱线图##################

#把数据转换成ggplot2输入文件

data=rt

data=melt(data, id.vars=c("Risk"))

colnames(data)=c("Risk", "Immune", "Expression")

#绘制箱线图

group=levels(factor(data$Risk))

data$Risk=factor(data$Risk, levels=c("Control Abdominal Aorta","Abdominal Aortic Aneurysm"))

bioCol=c("#00BFFF","#F08080","#0066FF","#FF0000","#6E568C","#7CC767","#223D6C","#D20A13","#FFD121","#088247","#11AA4D")

bioCol=bioCol[1:length(group)]

boxplot=ggboxplot(data, x="Immune", y="Expression", fill="Risk", xlab="",

ylab="Enrichment score",

legend.title="Type",

width=0.8,

palette=bioCol)+

rotate\_x\_text(50)+

stat\_compare\_means(aes(group=Risk),symnum.args=list(cutpoints=c(0, 0.001, 0.01, 0.05, 1), symbols=c("\*\*\*", "\*\*", "\*", "ns")), label="p.signif")

#输出图片

pdf(file="IRP.pdf", width=9, height=7)

print(boxplot)

dev.off()

**KEGG****差异分析**

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA\\KEGG")

inputfile1="id.txt" #生存时间数据

inputfile2="KEGG.txt" #差异基因表达数据

time\_data<-read.table(inputfile1,header = T,sep = "\t",check.names = F)

geneEXP<-read.table(inputfile2,header = T,sep = "\t",check.names = F)

head(time\_data)

head(geneEXP)

merger\_data<-merge(time\_data,geneEXP,by="id")

write.table(merger\_data,"KEGGx.txt",sep = "\t",row.names = F,quote = F)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA\\KEGG")

rt=read.table("KEGGx.txt", header=T, sep="\t", row.names=1)

library(limma)

# 设置或导入分组

group <- factor(c(rep("Tumor", 17), rep("Normal", 31)), levels = c('Tumor', 'Normal'))

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

colnames(design) = levels(factor(group))

rownames(design) = colnames(rt)

design

# Tumor VS Normal

compare <- makeContrasts(Tumor - Normal, levels=design)

fit <- lmFit(rt, design)

fit2 <- contrasts.fit(fit, compare)

fit3 <- eBayes(fit2)

dif <- topTable(fit3, coef=1, number=200)

head(dif)

write.table(dif,file="KEGGDiff.txt",sep="\t",quote=F)

rt=read.table("KEGGDif.txt", header=T, sep="\t", row.names=1)

dat\_plot<-data.frame(id=row.names(rt),t=rt$t) #去掉"KEGG\_"

library(stringr)

library(dplyr)

dat\_plot$id<-str\_replace(dat\_plot$id,"KEGG ","")

#新增一列根据t阈值分类

dat\_plot$threshold=factor(ifelse(dat\_plot$t>-2,ifelse(dat\_plot$t>=2,'Up','NoSignifi'),'Down'),levels=c('Up','Down','NoSignifi'))

#排序

dat\_plot<-dat\_plot%>%arrange(t)

#变成因子类型

dat\_plot$id<-factor(dat\_plot$id,levels=dat\_plot$id)

#绘制

library(ggplot2)

library(ggthemes)

#install.packages("ggprism")

library(ggprism)

p<-ggplot(data=dat\_plot,aes(x=id,y=t,fill=threshold))+

geom\_col()+

coord\_flip()+

scale\_fill\_manual(values=c('Up'='#36638a','NoSignifi'='#cccccc','Down'='#7bcd7b'))+

geom\_hline(yintercept=c(-2,2),color='white',size=0.5,lty='dashed')+

xlab('')+

ylab('t value of GSVA score, Abdominal Aortic Aneurysm versus Control Abdominal Aorta')+#注意坐标轴旋转了

guides(fill=F)+#不显示图例

theme\_prism(border=T)+

theme(

axis.text.y=element\_blank(),

axis.ticks.y=element\_blank()

)

p

#小于-2的数量

low1 <- dat\_plot %>% filter(t < -2) %>% nrow()

# 小于0总数量

low0 <- dat\_plot %>% filter( t < 0) %>% nrow()

# 小于2总数量

high0 <- dat\_plot %>% filter(t < 2) %>% nrow()

# 总的柱子数量

high1 <- nrow(dat\_plot)

#依次从下到上添加标签

p<-p+geom\_text(data=dat\_plot[1:low1,],aes(x=id,y=0.1,label=id),

hjust=0,color='black')+#小于-1的为黑色标签

geom\_text(data=dat\_plot[(low1+1):low0,],aes(x=id,y=0.1,label=id),

hjust=0,color='grey')+#灰色标签

geom\_text(data=dat\_plot[(low0+1):high0,],aes(x=id,y=-0.1,label=id),

hjust=1,color='grey')+#灰色标签

geom\_text(data=dat\_plot[(high0+1):high1,],aes(x=id,y=-0.1,label=id),

hjust=1,color='black')#大于1的为黑色标签

ggsave("GSVA.pdf",p,width=10, height=5)

**Cor GSVA**

library(limma)

library(reshape2)

library(ggplot2)

immFile="GSVA.txt" #免疫细胞浸润结果文件

riskFile="Risk.txt" #风险文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\GSVA Cor") #设置工作目录

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

data=data[sameSample,,drop=F]

risk=risk[sameSample,,drop=F]

#相关性分析

outTab=data.frame()

for(immune in colnames(data)){

for(gene in colnames(risk)){

x=as.numeric(data[,immune])

y=as.numeric(risk[,gene])

corT=cor.test(x,y,method="spearman")

cor=corT$estimate

pvalue=corT$p.value

text=ifelse(pvalue<0.001,"\*\*\*",ifelse(pvalue<0.01,"\*\*",ifelse(pvalue<0.05,"\*","")))

outTab=rbind(outTab,cbind(Gene=gene, Immune=immune, cor=cor, text, pvalue))

}

}

write.table(outTab,file="Cor.txt",sep="\t",row.names=F,quote=F)

#绘制相关性热图

#outTab=read.table("Cor.txt",sep="\t",header=T,check.names=F)

outTab$cor=as.numeric(outTab$cor)

pdf(file="Cor.pdf", width=9, height=6)

ggplot(outTab, aes(Immune, Gene)) +

geom\_tile(aes(fill = cor), colour = "grey", size = 1)+

scale\_fill\_gradient2(low = "#5C5DAF", mid = "white", high = "#EA2E2D") +

geom\_text(aes(label=text),col ="black",size = 3) +

theme\_minimal() + #去掉背景

theme(axis.title.x=element\_blank(), axis.ticks.x=element\_blank(), axis.title.y=element\_blank(),

axis.text.x = element\_text(angle = 30, hjust = 1, size = 10, face = "bold"), #x轴字体

axis.text.y = element\_text(size = 10, face = "bold")) + #y轴字体

labs(fill =paste0("\*\*\* p<0.001","\n", "\*\* p<0.01","\n", " \* p<0.05","\n", "\n","Correlation")) + #设置图例

scale\_x\_discrete(position = "bottom") #X轴名称显示位置

dev.off()

**基础验证**

**PCR**

library(limma)

library(reshape2)

library(ggpubr)

riskFile="Group.txt" #分组

immFile="PCR.txt" #基因表达结果文件

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\基础研究")

#设置工作目录

#读取免疫细胞结果文件，并对数据进行整理

data=read.table(immFile, header=T, sep="\t", check.names=F, row.names=1)

#读取风险文件

risk=read.table(riskFile, header=T, sep="\t", check.names=F, row.names=1)

sameSample=intersect(row.names(data), row.names(risk))

rt=cbind(data[sameSample,,drop=F], risk[sameSample,"Risk",drop=F])

rt=rt[order(rt$Risk, decreasing=T),]

conNum=nrow(rt[rt$Risk=="Control",])

treatNum=nrow(rt[rt$Risk=="Abdominal Aortic Aneurysm",])

##################绘制箱线图##################

#把数据转换成ggplot2输入文件

data=rt

data=melt(data, id.vars=c("Risk"))

colnames(data)=c("Risk", "Immune", "Expression")

#绘制箱线图

group=levels(factor(data$Risk))

data$Risk=factor(data$Risk, levels=c("Control","Abdominal Aortic Aneurysm"))

bioCol=c("#0066FF","#FF0000","#6E568C","#7CC767","#223D6C","#D20A13","#FFD121","#088247","#11AA4D")

bioCol=bioCol[1:length(group)]

boxplot=ggboxplot(data, x="Immune", y="Expression", fill="Risk",

xlab="",

ylab="Relative Expression",

legend.title="Type",

width=0.6,

palette=bioCol)+

rotate\_x\_text(50)+

stat\_compare\_means(aes(group=Risk),symnum.args=list(cutpoints=c(0, 0.001, 0.01, 0.05, 1), symbols=c("\*\*\*", "\*\*", "\*", "ns")), label="p.signif")

#输出图片

pdf(file="PRG.pdf", width=9, height=6)

print(boxplot)

dev.off()

**microRNA比较**

library(ggpubr)

setwd("D:\\科研论文\\科研学术\\非肿瘤生信\\AAA\\数据分析\\基础研究")#设置工作目录

data=read.table("MicroRNA.txt", header=T, sep="\t", check.names=F, row.names=1)

#设置比较组

data$group=factor(data$group, levels=c("Control", "AAA SMC"))

group=levels(factor(data$group))

comp=combn(group, 2)

my\_comparisons=list()

for(i in 1:ncol(comp)){my\_comparisons[[i]]<-comp[,i]}

#对免疫治疗打分进行循环,分别绘制小提琴图

for(i in colnames(data)[1:(ncol(data)-1)]){

rt=data[,c(i, "group")]

colnames(rt)=c("IPS", "group")

gg1=ggboxplot(rt, x="group", y="IPS", fill = "group",

xlab="", ylab=i,

legend.title="",

palette=c("#2E8B57", "#D2691E"), width = .3) + rotate\_x\_text(20)+

stat\_compare\_means(comparisons = my\_comparisons,symnum.args=list(cutpoints = c(0, 0.001, 0.01, 0.05, 1), symbols = c("\*\*\*", "\*\*", "\*", "ns")),label = "p.signif")

pdf(file=paste0(i, ".pdf"), width=5, height=6)

print(gg1)

dev.off()}