setwd("D:/AaPaper/LUADFerro/")

#####数据读入#####

LUAD\_TPM <- read.csv("D:/AaPaper/Date整理/TCGA/LUAD/LUAD\_TPM\_Tumor.csv")

library(tidyverse)

LUAD\_TPM <- LUAD\_TPM %>% remove\_rownames() %>% column\_to\_rownames("X")

LUAD\_Survival <- read.csv("D:/AaPaper/Date整理/TCGA/LUAD/LUAD\_Survival.csv")

LUAD\_Survival$X <- NULL

LUAD\_Survival$OS.time <- round(LUAD\_Survival$OS.time/30,1)

LUAD\_Clinical <- read.csv("D:/AaPaper/Date整理/TCGA/LUAD/LUAD\_PhenoType.csv")

LUAD\_Clinical <- tibble::column\_to\_rownames(LUAD\_Clinical,"X")

#通路基因集

Ferroptosis <- read.csv("D:/AaPaper/LUSC/数据/整理/Ferroptosis Gene Set.csv")

#####评分机制#####

library(IOBR)

scoreIPS <- deconvo\_ips(eset = 2^LUAD\_TPM-1,project = "TCGA-LUAD",plot=F)

scoreESTIMATE <- deconvo\_tme(eset = 2^LUAD\_TPM-1,method = "estimate",

platform ="affymetrix")#数据必须non-log scale

#Score <- merge(dplyr::select(scoreESTIMATE,1,2,3,5),dplyr::select(scoreIPS,1,9),by="ID")

Score <- scoreESTIMATE

#####WGCNA#####

# Load the WGCNA package

library(flashClust)

library(iterators)

library(WGCNA);

# The following setting is important, do not omit.

options(stringsAsFactors = FALSE)

#Read in the cancer data set

RawData <- LUAD\_TPM[which(rownames(LUAD\_TPM) %in% Ferroptosis$Ferroptosis),]

any(duplicated(rownames(RawData))) #检测有无重复的名字

datExpr0 = as.data.frame(t(RawData))

gsg = goodSamplesGenes(datExpr0, verbose = 3);#检测缺失值

gsg$allOK #结果为TRUE，则所有选定基因都用于后续WGCNA

#如果gsg$allOK结果为FALSE，则后续选择good gene用于WGCNA

if (!gsg$allOK)

{

# Optionally, print the gene and sample names that were removed:

if (sum(!gsg$goodGenes) > 0)

printFlush(paste("Removing genes:", paste(names(datExpr0)[!gsg$goodGenes], collapse = ", ")));

if (sum(!gsg$goodSamples) > 0)

printFlush(paste("Removing samples:", paste(rownames(datExpr0)[!gsg$goodSamples], collapse = ", ")));

# Remove the offending genes and samples from the data:

datExpr0 = datExpr0[gsg$goodSamples, gsg$goodGenes]

}

#Ferroptosis结果Removing genes:SNORA16A,XBP1

#聚类

sampleTree = hclust(dist(datExpr0), method = "average");

# Plot the sample tree: Open a graphic output window of size 12 by 9 inches

# The user should change the dimensions if the window is too large or too small.

sizeGrWindow(14,9)

#pdf(file = "Plot/sampleClustering.pdf", width = 14, height = 9);

par(cex = 0.6);

par(mar = c(0,4,2,0))

#dev.off()

plot(sampleTree, main = "Sample clustering to detect outliers", sub="", xlab="", cex.lab = 1.5,

cex.axis = 1.5, cex.main = 2)

# Plot a line to show the cut

abline(h = 30, col = "red");

# Determine cluster under the line

clust = cutreeStatic(sampleTree, cutHeight = 30, minSize = 10)

table(clust)

# clust 1 contains the samples we want to keep.

keepSamples = (clust==1)

datExpr = datExpr0[keepSamples, ]

nGenes = ncol(datExpr)

nSamples = nrow(datExpr)

#clinical traits

datTraits <- Score %>% column\_to\_rownames("ID")

datTraits <- datTraits[which(rownames(datTraits) %in% rownames(datExpr)),]

collectGarbage() #释放内存？

#重新聚类，包含clinical traits

# Re-cluster samples

sampleTree2 = hclust(dist(datExpr), method = "average")

# Convert traits to a color representation: white means low, red means high, grey means missing entry

traitColors = numbers2colors(datTraits, signed = FALSE);

# Plot the sample dendrogram and the colors underneath.

plotDendroAndColors(sampleTree2, traitColors,

groupLabels = names(datTraits),

main = "Sample dendrogram and trait heatmap")

# Choose a set of soft-thresholding powers

powers = c(c(1:10), seq(from = 12, to=20, by=2))

# Call the network topology analysis function

sft = pickSoftThreshold(datExpr, powerVector = powers, verbose = 5)

# Plot the results:

sizeGrWindow(9, 5)

par(mfrow = c(1,2));

cex1 = 0.9;

# Scale-free topology fit index as a function of the soft-thresholding power

plot(sft$fitIndices[,1], -sign(sft$fitIndices[,3])\*sft$fitIndices[,2],

xlab="Soft Threshold (power)",ylab="Scale Free Topology Model Fit,signed R^2",type="n",

main = paste("Scale independence"));

text(sft$fitIndices[,1], -sign(sft$fitIndices[,3])\*sft$fitIndices[,2],

labels=powers,cex=cex1,col="red");

# this line corresponds to using an R^2 cut-off of h

abline(h=0.90,col="red")

# Mean connectivity as a function of the soft-thresholding power

plot(sft$fitIndices[,1], sft$fitIndices[,5],

xlab="Soft Threshold (power)",ylab="Mean Connectivity", type="n",

main = paste("Mean connectivity"))

text(sft$fitIndices[,1], sft$fitIndices[,5], labels=powers, cex=cex1,col="red")

# sft$powerEstimate是推荐的软阈值

net = blockwiseModules(datExpr, power = sft$powerEstimate,

TOMType = "unsigned", minModuleSize = 30,

reassignThreshold = 0, mergeCutHeight = 0.25,

numericLabels = TRUE, pamRespectsDendro = FALSE,

saveTOMs = F,verbose = 3)

#open a graphics window

sizeGrWindow(12, 9)

# Convert labels to colors for plotting

mergedColors = labels2colors(net$colors)

# Plot the dendrogram and the module colors underneath

plotDendroAndColors(net$dendrograms[[1]], mergedColors[net$blockGenes[[1]]],

"Module colors",

dendroLabels = FALSE, hang = 0.03,

addGuide = TRUE, guideHang = 0.05)

moduleLabels = net$colors

moduleColors = labels2colors(net$colors)

MEs = net$MEs

geneTree = net$dendrograms[[1]]

# Recalculate MEs with color labels

MEs0 = moduleEigengenes(datExpr, moduleColors)$eigengenes

MEs = orderMEs(MEs0)

moduleTraitCor = cor(MEs, datTraits, use = "p")

moduleTraitPvalue = corPvalueStudent(moduleTraitCor, nSamples)

#heatmap

sizeGrWindow(10,6)

# Will display correlations and their p-values

textMatrix = paste(signif(moduleTraitCor, 2), "\n(",

signif(moduleTraitPvalue, 1), ")", sep = "");

dim(textMatrix) = dim(moduleTraitCor)

par(mar = c(6, 8.5, 3, 3));

# Display the correlation values within a heatmap plot

labeledHeatmap(Matrix = moduleTraitCor,

xLabels = names(datTraits),

yLabels = names(MEs),

ySymbols = names(MEs),

colorLabels = FALSE,

colors = greenWhiteRed(50),

textMatrix = textMatrix,

setStdMargins = FALSE,

cex.text = 0.5,

zlim = c(-1,1),

main = paste("Module-trait relationships"))

# 各基因模块的名字（颜色）

modNames = substring(names(MEs), 3)

#提取感兴趣的module内所有的基因名称

module <- "grey"

moduleGenes <- moduleColors== module

table(moduleGenes)

grey\_module <- as.data.frame(dimnames(data.frame(datExpr))[[2]][moduleGenes])

names(grey\_module) <- "genename"

write.csv(grey\_module,file = "Data/moduleGenes.csv")

#####筛选hubgene没结果####

#筛选hub gene

##为MM>0.8，GS>0.5或者0.6

# 指定datTrait中感兴趣的一个性状

traitSelect = as.data.frame(datTraits$ESTIMATEScore\_estimate)

names(traitSelect) = "ESTIMATEscore"

# 计算MM的P值

geneModuleMembership = as.data.frame(cor(datExpr, MEs, use = "p"))

MMPvalue = as.data.frame(corPvalueStudent(as.matrix(geneModuleMembership ), nSamples))

names(geneModuleMembership) = paste("MM", modNames, sep="")

names(MMPvalue) = paste("p.MM", modNames, sep="")

# 计算性状和基因表达量之间的相关性（GS）

geneTraitSignificance = as.data.frame(cor(datExpr,traitSelect, use = "p"))

GSPvalue = as.data.frame(corPvalueStudent(as.matrix(geneTraitSignificance),

nSamples))

names(geneTraitSignificance) = paste("GS.", names(traitSelect), sep="")

names(GSPvalue) = paste("p.GS.", names(traitSelect), sep="")

#筛选hub gene

#MEs和MEs0是同一个文件，一个是模块颜色，一个是模块编号，看对应关系

hub<- abs(geneModuleMembership$MMgrey) > 0.8 & abs(geneTraitSignificance) > 0.5

table(hub)

hubgene\_grey <- dimnames(data.frame(datExpr))[[2]][hub]

#构建散点图矩阵

##column是感兴趣的色块所在的列 MM<-abs(geneModuleMembership[moduleGenes,column])

MM < -abs(geneModuleMembership[moduleGenes,1])

GS <- abs(geneTraitSignificance[moduleGenes, 1])

c<-as.data.frame(cbind(MM,GS))

rownames(c)=brown\_module$genename

head(c)

#对基因进行分组

# hub基因和module内全部基因进行匹配，匹配成功返回1，没有匹配到的返回0

match<- brown\_module$genename %in% hubgene\_brown

# 将匹配信息添加到散点图矩阵最后一列

c$group<-match

head(c)

#绘制散点图

library(ggplot2)

pdf("MM vs. GS\_blue\_TL.pdf",width = 7,height = 7)

ggplot(data=c, aes(x=MM, y=GS,color=group))+geom\_point(size=1.5)+

scale\_colour\_manual(values=c("grey60","#DE6757"))+ theme\_bw()+

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

labs(x="Module Membership in blue module", y="Gene significance for TL",

title = "Module membership vs. gene significance ")+

theme(axis.title.x =element\_text(size=14), axis.title.y=element\_text(size=14),

axis.text = element\_text(size = 12),

axis.text.x = element\_text(colour = "black"),

axis.text.y = element\_text(colour = "black"),

plot.title = element\_text(hjust = 0.5,size = 16,face = "bold"),

plot.margin = unit(rep(2,4),'lines')) +

theme(legend.position = 'none')+

geom\_hline(aes(yintercept=0.6),colour="#5B9BD5",lwd=1,linetype=5)+

geom\_vline(aes(xintercept=0.8),colour="#5B9BD5",lwd=1,linetype=5)

######module基因做单因素cox分析#####

library(survival)

library(survminer)

Survival\_model <- datExpr

Survival\_model$sample <- rownames(Survival\_model)

Survival\_model <- merge(LUAD\_Survival[which(LUAD\_Survival$sample %in% rownames(datExpr)),],

Survival\_model,by = "sample")

names(Survival\_model) <- gsub("-","\_",names(Survival\_model))

covariates <- grey\_module$genename

covariates <- gsub("\\.","\_",covariates)##几个基因名不匹配

model\_univ <- sapply(covariates,

function(x) as.formula(paste('Surv(OS.time,OS)~', x)))

model\_univ <- lapply(model\_univ, function(x) {coxph(x, data = Survival\_model)})

model\_univ <- lapply(model\_univ,

function(x){

x <- summary(x)

#获取p值

p.value<-signif(x$wald["pvalue"], digits=2)

#获取HR

HR <-signif(x$coef[2], digits=2);

#获取95%置信区间

HR.confint.lower <- signif(x$conf.int[,"lower .95"], 2)

HR.confint.upper <- signif(x$conf.int[,"upper .95"],2)

HR <- paste0(HR, " (",

HR.confint.lower, "-", HR.confint.upper, ")")

res<-c(p.value,HR)

names(res)<-c("p.value","HR (95% CI for HR)")

return(res)

})

#转换成数据框，并转置

res\_model <- t(as.data.frame(model\_univ,check.names = FALSE))

res\_model <- as.data.frame(res\_model)

res\_model

write.csv(res\_model,file = "Data/Modelgene Univariate Analysis.csv")

names(res\_model)[1] <- "pvalue"

hubGene\_univ <- rownames(res\_model[which(res\_model$pvalue <0.05),])

######lasso分析#####

library(survival)

library(glmnet)

#生存状态和生存时间需要转化为double类型

Survival\_model$OS <- as.double(Survival\_model$OS)

Survival\_model$OS.time <- as.double(Survival\_model$OS.time)

rownames(Survival\_model) <- Survival\_model$sample

dep\_var <- data.matrix(Surv(time = Survival\_model$OS.time,

event = Survival\_model$OS))#设置因变量

Lasso\_exp <- colnames(Survival\_model) %in% hubGene\_univ

marker\_exp <- Survival\_model[,Lasso\_exp]#表达谱

#构建模型

fit\_lasso <- glmnet(x = marker\_exp,dep\_var,family = "cox",alpha = 1)

plot(fit\_lasso,xvar = "lambda")

#交叉验证 #源代码无data.matrix会报错

set.seed(16)

lasso\_fit <- cv.glmnet(x = data.matrix(marker\_exp),dep\_var,nfolds = 10,

family ="cox",alignment = "lambda")

plot(lasso\_fit)

#筛选变量

coefficient <- coef(lasso\_fit,s = lasso\_fit$lambda.min)

Active\_index <- which(as.numeric(coefficient)!=0)

active\_coefficients <- as.numeric(coefficient)[Active\_index]

si\_gene\_multi\_cox <- rownames(coefficient)[Active\_index]

si\_gene\_multi\_cox

######多因素multivariate-cox#####

#fit\_mul <- coxph(Surv(OS.time,OS)~CISD1+DDIT4+DECR1+GLS2+HERPUD1+OTUB1+PEBP1+PIR+PPP1R13L+YWHAE,

# x = T, y = T, data = Survival\_model)#原

#fit\_mul <- coxph(Surv(OS.time,OS)~CA9+GLS2+PEBP1+PIR+PPP1R13L+DECR1+OTUB1+GDF15+DDIT4,

# x = T, y = T, data = Survival\_model)#1se16多因素选p < 0.05加到cindex > 0.7，9个

fit\_mul <- coxph(Surv(OS.time,OS)~PIR+PEBP1+PPP1R13L+CA9+GLS2+DECR1+OTUB1+YWHAE,

x = T, y = T, data = Survival\_model)#min16，8个

sumfit <- summary(fit\_mul)

sumfit$conf.int

sumfit$coefficients

#Determine the C-index

c.index<-t(as.data.frame(sumfit$concordance))

c.index

Low95 <- (c.index[1]) - 1.96\*(c.index[2])

Upper95 <-(c.index[1]) + 1.96\*(c.index[2])

c.index<-cbind(c.index[1], Low95, Upper95)

c.index

#1se16:CA9+CISD1+DDIT4+DECR1+GCLC+GDF15+GLS2+HSF1+IL33+OTUB1+PEBP1+PIR+PPP1R13L+YWHAE

#min16:ALOX15+ANGPTL7+BCAT2+FTMT+HNF4A+CA9+CISD1+DDIT4+DECR1+GCLC+GDF15+GLS2+HSF1+IL33+OTUB1+PEBP1+PIR+PPP1R13L+YWHAE

#####生存曲线#####

library(rms)

library(survival)

hubgenes <- c("PIR","PEBP1","PPP1R13L","CA9","GLS2","DECR1","OTUB1","YWHAE")

pred.multinom <- predict(fit\_mul,

Survival\_model[,which(colnames(Survival\_model) %in% hubgenes)])

pred.multinom <- as.data.frame(pred.multinom)

pred.multinom$sample <- rownames(pred.multinom)

Survival\_group <- merge(dplyr::select(Survival\_model,c(1,2,4)),pred.multinom,by="sample")

#要有行名，会带着下面的都出现行名

rownames(Survival\_group) <- Survival\_group$sample

#取界值

library(survival)

library(survminer)

res.cut <- surv\_cutpoint(Survival\_group, time = "OS.time", event = "OS",

variables = "pred.multinom")

summary(res.cut)

#按界值分组

res.cat <- surv\_categorize(res.cut)

head(res.cat)

table(res.cat$pred.multinom)

#自行分组

Survival\_group <- dplyr::arrange(Survival\_group,pred.multinom)

Survival\_group$group <- "High Score"

Survival\_group$group[1:241] <- "Low Score"

#拟合生存曲线

fit\_cut <- survfit(Surv(OS.time, OS) ~group, data = Survival\_group)

#conf.int表示上下的可信区间

ggsurvplot(fit\_cut, data=Survival\_group, linetype = 1,

palette = c("#EE0000B2","#3B4992B2"),

size=1,surv.scale = c("percent"),pval = TRUE,legend.title = "",

legend.labs = c("High Scores", "Low Scores"),

break.time.by =12,

xlim = c(0,120),

risk.table = T,risk.table.title = "Patients at risk",

ylab = "Overall Survival, %",

xlab = "Months",font.x = c(20,"plain","black"),

font.y = c(20,"plain","black"),font.tickslab = c(20,"plain","black"),

risk.table.fontsize = 6.5,font.legend =c(25,"plain","black"),

font.main = c(20,"plain","black"),pval.size = 10)

GroupHigh <- rownames(Survival\_group)[which(Survival\_group$group=="High Score")]

GroupLow <- rownames(Survival\_group)[which(Survival\_group$group=="Low Score")]

#####GEO生存验证#####

library(survival)

library(survminer)

GEO\_Surv <- read.csv("D:/AaPaper/Date整理/GEO/LUAD/LUAD\_Ferro/3.csv")#6个队列分别验证

GEO\_Surv$X <- NULL

rownames(GEO\_Surv) <- GEO\_Surv$accession

pred\_GEO <- predict(fit\_mul,GEO\_Surv)

head(pred\_GEO)

#表达矩阵要有行名，predict函数行名会用表达矩阵的行名

pred\_GEO <- as.data.frame(pred\_GEO)

pred\_GEO$accession <- rownames(pred\_GEO)

GEO\_Surv <- merge(GEO\_Surv,pred\_GEO,by = "accession")

res.cut\_GEO <- surv\_cutpoint(GEO\_Surv, time = "time", event = "status",

variables = "pred\_GEO")

summary(res.cut\_GEO)

#按界值分组

res.cat\_GEO <- surv\_categorize(res.cut\_GEO)

head(res.cat\_GEO)

table(res.cat\_GEO$pred\_GEO)

#自行分组

GEO\_Surv <- dplyr::arrange(GEO\_Surv,pred\_GEO)

GEO\_Surv$group <- "High Score"

GEO\_Surv$group[1:29] <- "Low Score"

#拟合生存曲线

fit\_GEO <- survfit(Surv(time, status) ~group, data = GEO\_Surv)

#conf.int表示上下的可信区间

ggsurvplot(fit\_GEO, data=GEO\_Surv, linetype = 1,

palette = c("#EE0000B2","#3B4992B2"),

size=1,surv.scale = c("percent"),pval = TRUE,legend.title = "",

legend.labs = c("High Scores", "Low Scores"),

break.time.by =12,

xlim = c(0,120),

risk.table = T,risk.table.title = "Patients at risk",

ylab = "Overall Survival, %",

xlab = "Months",font.x = c(20,"plain","black"),

font.y = c(20,"plain","black"),font.tickslab = c(20,"plain","black"),

risk.table.fontsize = 6.5,font.legend =c(25,"plain","black"),

font.main = c(20,"plain","black"),pval.size = 10)

#####time RoC curve######

library(survival)

library(timeROC)

#GEO数据:GEO\_Surv

#TCGA数据:Survival\_group

ROC\_TCGA <- timeROC(T=Survival\_group$OS.time, #事件时间

delta=Survival\_group$OS, ##事件状态

marker=Survival\_group$pred.multinom,##默认marker 值越大。事件越可能发生，相反的话需要加负号

cause=1, #所关心的事件结局，死在事件是1

weighting="marginal", ##"marginal", "cox","aalen"分别是KM,cox,additive Aalen模型

times=c(12,24,36,48,60), ##1.2.3年ROC

ROC=TRUE) ##保存灵敏度和特异度的值

ROC\_TCGA

plot(ROC\_TCGA, time=12, col="#D86779", title=FALSE, lwd=2)

plot(ROC\_TCGA, time=24, col="#21A2A2",add=TRUE,title=FALSE, lwd=2)

plot(ROC\_TCGA, time=36, col="#E3B227", add=TRUE, title=FALSE, lwd=2)

legend(x=0.5, y=0.55,

c(paste0("AUC at 1 years: ", round (ROC\_TCGA$AUC[1],4)),

paste0("AUC at 2 years: ", round (ROC\_TCGA$AUC[2],4)),

paste0("AUC at 3 years: ", round (ROC\_TCGA$AUC[3],4))),

col=c("#D86779","#21A2A2","#E3B227") , lwd=2,bty="n")

#lty:line type。可以是数字或字符c(0="blank",1="solid" (default), 2 ="dashed" , 3 ="dotted",4= ）

#lwd: line width,默认是1

#bty:图例框是否输出，0为画出，默认为n不画出

GEO\_Surv <- read.csv("D:/AaPaper/Date整理/GEO/LUAD/LUAD\_Ferro/6.csv")#6个队列分别验证

GEO\_Surv$X <- NULL

rownames(GEO\_Surv) <- GEO\_Surv$accession

pred\_GEO <- predict(fit\_mul,GEO\_Surv)

head(pred\_GEO)

#表达矩阵要有行名，predict函数行名会用表达矩阵的行名

pred\_GEO <- as.data.frame(pred\_GEO)

pred\_GEO$accession <- rownames(pred\_GEO)

GEO\_Surv <- merge(GEO\_Surv,pred\_GEO,by = "accession")

ROC\_GEO <- timeROC(T=GEO\_Surv$time, #事件时间

delta=GEO\_Surv$status, ##事件状态

marker=GEO\_Surv$pred\_GEO,##默认marker 值越大。事件越可能发生，相反的话需要加负号

cause=1, #所关心的事件结局，死在事件是1

weighting="marginal", ##"marginal", "cox","aalen"分别是KM,cox,additive Aalen模型

times=c(12,24,36,48,60), ##1.2.3年ROC

ROC=TRUE) ##保存灵敏度和特异度的值

#GEO数据:GEO\_Surv

#TCGA数据:Survival\_cut

ROC\_GEO

plot(ROC\_GEO, time=12, col="#D86779", title=FALSE, lwd=2)

plot(ROC\_GEO, time=24, col="#21A2A2",add=TRUE,title=FALSE, lwd=2)

plot(ROC\_GEO, time=36, col="#E3B227", add=TRUE, title=FALSE, lwd=2)

legend(x=0.5, y=0.55,

c(paste0("AUC at 1 years: ", round (ROC\_GEO$AUC[1],4)),

paste0("AUC at 2 years: ", round (ROC\_GEO$AUC[2],4)),

paste0("AUC at 3 years: ", round (ROC\_GEO$AUC[3],4))),

col=c("#D86779","#21A2A2","#E3B227") , lwd=2,bty="n")

#####单基因生存#####

library(survminer)

Survival\_singlegene <- dplyr::select(Survival\_model,c(2,4,67,98,137,240,247,257,265,360))

Survival\_singlegene <- dplyr::arrange(Survival\_singlegene,OTUB1)

Survival\_singlegene$Group <- "High Expression"

Survival\_singlegene$Group[1:241] <- "Low Expression"

fit\_singlegene <- survfit(Surv(OS.time, OS) ~ Group, data = Survival\_singlegene)

ggsurvplot(fit\_singlegene, data=Survival\_singlegene, linetype = 1,

palette = c("#EE0000B2","#3B4992B2"),

size=1,surv.scale = c("percent"),pval = TRUE,legend.title = "",

legend.labs = c("High Expression","Low Expression"),

break.time.by =12,

xlim = c(0,120),title="OTUB1",

risk.table = F,risk.table.title = "Patients at risk",

ylab = "Overall Survival, %",

xlab = "Months",font.x = c(20,"plain","black"),

font.y = c(20,"plain","black"),font.tickslab = c(20,"plain","black"),

risk.table.fontsize = 6.5,font.legend =c(25,"plain","black"),

font.main = c(20,"plain","black"),pval.size = 10)

#####计算免疫评分的差异#####

score\_high <- Score[which(Score$ID %in% GroupHigh),]

score\_low <- Score[which(Score$ID %in% GroupLow),]

score\_long <- rbind(score\_high,score\_low)

score\_long <- score\_long %>% gather(key = scores,value = value,2:5)

score\_long$Group <- "High Score"

score\_long$Group[which(score\_long$ID %in% GroupLow)] <- "Low Score"

library(ggpubr)

ggplot(data = score\_long,aes(x = scores,y = value,fill = Group),

outlier.shape = NA)+

geom\_boxplot() + theme\_minimal()+

scale\_fill\_manual(values = c("#EE3A8C","#00B2EE"))+

stat\_compare\_means(label = "p.format",method = "t.test",hide.ns = T)+

theme(axis.text.x = element\_text(size = 12,angle = 0,hjust = 0.5,vjust = 0.5),

axis.title.x = element\_text(size = 15),

axis.title.y = element\_text(size = 15),legend.position = "right")

#小提琴图

library(ggpubr)

#1/2/3

ggviolin(score\_long[1:1446,],x = "scores",y = "value",color="Group",

xlab = "",ylab = "Scores",main="",

palette = c("#B10E0E","#10129E"),add = "boxplot",outlier.shape=NA)+

stat\_compare\_means(aes(group=Group),label = "p",

method = "t.test", hide.ns = T,

cex=5,hjust=0.5,vjust = -2)+

theme(axis.text.x = element\_text(angle = 0,hjust = 0.5,vjust = 1))

#Tumor Purity

ggviolin(score\_long[1447:1928,],

x = "Group",y = "value",color="Group",

xlab = "",ylab = "Tumor Purity",main="",

palette = c("#B10E0E","#10129E"),add="boxplot",outlier.shape=NA)+

stat\_compare\_means(aes(group=Group),label = "p",

method = "t.test", hide.ns = T,

cex=5,hjust=-1,vjust = -1)+

theme(axis.text.x = element\_text(angle = 0,hjust = 0.5,vjust = 1))

#Stromal Score

ggviolin(score\_long[1:482,],

x = "Group",y = "value",color="Group",

xlab = "",ylab = "Stromal Score",main="",

palette = c("#B10E0E","#10129E"),add="boxplot",outlier.shape=NA)+

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "t.test", hide.ns = T,

cex=10,hjust=-2.5,vjust = 0)+

theme(axis.text.x = element\_text(angle = 0,hjust = 0.5,vjust = 1))

#Immune Score

ggviolin(score\_long[483:964,],

x = "Group",y = "value",color="Group",

xlab = "",ylab = "Immune Score",main="",

palette = c("#B10E0E","#10129E"),add="boxplot",outlier.shape=NA)+

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "t.test", hide.ns = T,

cex=10,hjust=-2.5,vjust = 0)+

theme(axis.text.x = element\_text(angle = 0,hjust = 0.5,vjust = 1))

#Estimate Score

ggviolin(score\_long[965:1446,],

x = "Group",y = "value",color="Group",

xlab = "",ylab = "Estimate Score",main="",

palette = c("#B10E0E","#10129E"),add="boxplot",outlier.shape=NA)+

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "t.test", hide.ns = T,

cex=10,hjust=-2.5,vjust = 0)+

theme(axis.text.x = element\_text(angle = 0,hjust = 0.5,vjust = 1))

#####差异基因+lncRNA#####

library(edgeR)

#筛选编码蛋白基因

GeneType <- read.csv("D:/AaPaper/Date整理/GeneType.csv")

Protein\_Coding <- GeneType[which(GeneType$gene\_type =="protein\_coding"),]

lncName <- GeneType[which(GeneType$gene\_type == "lincRNA"),]

#去掉名字版本细节

lncName$gene\_name <- gsub("\\..\*$","",lncName$gene\_name)

#分组

#样本需要按分组排序

#Exp\_High <- LUAD\_TPM[,which(colnames(LUAD\_TPM) %in% GroupHigh)]

#Exp\_Low <- LUAD\_TPM[,which(colnames(LUAD\_TPM) %in% GroupLow)]

#LUAD\_DEGExp <- cbind(Exp\_High,Exp\_Low)

#edgR差异基因

LUAD\_Counts <- read.csv("D:/AaPaper/Date整理/TCGA/LUAD/LUAD\_Counts.csv")

LUAD\_Counts <- LUAD\_Counts %>% column\_to\_rownames("X")

#排序

Counts\_High <- LUAD\_Counts[,which(colnames(LUAD\_Counts) %in% GroupHigh)]

Counts\_Low <- LUAD\_Counts[,which(colnames(LUAD\_Counts) %in% GroupLow)]

LUAD\_Counts <- cbind(Counts\_High,Counts\_Low)

#edgR

group <- c(rep("1",241),rep("0",241))

design <- model.matrix(~group)

LUAD\_Counts <- 2^LUAD\_Counts-1 #还原数据

LUAD\_Counts <- as.matrix(LUAD\_Counts)

Counts\_pro <- LUAD\_Counts[which(rownames(LUAD\_Counts) %in% Protein\_Coding$gene\_name),]

y <- DGEList(counts = Counts\_pro,group = group)

y <- calcNormFactors(y)#标准化数据

y <- estimateCommonDisp(y)#普通离散度

y <- estimateTagwiseDisp(y)#基因间范围内的离散度

et <- exactTest(y,pair = c("1","0"))#精确检验

ordered\_tags <- topTags(et,n = 100000)

deg <- as.data.frame(ordered\_tags)

deg$Group <- "not-significant"

deg$Group[which((deg$PValue < 0.05) & (deg$logFC > 1))] <- "Upregulated"

deg$Group[which((deg$PValue < 0.05) & (deg$logFC < -1))] <- "Downregulated"

deg$logP<- -log10(deg$PValue)

table(deg$Group)#protein-coding mrna

#deg <- dplyr::arrange(deg,-logFC)#添加负号升序

write.csv(deg[deg$Group!="not-significant",],file = "Plot/PPI/deg.csv")

#火山图

library(ggpubr)

deg$ID <- rownames(deg)

deg <- deg[order(deg$PValue),]

up.genes <- head(deg$ID[which(deg$Group =="Upregulated")],10)

down.genes <- head(deg$ID[which(deg$Group =="Downregulated")],10)

deg.top10.genes <- c(as.character(up.genes),as.character(down.genes))

deg$Label = ""

deg$Label[match(deg.top10.genes,deg$ID)]<-deg.top10.genes

ggscatter(deg,x="logFC",y="logP",

color = "Group",

palette = c("#1f1fc8","gray","#D83131"),

label = deg$Label,

font.label = 10,

repel = T,

size = 2,

alpha = 0.7,

ylab = "-log10(P-Value)")+

geom\_hline(yintercept = 1.3010,linetype = "dashed") +

geom\_vline(xintercept = c(-1,1),linetype = "dashed")+

theme\_minimal()+

theme(axis.text.x = element\_text(size = 15, angle = 0, hjust = 0.5, vjust = 3.0),

axis.text.y = element\_text(size = 15),

axis.title.x = element\_text(size = 15),

axis.title.y = element\_text(size = 15),legend.position = "right")

#lncRNA

Counts\_lnc <- LUAD\_Counts[which(rownames(LUAD\_Counts) %in% lncName$gene\_name),]

#group <- c(rep("1",211),rep("0",271))

y2 <- DGEList(counts = Counts\_lnc,group = group)

y2 <- calcNormFactors(y2)#标准化数据

y2 <- estimateCommonDisp(y2)#普通离散度

y2 <- estimateTagwiseDisp(y2)#基因间范围内的离散度

et2 <- exactTest(y2,pair = c("1","0"))#精确检验

ordered\_tags2 <- topTags(et2,n = 100000)

delncRNA <- as.data.frame(ordered\_tags2)

delncRNA$Group = "not-significant"

delncRNA$Group[which((delncRNA$PValue<0.05) & (delncRNA$logFC > 1))] = "Upregulated" #up in HS group

delncRNA$Group[which((delncRNA$PValue<0.05) & (delncRNA$logFC < -1))] = "Downregulated"

delncRNA$logP<- -log10(delncRNA$PValue)

table(delncRNA$Group)

write.csv(delncRNA[delncRNA$Group!="not-significant",],file = "Data/DElncRNA.csv")

#火山图

library(ggpubr)

delncRNA$ID <- rownames(delncRNA)

delncRNA <- delncRNA[order(delncRNA$PValue),]

up.lncRNA <- head(delncRNA$ID[which(delncRNA$Group =="Upregulated")],10)

down.lncRNA <- head(delncRNA$ID[which(delncRNA$Group =="Downregulated")],10)

deg.top10.lncRNA <- c(as.character(up.lncRNA),as.character(down.lncRNA))

delncRNA$Label = ""

delncRNA$Label[match(deg.top10.lncRNA,delncRNA$ID)]<-deg.top10.lncRNA

ggscatter(delncRNA,x="logFC",y="logP",

color = "Group",

palette = c("#1f1fc8","gray","#D83131"),

label = delncRNA$Label,

font.label = 10,

repel = T,

size = 2,

alpha = 0.7,

ylab = "-log10(P-Value)")+

geom\_hline(yintercept = 1.3010,linetype = "dashed") +

geom\_vline(xintercept = c(-1,1),linetype = "dashed")+

theme\_minimal()+

theme(axis.text.x = element\_text(size = 15, angle = 0, hjust = 0.5, vjust = 3.0),

axis.text.y = element\_text(size = 15),

axis.title.x = element\_text(size = 15),

axis.title.y = element\_text(size = 15),legend.position = "right")

#####miRNA#####格式是log TPM，只能limma

library(limma)

miRNA <- read.csv("D:/AaPaper/Date整理/TCGA/LUAD/LUAD\_miRNA.csv")

rownames(miRNA) <- miRNA$X

miRNA <- miRNA[,which(colnames(miRNA) %in% rownames(datExpr))]

#排序

miRNA\_High <- miRNA[,which(colnames(miRNA) %in% GroupHigh)]

miRNA\_Low <- miRNA[,which(colnames(miRNA) %in% GroupLow)]

miRNA <- cbind(miRNA\_High,miRNA\_Low)

#分组

Group\_DEmiRNA <- data.frame("HS" = c(rep(1,208),rep(0,208)),"LS" = c(rep(0,208),rep(1,208)))

rownames(Group\_DEmiRNA)<-c(colnames(miRNA\_High),colnames(miRNA\_Low))

#差异分析

fit\_DEmiRNA <- lmFit(miRNA,Group\_DEmiRNA,method = "ls")#线性拟合

contrast.matrix <- makeContrasts(HS - LS,levels=Group\_DEmiRNA)#确定比较两组

fit\_DEmiRNA <- contrasts.fit(fit\_DEmiRNA,contrast.matrix)

fit\_DEmiRNA <- eBayes(fit\_DEmiRNA)#经验贝叶斯公式计算拟合标准误差

#p值矫正，提取差异分析结果，Toptable

all\_diff\_miRNA <- topTable(fit\_DEmiRNA, adjust.method = 'fdr',coef=1,p.value = 1,lfc <- log(1,2),number = 60000,sort.by = 'logFC')

all\_diff\_miRNA$logP<- -log10(all\_diff\_miRNA$P.Value)

all\_diff\_miRNA$ID<-rownames(all\_diff\_miRNA)

all\_diff\_miRNA$Group = "not-significant"

#将P.Val<0.05,logFC>0.5的基因设置为显著上调基因

#将P.Val<0.05,logFC<0.5的基因设置为显著下调基因

all\_diff\_miRNA$Group[which((all\_diff\_miRNA$P.Value < 0.05) & (all\_diff\_miRNA$logFC > 0.5))] = "Upregulated in HS group"

all\_diff\_miRNA$Group[which((all\_diff\_miRNA$P.Value < 0.05) & (all\_diff\_miRNA$logFC < -0.5))] = "Downregulated in HS group"

table(all\_diff\_miRNA$Group)

library(ggpubr)

#all\_diff\_miRNA$ID <- rownames(all\_diff\_miRNA)

all\_diff\_miRNA <- all\_diff\_miRNA[order(all\_diff\_miRNA$P.Value),]

up.miRNA <- head(all\_diff\_miRNA$ID[which(all\_diff\_miRNA$Group =="Upregulated in HS group")],10)

down.miRNA <- head(all\_diff\_miRNA$ID[which(all\_diff\_miRNA$Group =="Downregulated in HS group")],10)

deg.top10.miRNA <- c(as.character(up.miRNA),as.character(down.miRNA))

all\_diff\_miRNA$Label = ""

all\_diff\_miRNA$Label[match(deg.top10.miRNA,all\_diff\_miRNA$ID)]<-deg.top10.miRNA

ggscatter(all\_diff\_miRNA,x="logFC",y="logP",

color = "Group",

palette = c("#1f1fc8","gray","#D83131"),

label = all\_diff\_miRNA$Label,

font.label = 10,

repel = T,

size = 2,

alpha = 0.7,

ylab = "-log10(P-Value)")+

geom\_hline(yintercept = 1.3010,linetype = "dashed") +

geom\_vline(xintercept = c(-0.5,0.5),linetype = "dashed")+

theme\_minimal()+

theme(axis.text.x = element\_text(size = 15, angle = 0, hjust = 0.5, vjust = 3.0),

axis.text.y = element\_text(size = 15),

axis.title.x = element\_text(size = 15),

axis.title.y = element\_text(size = 15),legend.position = "right")

#####ceRNA network#####

NameDEmiRNAup <- all\_diff\_miRNA$ID[all\_diff\_miRNA$Group == "Upregulated in HS group"]

NameDEmiRNAup <- paste0("hsa-",NameDEmiRNAup)

NameDEmiRNAdown <- all\_diff\_miRNA$ID[all\_diff\_miRNA$Group == "Downregulated in HS group"]

NameDEmiRNAdown <- paste0("hsa-",NameDEmiRNAdown)

NameDEproteindown <- deg$ID[deg$Group=="Downregulated"]

NameDEproteinup <- deg$ID[deg$Group=="Upregulated"]

NameDElncdown <- delncRNA$ID[delncRNA$Group=="Downregulated"]

NameDElncup <- delncRNA$ID[delncRNA$Group=="Upregulated"]

##miRcode lncRNA

miRcode <- read.table("D:/AaPaper/LUAD/数据/mircode\_highconsfamilies.txt",

sep = "\t",header = T,quote = "",fill = T,comment.char = "!",

stringsAsFactors = FALSE)

miRcode <- dplyr::select(miRcode,1,2,3,4)

#table(miRcode$gene\_class)

miRcode\_uplnc <- miRcode[which(miRcode$gene\_symbol %in% NameDElncup),]

miRcode\_downlnc <- miRcode[which(miRcode$gene\_symbol %in% NameDElncdown),]

##multiMiR

library(multiMiR)

#联网检索，跟网速有关

mRNA\_miRNA <- get\_multimir(org = "hsa",target = NameDEprotein,

mirna = NameDEmiRNA,

table = "mirtarbase",#搜索的数据库，单个数据库或"validated","predicted","disease.drug","all"

summary = T)

mRNA\_miRNAcorr <- mRNA\_miRNA@data

mRNA\_miRNAsum <- mRNA\_miRNA@summary

#apply(mRNA\_miRNA@summary[,6:10],2,sum)

mirTarbase\_upmiRNA <- mRNA\_miRNAcorr[mRNA\_miRNAcorr$mature\_mirna\_id %in% NameDEmiRNAup,]

mirTarbase\_downmiRNA <- mRNA\_miRNAcorr[mRNA\_miRNAcorr$mature\_mirna\_id %in% NameDEmiRNAdown,]

######功能富集#####

###1.GO

# GO三大注释——BP:生物学过程 CC:细胞学组分 MF：分子生物学功能

#ENTREZID: 4312 8318

#SYMBOL: MMP1 CDC45

#ENSEMBL: ENSG00000196611 ENSG00000093009

library(clusterProfiler)

library(org.Hs.eg.db)

DEGname <- as.character(deg[deg$Group != "not-significant",]$ID)

#基因ID类型为ENSEMBL的ID形式，选择BP功能组(BP/CC/MF)

#c("PI3","LCE2A","SPRR2E","SPRR2A","SPRR2G","SPRR2B","LCE3D","LCE3E","RPTN")

Go\_resultBP <- enrichGO(DEGname,'org.Hs.eg.db',

keyType = "SYMBOL",

ont="BP",

pvalueCutoff=0.05)

Go\_resultCC <- enrichGO(DEGname, 'org.Hs.eg.db',

keyType = "SYMBOL",

ont="CC",

pvalueCutoff=0.05)

Go\_resultMF <- enrichGO(DEGname, 'org.Hs.eg.db',

keyType = "SYMBOL",

ont="MF",

pvalueCutoff=0.05)

#enrichplot::cnetplot(Go\_resultBP,showCategory = 10,circular = T,colorEdge = T) #B站看到的代码

barplot(Go\_resultBP, showCategory=10)

barplot(Go\_resultCC, showCategory=10)

barplot(Go\_resultMF, showCategory=10)

dotplot(Go\_resultBP, showCategory=10)

###2.KEGG(需要用ENTREZID形式的数据)

library(org.Hs.eg.db)

library(clusterProfiler)

#将基因名转换为ENTREZID格式

DEGname <- bitr(DEGname, fromType="SYMBOL", toType=c("ENTREZID"), OrgDb="org.Hs.eg.db")

DEGname <- DEGname$ENTREZID

enrich\_KEGG <- enrichKEGG(DEGname,

organism = "hsa",

pvalueCutoff = 0.05)

barplot(enrich\_KEGG, showCategory=10)

dotplot(enrich\_KEGG, showCategory=10)

#####免疫细胞ssGSEA#####

library(GSVA)

ImmuneGeneSet <- read.csv("D:/AaPaper/Date整理/geneset(immune).csv")

#gsvasig<-lapply(gsvasig, function(x) x[!is.na(x)])

#基因表达一个行是基因，列是样本的matrix

Immunecellresult <- GSVA::gsva(as.matrix(LUADFerro),ImmuneGeneSet,

method = "ssgsea",min.sz = 1,max.sz = Inf,

mx.diff=TRUE,verbose=FALSE, parallel.sz=0)

Immunecellresult <- as.data.frame(t(Immunecellresult))

#GSVA\_long <- cbind(rownames(GSVAresult),GSVAresult[,c(2,3,4,7,13,15,16,23,25,26)])

GSVA\_long <- cbind(rownames(Immunecellresult),Immunecellresult)

names(GSVA\_long)[1] <- "sample"

GSVA\_long$Group <- "High"

GSVA\_long$Group[which(GSVA\_long$sample %in% GroupLow)] <- "Low"

GSVA\_long <- tidyr::gather(GSVA\_long,key = Cell\_type,value = Proportion,2:25)

library(ggpubr)

ggpubr::ggboxplot(GSVA\_long,outlier.shape = NA,

x = "Cell\_type",y = "Proportion",

color = "black",fill = "Group",

group = "Group",size = 0.3,palette = c("#E741AD","#1CAAC6"),

xlab = "",ylab = "Enrichment Score") +

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "t.test",hide.ns = T,cex=5)+

theme(axis.text.x = element\_text(angle = 60,hjust = 1,vjust = 1,size = 7.5))

#相互关系

library(ggplot2)

library(ggcorrplot)

library(ggthemes)

Immune\_corr<-as.data.frame(round(cor(Immunecellresult,method = "spearman"),3))##默认pearson

Immune\_p.mat<-as.data.frame(cor\_pmat(Immunecellresult))

ggcorrplot(Immune\_corr,hc.order=TRUE,hc.method="complete",

outline.col='white',ggtheme = theme\_bw(),type = "upper",

lab=TRUE,lab\_size=1.5,p.mat=Immune\_p.mat,insig="blank",tl.cex = 7.5)

#tl.cex坐标轴文本大小

#####免疫检查点#####

ImmuneCP <- c("IDO1","CTLA4","TNFRSF9","ICOS","CD80","TIGIT","CD70","TNFSF9",

"CD86","POCD1","LAIR1","TNFRSF8","TNFSF15","TNFRSF14","CD276","CD40",

"TNFRSF4","TNFSF14","HLA2","CD244","CD274","HAVCR2","CD27","BTLA",

"LGALS9","CD28","CD48","TNFRSF25","CD40LG","VTCN1","CD160","CD44",

"TNFSF18","TNFRSF18","BTNL2","CD200R1","TNFSF4","CD200","NRP1")

CP\_exp <- LUADFerro[rownames(LUADFerro) %in% ImmuneCP,]

CP\_exp <- as.data.frame(t(CP\_exp))

CP\_exp <- cbind(rownames(CP\_exp),CP\_exp)

names(CP\_exp)[1] <- "sample"

CP\_exp$Group <- "High Score"

CP\_exp$Group[which(CP\_exp$sample %in% GroupLow)] <- "Low Score"

CP\_exp <- tidyr::gather(CP\_exp,key = Gene\_type,value = Expression,2:38)

ggpubr::ggboxplot(CP\_exp,outlier.shape = NA,

x = "Gene\_type",y = "Expression",

color = "black",fill = "Group",

group = "Group",size = 0.3,palette = c("#E741AD","#1CAAC6"),

xlab = "",ylab = "Gene Expression") +

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "t.test",hide.ns = T,cex=4)+

theme(axis.text.x = element\_text(angle = 70,hjust = 1,vjust = 1,size = 12))

#####KEGG######

library(GSEABase)

KEGGSet <- getGmt("D:/AaPaper/Date整理/c2.KEGG.gmt")

#GSVA分析

library(GSVA)

kegg <- gsva(expr=as.matrix(LUADFerro), KEGGSet,

method = "ssgsea",min.sz = 1,max.sz = Inf,

mx.diff=TRUE,verbose=FALSE, parallel.sz=0)

#分组

GroupKEGG <- ifelse(colnames(kegg) %in% GroupHigh, "High", "Low")

GroupKEGG <- factor(GroupKEGG,levels = c("High","Low"))

#差异分析

Design\_KEGG <- model.matrix(~GroupKEGG)

colnames(Design\_KEGG) <- levels(GroupKEGG)

fit\_KEGG <- lmFit(kegg,Design\_KEGG)

fit\_KEGG <- eBayes(fit\_KEGG)

KEGGDiff <- topTable(fit\_KEGG,adjust='fdr',coef=2,number=Inf)

KEGGDiff <- KEGGDiff[KEGGDiff$adj.P.Val < 0.05,]

KEGGDiffdown <- KEGGDiff[KEGGDiff$logFC > 0,]

KEGGDiffup <- KEGGDiff[KEGGDiff$logFC < 0,]

library(pheatmap)

pheatmap(kegg[rownames(kegg) %in% rownames(KEGGDiffup),],

cluster\_cols = F,cluster\_rows = T,

show\_colnames = F,show\_rownames = T,

annotation\_col = dplyr::select(Survival\_group,5),scale = "row",

treeheight\_row = 30,

color=c(colorRampPalette(colors=c("blue","white"))(length(bk)/2)

,colorRampPalette(color=c("white","red"))(length(bk)/2)),

legend\_breaks = seq(-1,1,1),breaks = bk)

#####Hallmark gene set#####

library(GSVA)

HallmarkGeneSet <- read.csv("D:/AaPaper/Date整理/Hallmark geneset.csv")

#gsvasig<-lapply(gsvasig, function(x) x[!is.na(x)])

#基因表达一个行是基因，列是样本的matrix

GSVAresult <- GSVA::gsva(as.matrix(LUADFerro),HallmarkGeneSet,

method = "ssgsea",min.sz = 1,max.sz = Inf,

mx.diff=TRUE,verbose=FALSE, parallel.sz=0)

GSVAresult <- as.data.frame(t(GSVAresult))

hall\_p <- as.data.frame(matrix(nrow = 50,ncol = 2))

names(hall\_p) <- c("geneset","pvalue")

for (i in 1:50){

hallHigh <- hall\_heat[i,which(colnames(hall\_heat)%in% GroupHigh)]

hallLow <- hall\_heat[i,which(colnames(hall\_heat)%in% GroupLow)]

ttest <- t.test(hallHigh,hallLow)

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

hall\_p$geneset[i] <- rownames(hall\_heat)[i]

hall\_p$pvalue[i] <- ttest$p.value

}

hall\_pname <- hall\_p$geneset[hall\_p$pvalue < 0.05]

hall\_heat <- as.data.frame(t(GSVAresult))

hall\_heat <- hall\_heat[rownames(hall\_heat) %in% hall\_pname,]

rownames(hall\_heat) <- gsub("HALLMARK\_","",rownames(hall\_heat))

bk <- c(seq(-1,0,by=0.01),seq(0.01,1,by=0.01)) #色度条调节

library(pheatmap)

pheatmap(hall\_heat,cluster\_cols = F,cluster\_rows = T,

show\_colnames = F,show\_rownames = T,

annotation\_col = dplyr::select(Survival\_group,5),scale = "row",

treeheight\_row = 30,

color=c(colorRampPalette(colors=c("blue","white"))(length(bk)/2)

,colorRampPalette(color=c("white","red"))(length(bk)/2)),

legend\_breaks = seq(-1,1,1),breaks = bk)

#GSVA\_long <- cbind(rownames(GSVAresult),GSVAresult[,c(2,3,4,7,13,15,16,23,25,26)])

hall\_long <- cbind(rownames(GSVAresult),GSVAresult)

names(hall\_long)[1] <- "sample"

hall\_long$Group <- "High Score"

hall\_long$Group[which(hall\_long$sample %in% GroupLow)] <- "Low Score"

hall\_long <- tidyr::gather(hall\_long,key = Gene\_Set,value = Proportion,2:51)

hall\_long$Gene\_Set <- gsub("HALLMARK\_","",hall\_long$Gene\_Set)

hall\_long <- hall\_long[which(hall\_long$Gene\_Set %in% hall\_pname),]

library(ggpubr)

ggboxplot(hall\_long,outlier.shape = NA,

x = "Gene\_Set",y = "Proportion",

color = "black",fill = "Group",

group = "Group",size = 0.3,palette = c("#E741AD","#1CAAC6"),

xlab = "",ylab = "Enrichment Score") +

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "t.test",hide.ns = T,cex=5)+

theme(axis.text.x = element\_text(angle = 30,hjust = 1,vjust = 1,size = 10))

######GSEA######

deg\_GSEA <- deg[which(deg$Group!="not-significant"),]

deg\_GSEA <- dplyr::arrange(deg\_GSEA,logFC)#排序

deg\_GSEA <- dplyr::select(deg\_GSEA,1,7)#logFC和ID列

library(clusterProfiler)

zz <- bitr(deg\_GSEA$ID,fromType = "SYMBOL",toType = "ENTREZID",OrgDb = "org.Hs.eg.db")

names(deg\_GSEA)[2]<-"SYMBOL"

zz <- merge(deg\_GSEA,zz,by="SYMBOL",all=F)

zz <- zz[order(zz$logFC,decreasing = T),] #必须是从大到小

#用于分析的z1为向量，，内容是logFC,名字是etrezid,从大到小排序

z1 <- zz$logFC

names(z1)<-zz$ENTREZID

z2 <- gseKEGG(z1,organism = "hsa")

z3 <- z2[order(z2$enrichmentScore,decreasing=T),]

library(enrichplot)

dotplot(z2,showCategory=20)

gseaplot2(z2,row.names(z3)[1],subplots = 1:2,#显示画几部分

base\_size = 20,pvalue\_table = F,color = "orange",ES\_geom = "line")

#####免疫浸润#####

library(IOBR)

TPM\_data <- LUAD\_TPM[,which(names(LUAD\_TPM) %in% rownames(datExpr))]

ImmuneXcell <- deconvo\_tme(eset = 2^TPM\_data-1,method = "xcell",arrays = F)#数据必须non-log scale

ImmuneMCP <- deconvo\_tme(eset = 2^TPM\_data-1,method = "mcpcounter")#数据必须non-log scale

ImmuneCibersort <- deconvo\_tme(eset = 2^TPM\_data-1,method = "cibersort",

arrays = F,perm = 200)#数据必须non-log scale

ImmuneEpic <- deconvo\_tme(eset = 2^TPM\_data-1,method = "epic",tumor = T)#数据必须non-log scale

Immune\_Analyse <- dplyr::select(Survival\_group,1:3)

names(Immune\_Analyse)[1] <- "ID"

Immune\_Analyse <- merge(Immune\_Analyse,ImmuneMCP[,1:11],by="ID")

#names(Immune\_Analyse)[4:13] <- gsub("\_MCPcounter","",names(Immune\_Analyse)[4:13])

#Immune Gene Set

Immune\_cell <- cbind(rownames(Immunecellresult),Immunecellresult)

names(Immune\_cell)[1] <- "ID"

Immune\_Analyse <- merge(Immune\_Analyse,Immune\_cell,by="ID")

library(survival)

immuneCOX<-function(x){

FML <- as.formula(paste0("Surv(OS.time,OS)~",x))

cox <- coxph(FML, data=Immune\_Analyse)

sum <- summary(cox)

sum$coefficients

HR <- round(sum$coefficients[,2],2)

pvalue <- round((sum$coefficients[,5]),3)

CI <- paste0(round(sum$coefficients[,2],2),"(",paste0(round(sum$conf.int[,3:4],2),collapse = "-"),")")

upper <- round(sum$conf.int[,4],2)

lower <- round(sum$conf.int[,3],2)

unicox <- data.frame("characteristics"=x,

"Hazard Ratio"=HR,

"CI95"=CI,

"pvalue"=pvalue,

"upper"=upper,

"lower"=lower)

unicox$Variance<-rownames(unicox)

return(unicox)

}

immuneCOX("Bcells\_EPIC")

library(plyr)

#names(Immune\_Analyse) <- gsub("\\+","",names(Immune\_Analyse)) #xcell

#names(Immune\_Analyse) <- gsub("\\(","",names(Immune\_Analyse)) #cibersort

Immune\_var <- lapply(names(Immune\_Analyse)[4:27],immuneCOX)

Immune\_var <- ldply(Immune\_var,data.frame)

#Immune\_var <- Immune\_var[c(4,10,15,20,29,34),]

Immune\_var<- Immune\_var[-7,]

#森林图

Immune\_label <- select(Immune\_var,c(1,4,3))

#Immune\_label$characteristics <- gsub("\_EPIC","",Immune\_label$characteristics)

#Immune\_label$characteristics <- gsub("\_xCell","",Immune\_label$characteristics)

library(forestplot)

forestplot(Immune\_label,#文本部分的数据

mean=Immune\_var$Hazard.Ratio,#指定HR

lower=Immune\_var$lower, #指定下区间

upper=Immune\_var$upper, #指定上区间

zero = 1, #参照竖线取值，一般取在1

lwd.zero=2, ##参照线粗细

lwd.ci=3, ##可信区间线的粗细

lwd.xaxis=2, ##x轴的粗细

boxsize=0.2, ##线中间的方块大小

graph.pos = 3, #设置森林图出现在在表格中的位置

xlab = "", #设置x轴的字

clip=c(0,2),xticks=seq(0,2,by=0.5), #可信区间控制在箭头范围内

lineheight = "auto", ##自动行距

txt\_gp = fpTxtGp(ticks = gpar(fontsize =25),xlab = gpar(fontsize =10),label = gpar(fontsize =15)),

col=fpColors(box="#FA7F6F",zero = "#E7DAD2",line="#FFBE7A"))

#森林图（不好看）

library(finalfit)

Immune\_fit <- finalfit(Immune\_Analyse,names(Immune\_Analyse)[3],names(Immune\_Analyse)[4:10])

ff\_plot(Immune\_Analyse,names(Immune\_Analyse)[3],names(Immune\_Analyse)[4:10])

#####TIDEscore####

TIDE\_expression <- LUADFerro

TIDE\_expression$mean <- apply(TIDE\_expression,1,mean)

TIDE\_expression <- TIDE\_expression-TIDE\_expression$mean

write.csv(TIDE\_expression,file = "TIDEexpression.csv")

TIDE\_score <- read.csv("Data/TIDEscore.csv")

TIDE\_score <- dplyr::select(TIDE\_score,c(1,4))

TIDE\_score$Group <- "High Score"

TIDE\_score$Group[TIDE\_score$Patient %in% GroupLow] <- "Low Score"

library(ggpubr)

ggviolin(TIDE\_score,x = "Group",y = "TIDE",

color = "black",draw\_quantiles = T,ylab = "TIDE Score",xlab = "",

alpha = 0.5,palette = "npg",size = 0.5,fill = "Group",

add = c("boxplot")) +theme\_classic2()+

stat\_compare\_means(method = "t.test",label = "p.signif",

hide.ns = T,size=6.5,vjust = 0.5,

comparisons = combn(unique(TIDE\_score$Group),2,simplify = F))+

scale\_x\_discrete(limits = c("High Score","Low Score"))+

theme(text = element\_text(size = 20))

#####Cibersort#####

library(ggplot2)

library(ggcorrplot)

library(ggthemes)

Cibersort\_corr<-as.data.frame(round(cor(ImmuneCibersort[,c(2:5,7:23)]),3),method="pearson") ##默认pearson

Cibersort\_p.mat<-as.data.frame(cor\_pmat(ImmuneCibersort[,c(2:5,7:23)]))

ggcorrplot(Cibersort\_corr,hc.order=TRUE,hc.method="complete",

outline.col='white',ggtheme = theme\_bw(),type = "upper",

lab=TRUE,lab\_size=1.5,p.mat=Cibersort\_p.mat,insig="blank",tl.cex = 7.5)

#tl.cex坐标轴文本大小

#####主成分分析#####

model\_exp <- Survival\_model[,which(colnames(Survival\_model) %in% hubgenes)]

write.csv(t(model\_exp),file = "Data/modelexp.csv")

#####基因突变######

#####maf文件里核心文件是data,其他是衍生的

library(maftools)

LUADmaf = read.maf(maf = "D:/AaPaper/Date整理/TCGA/LUAD/TCGA-LUAD.maf")

getSampleSummary(LUADmaf)

getGeneSummary(LUADmaf)

datas<-LUADmaf@data

datas<-as.data.frame(datas)

datas$Tumor\_Sample\_Barcode<-gsub('-','\_',datas$Tumor\_Sample\_Barcode)

##正则表达式去除多余的ID部分

datas$Tumor\_Sample\_Barcode<-gsub('............$','',datas$Tumor\_Sample\_Barcode)

##先把data拿出来修改再放回去

data\_high<-datas[datas$Tumor\_Sample\_Barcode %in% GroupHigh,]

data\_low<-datas[datas$Tumor\_Sample\_Barcode %in% GroupLow,]

data\_high\_maf<-read.maf(data\_high)

data\_low\_maf<-read.maf(data\_low)

#####比较两组间差异

mutation\_compare <- mafCompare(m1 = data\_high\_maf, m2 = data\_low\_maf,

minMut = 5,#只纳入最少n个样本发生突变的基因

m1Name = 'High Score', m2Name = 'Low Score')

print(mutation\_compare)

LUAD\_diff\_mut<-mutation\_compare$results[mutation\_compare$results$pval<0.05,]

mute\_gene <- head(LUAD\_diff\_mut$Hugo\_Symbol,20)

#展示重点变量的总结信息

#Shows sample summry.

getSampleSummary(data\_high\_maf)

#Plot summarizision

plotmafSummary(maf = data\_high\_maf, rmOutlier = TRUE, addStat = 'median',

dashboard = TRUE, titvRaw = FALSE)

plotmafSummary(maf = data\_low\_maf, rmOutlier = TRUE, addStat = 'median',

dashboard = TRUE, titvRaw = FALSE)

#Summarize Transition and Transversions，转换和颠换的占比，SNP

data\_low\_maf\_titv = titv(maf = data\_low\_maf, plot = FALSE, useSyn = TRUE)

data\_high\_maf\_titv = titv(maf = data\_high\_maf, plot = FALSE, useSyn = TRUE)

#plot titv summary

plotTiTv(res = data\_low\_maf\_titv)

plotTiTv(res = data\_high\_maf\_titv)

#Lollipop plots，需要有Protein\_change这一列，即AACol

#每个图对应一个特定的基因的氨基酸改变情况

lollipopPlot(maf = LUADmaf, gene = 'TP53', AACol = 'HGVSp\_Short', showMutationRate = TRUE)

#瀑布图

RColorBrewer::display.brewer.all() ##查看调色板

vc\_cols = RColorBrewer::brewer.pal(n = 8, name = 'Paired')

names(vc\_cols) = c('Frame\_Shift\_Del','Missense\_Mutation',

'Nonsense\_Mutation','Multi\_Hit','Frame\_Shift\_Ins','In\_Frame\_Ins',

'Splice\_Site','In\_Frame\_Del')

oncoplot(maf=data\_high\_maf,#top=20,

genes = mute\_gene,keepGeneOrder = F,

#top=20可以换成gene=想要的基因，KeepGeneOrder=T默认为False

fontSize = 0.8,#colors = vc\_cols,

showTumorSampleBarcodes = F,draw\_titv=F,legend\_height =6,anno\_height =5,

titleFontSize = 1.6,legendFontSize =2,barcode\_mar = 0.1,gene\_mar = 10)

oncoplot(maf=data\_low\_maf,#top = 20,

genes = mute\_gene,keepGeneOrder = F,

fontSize = 0.8,#colors = vc\_cols,

showTumorSampleBarcodes = F,draw\_titv=F,legend\_height =6 ,anno\_height =5,

titleFontSize = 1.6,legendFontSize =2,barcode\_mar = 0.1,gene\_mar = 10)

#######风险因子累积图#####

#Risk factor风险因子关联图

library(ggplotify)

library(ggplot2)

#biomarker\_data整合了生存信息+Score+Group

biomarker\_data <- Survival\_group

biomarker\_data <- biomarker\_data %>% remove\_rownames() %>%column\_to\_rownames("sample")

names(biomarker\_data)[3] <- "Score"

#biomarker\_risk <- biomarker\_data

#1.riskscore高低风险分组显示点图

biomarker\_data<-biomarker\_data[order(as.numeric(biomarker\_data$Score)),]

table(biomarker\_data$group)

p1=ggplot(data=biomarker\_data,aes(x=seq(0,481),y=Score,color=group))+

geom\_point()+

scale\_x\_continuous(breaks=seq(0,481,100))+#x轴范围及间隔,样本数，??必须减1

scale\_y\_continuous(breaks=seq(-2,2.5,1))+#y轴范围及间隔，评分

geom\_hline(aes(yintercept=-0.0325),colour="#BB0000",linetype="dashed")+#y轴标线，评分界值

geom\_vline(aes(xintercept=241),colour="#BB0000",linetype="dashed")+#x轴标线

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

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

panel.background = element\_blank(),

axis.line=element\_line(colour = "black"))+

labs(x="",y="Score")+

theme(axis.line = element\_line(size=1, colour = "black"),

axis.text.x = element\_text(size = 20,hjust = 0.5,vjust = 0.5),

axis.text.y = element\_text(size = 20,hjust = 0.5,vjust = 0.5),

axis.title.x = element\_text(size = 23),

axis.title.y = element\_text(size = 23),legend.position = "right")

p1

#2.生存状态散点图

biomarker\_data$OS=factor(biomarker\_data$OS,levels=c("0","1"),labels=c("Alive","Dead"))

#分组排序一下？

p2=ggplot(data=biomarker\_data)+

geom\_point(aes(x=seq(0,481),y=OS.time,color= OS)) +

scale\_x\_continuous(breaks=seq(0,482,100)) +

scale\_y\_continuous(breaks=seq(0,250,50)) +

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

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

panel.background = element\_blank(),

axis.line = element\_line(colour = "black"),

axis.text.x = element\_text(size = 20,hjust = 0.5,vjust = 0.5),

axis.text.y = element\_text(size = 20,hjust = 0.5,vjust = 0.5),

axis.title.x = element\_text(size = 23),

axis.title.y = element\_text(size = 23),legend.position = "right")+

labs(x="",y="Follow up months")+

theme(axis.line = element\_line(size=1, colour = "black"))+

geom\_vline(aes(xintercept=241),colour="#BB0000",linetype="dashed")

p2

#3. p3是热图

#hubGene

library(pheatmap)

Group <- Group\_DEG

Group$Group <- "High Score"

Group$Group[which(Group$LS ==1)] <- "Low Score"

hubgene\_Exp <- as.data.frame(t(datExpr))

hubgene\_Exphi <- hubgene\_Exp[,which(colnames(hubgene\_Exp) %in% GroupHigh)]

hubgene\_Explo <- hubgene\_Exp[,which(colnames(hubgene\_Exp) %in% GroupLow)]

hubgene\_Exp <- cbind(hubgene\_Exphi,hubgene\_Explo)

hubgene\_Exp <- hubgene\_Exp[which(rownames(hubgene\_Exp) %in% hubgenes),]

bk <- c(seq(-2,0,by=0.01),seq(0.01,2,by=0.01)) #色度条调节

p3=pheatmap(hubgene\_Exp,cluster\_cols = F,cluster\_rows = T,

show\_colnames = F,show\_rownames = T,

annotation\_col = dplyr::select(Survival\_group,5),scale = "row",

treeheight\_row = 30,

color=c(colorRampPalette(colors=c("blue","white"))(length(bk)/2)

,colorRampPalette(color=c("white","red"))(length(bk)/2)),

legend\_breaks = seq(-2,2,1),breaks = bk)

p3=ggplotify::as.ggplot(p3)

p3

#三图合并大法

library(cowplot)

plot\_grid(p1,p2,ncol = 1,axis='l',align='v')

#####药物敏感性分析#####

library(data.table)

Drugsensitive <- fread('D:/AaPaper/Date整理/TCGA/LUAD/LUAD-WGCNA-Ferro药敏//DrugPredictions.csv', data.table = F)

rownames(Drugsensitive) <- Drugsensitive$V1

names(Drugsensitive)[1] <- 'sample'

#排序

Drug\_high <- Drugsensitive[which(rownames(Drugsensitive) %in% GroupHigh),]

Drug\_low <- Drugsensitive[which(rownames(Drugsensitive) %in% GroupLow),]

Drugsensitive <- rbind(Drug\_high,Drug\_low)

#多组小提琴图计算P值

pvalue <- as.data.frame(matrix(nrow = 198,ncol = 2))

names(pvalue) <- c("chemical","pvalue")

for (i in 2:199){

DrugtestHigh <- Drugsensitive[1:241,i]

DrugtestLow <- Drugsensitive[242:482,i]

wilcoxtest <- wilcox.test(DrugtestHigh,DrugtestLow)

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

pvalue$chemical[i-1] <- names(Drugsensitive)[i]

pvalue$pvalue[i-1] <- wilcoxtest$p.value

}

pvalue$drug <- gsub("\_.\*$","",pvalue$chemical)

#分Pvalue

pvalue0.05 <- pvalue$chemical[which(pvalue$pval < 0.05 & pvalue$pval >= 0.01)]

pvalue0.01 <- pvalue$chemical[which(pvalue$pval < 0.01 & pvalue$pval >= 0.001)]

pvalue0.001 <- pvalue$chemical[which(pvalue$pval < 0.001)]

#分批

chemical1 <- Drugsensitive[,which(colnames(Drugsensitive) %in% pvalue0.05)]

colnames(chemical1) <- gsub("\_.\*$","",colnames(chemical1))

chemical2 <- Drugsensitive[,which(colnames(Drugsensitive) %in% pvalue0.01)]

colnames(chemical2) <- gsub("\_.\*$","",colnames(chemical2))

chemical3 <- Drugsensitive[,which(colnames(Drugsensitive) %in% pvalue0.001)]

colnames(chemical3) <- gsub("\_.\*$","",colnames(chemical3))

####假设rt：行是样本名字，列是细胞系

library(tidyr)

library(tidyverse)

#rt1 <- chemical1[,c(1,2,5,7,9,10,14,15,16,18,19,21)]

#0.1:c(4),1:c(3,6),60:c(1,2),200:c(7),300:c(),800:c(5,10)

#rt1 <- chemical2[,c(1,3,6,7,8,10,12,15)]

#rt1 <- chemical3[,c(1,2,3,4,7,8,10,11,12,13,14,16,24,25,27,34,35,36,37,39,43,44,47)]

#rt <- chemical3[,c(3,7,16,43)]#指南 IC60

rt <- cbind(chemical2[,1],chemical3[,c(1,13,14,34,35,47)])

#names(rt)[1] <- "Staurosporine"

rt$sample <- rownames(rt)

rt$Group <- "High"

rt$Group[which(rownames(rt) %in% GroupLow)] <- "Low"

rt <- rt %>% gather(key = chem,value = IC50,1:7)

library(ggpubr)

ggplot(rt,aes(x = Group,y = IC50,fill = Group))+

guides(fill = guide\_legend(title = "Group"))+

labs(x = "", y = "IC50")+

geom\_violin(alpha = 5,aes(linetype = NA))+

#scale\_fill\_manual(values = c("#E4C9DC","#61AEDC"))+ #填充颜色

facet\_wrap(~chem,nrow =2)+theme\_bw()+ylim(0,0.5)+

stat\_compare\_means(aes(group=Group),label = "p.signif",

method = "wilcox.test",hide.ns = T,cex =7.5,

hjust= 0,vjust = 1)+

geom\_boxplot(width=0.2,cex=0.8,position=position\_dodge(0.8),outlier.shape = NA)+

theme(axis.text.x = element\_text(angle = 0, hjust = 0.5))

#####clinical characteristic#####

#单因素

library(survival)

library(survminer)

Survival\_Clinical <- LUAD\_Clinical[LUAD\_Clinical$submitter\_id.samples %in% rownames(datExpr),]

Survival\_Clinical <- Survival\_Clinical[,c(1,6,42,43,44,78,98)]

names(Survival\_Clinical) <- c("sample","Age","M","N","T","Gender","Stage")

Survival\_Clinical <- merge(Survival\_Clinical,Survival\_group,by = "sample")

Survival\_Clinical$Stage <- gsub("stage ",'',Survival\_Clinical$Stage)

Survival\_Clinical$M <- gsub("NA","",Survival\_Clinical$M)

Survival\_Clinical$T <- gsub("b","",Survival\_Clinical$T)

#####Cox回归~各变量中不显示的类型就是对照reference

##等级资料排序,cox回归的对照组就是等级最低的那一组

#factor(bb$Stage,ordered = F,levels = c(这里面从小到大排序))

Survival\_Clinical$M <- factor(Survival\_Clinical$M,ordered = F,levels = c("M0","M1","MX"))

Survival\_Clinical$N <- factor(Survival\_Clinical$N,ordered = F,levels = c("N0","N1","N2","N3"))

Survival\_Clinical$T <- factor(Survival\_Clinical$T,ordered = F,levels = c("T1","T2","T3","T4"))

Survival\_Clinical$Gender <- factor(Survival\_Clinical$Gender,ordered = F,levels = c("male","female"))

Survival\_Clinical$Stage <- factor(Survival\_Clinical$Stage,ordered = F,levels = c("I","II","III","IV"))

Survival\_Clinical$group <- factor(Survival\_Clinical$group,ordered = F,levels = c("Low Score","High Score"))

#Univariate (可信区间为0-infinite可能是对照组样本量太少)

library(survival)

characteritics <- Survival\_Clinical

characteritics <- within(characteritics,{

Tstage <- NA

Tstage[T == "T1" | T == "T2"] = "T1/2"

Tstage[T == "M3" | T == "T4"] = "T3/4"

#Mstage[M == "M1" | M == "MX"] = "M1/X"

#Mstage[M == "M0"] = "M0"

#Nstage[N == "N1" | N == "N2" | N=="N3"] = "N1/2/3"

#Nstage[N=="N0"]="N0"

#stage[Stage == "I" | Stage == "II"] = "Stage I-II"

#stage[Stage == "III" | Stage == "IV"] = "Stage III-IV"

})

characteritics$T<-NULL

#循环函数构建所有变量

UNICOX<-function(x){

FML <- as.formula(paste0("Surv(OS.time,OS)~",x))

Clinical\_cox <- coxph(FML, data=Survival\_Clinical)

Clinical\_sum <- summary(Clinical\_cox)

Clinical\_sum$coefficients

HR <- round(Clinical\_sum$coefficients[,2],2)

pvalue <- round((Clinical\_sum$coefficients[,5]),3)

CI <- paste0(round(Clinical\_sum$conf.int[,3:4],2),collapse = "-")

unicox <- data.frame("characteristics"=x,

"Hazard Ratio"=HR,

"CI95"=CI,

"pvalue"=pvalue)

unicox$Variance<-rownames(unicox)

return(unicox)

}

UNICOX("Age")

library(plyr)

Clinical\_var <- lapply(names(Survival\_Clinical)[c(2:7,10)],UNICOX)

Clinical\_var <- ldply(Clinical\_var,data.frame)

Clinical\_var #二分类变量的对照组需要从Sexsum里看，没有显示的那个是对照

write.csv(Clinical\_var,file = "Plot/nomogram+calibration curve/univ.csv")

#多因素

fit\_clinical <- coxph(Surv(OS.time,OS)~pred.multinom+Stage+Age+Gender,

x = T, y = T, data=Survival\_Clinical)

sum\_clinical <- summary(fit\_clinical)

sum\_clinical$coefficients

sum\_clinical$conf.int

######subgroup#####

library(ggpubr)

subgroup <- LUAD\_Clinical[LUAD\_Clinical$submitter\_id.samples %in% rownames(datExpr),]

subgroup <- subgroup[,c(1,6,42,43,44,78,98)]

names(subgroup) <- c("sample","Age","M","N","T","Gender","Stage")

subgroup <- merge(Survival\_group,subgroup,by = "sample")

subgroup$Stage <- gsub("b",'',subgroup$Stage)

subgroup$M <- gsub("b","",subgroup$M)

subgroup$T <- gsub("a","",subgroup$T)

write.csv(subgroup,file = "clinical.csv")

subgroup2 <- read.csv("clinical.csv")

subgroup2$Stage[!is.na(subgroup2$Stage)] <- paste0("Stage",subgroup2$Stage[!is.na(subgroup2$Stage)])

subgroup2$Age65 <- "Age>62"

subgroup2$Age65[subgroup2$Age<63] <- "Age<=62"

ggviolin(subgroup2[!is.na(subgroup2$Gender),],x = "Gender",y = "pred.multinom",

color = "black",draw\_quantiles = T,ylab = "Score",xlab = "",

alpha = 0.5,palette = "npg",size = 0.5,fill = "Gender",

add = c("boxplot")) +theme\_classic2()+

stat\_compare\_means(method = "t.test",label = "p.signif",

hide.ns = T,size=6.5,vjust = 0.5,

comparisons = combn(unique(subgroup2[!is.na(subgroup2$Gender),]$Gender),2,simplify = F))+

#scale\_x\_discrete(limits = c("M0","M1","MX"))+

theme(text = element\_text(size = 20))

#c("T1","T2","T3","T4")

#c("N0","N1","N2","N3")

#c("StageI","StageII","StageIII","StageIV")

library(survminer)

library(survival)

subgroup3 <- characteritics[characteritics$Tstage=="T3/4",]

subgroup3 <- subgroup[subgroup$M=="M1",]

fit\_subgroup <- survfit(Surv(OS.time, OS) ~ group, data = subgroup3)

ggsurvplot(fit\_subgroup, data = subgroup3, linetype = 1,

palette = c("#EE0000B2","#3B4992B2"),

size=1,surv.scale = c("percent"),pval = TRUE,legend.title = "M1",

legend.labs = c("High Scores", "Low Scores"),

break.time.by =12,

xlim = c(0,120),

risk.table = F,risk.table.title = "Patients at risk",

ylab = "Overall Survival, %",

xlab = "Months",font.x = c(20,"plain","black"),

font.y = c(20,"plain","black"),font.tickslab = c(18,"plain","black"),

risk.table.fontsize = 6.5,font.legend =c(20,"plain","black"),

font.main = c(20,"plain","black"),pval.size = 10)

####nomogram#####

library(regplot)

nomoclinical <- Survival\_Clinical

nomoclinical$Stage <- factor(nomoclinical$Stage,ordered = F)

nomoclinical$N <- factor(nomoclinical$N,ordered = F)

nomoclinical$T <- factor(nomoclinical$T,ordered = F)

nomoclinical$M <- factor(nomoclinical$M,ordered = F)

nomoCox <- coxph(Surv(OS.time, OS) ~ pred.multinom +Stage,

data = Survival\_Clinical)

summary(nomoCox)

regplot(nomoCox,failtime = c(12,36,60),observation = T,droplines=T,points = T,)

#####new model#####

library(survcomp)

pred.multinom2 <- as.data.frame(predict(f,nomoclinical))

rownames(pred.multinom2) <- nomoclinical$sample

names(pred.multinom2) <- c("pred")

pred.multinom2$sample <- rownames(pred.multinom2)

nomoclinical <- merge(nomoclinical,pred.multinom2,by="sample")

nomoclinical <- dplyr::arrange(nomoclinical,pred)

library(timeROC)

ROC\_clinical <- timeROC(T = nomoclinical$OS.time,delta = nomoclinical$OS,

marker = nomoclinical$pred,weighting = "marginal",ROC = T,

times=c(12,24,36,48,60), cause = 1)

ROC\_clinical

plot(ROC\_clinical, time=12, col="#D86779", title=FALSE, lwd=2)

plot(ROC\_clinical, time=24, col="#21A2A2",add=TRUE,title=FALSE, lwd=2)

plot(ROC\_clinical, time=36, col="#E3B227", add=TRUE, title=FALSE, lwd=2)

legend(x=0.5, y=0.55,

c(paste0("AUC at 1 years: ", round (ROC\_clinical$AUC[1],4)),

paste0("AUC at 2 years: ", round (ROC\_clinical$AUC[2],4)),

paste0("AUC at 3 years: ", round (ROC\_clinical$AUC[3],4))),

col=c("#D86779","#21A2A2","#E3B227") , lwd=2,bty="n")

cindexScore <- concordance.index(nomoclinical$pred.multinom[1:476],

surv.time = nomoclinical$OS.time[1:476],

surv.event = nomoclinical$OS[1:476],

method = "noether")

cindexClinical <- concordance.index(nomoclinical$pred,na.rm = T,

surv.time = nomoclinical$OS.time,surv.event = nomoclinical$OS,

method = "noether")

cindex.comp(cindexScore$data$surv.event,cindexClinical$data$surv.event)

######校准曲线#############

library(rms)

#Calculation of C-index

f <- coxph(Surv(OS.time,OS) ~ pred.multinom +Stage,

data = nomoclinical)

sum.surv<-summary(f)

sum.surv$concordance #cindex

sum.surv

c.index2<-t(as.data.frame(sum.surv$concordance))

c.index2

Low952 <- (c.index2[1]) - 1.96\*(c.index2[2])

Upper952 <-(c.index2[1]) + 1.96\*(c.index2[2])

c.index2<-cbind(c.index2[1], Low952, Upper952)

c.index2

#calibration curve

cal<- calibrate(coxm, cmethod = 'KM', method = 'boot', u = 60, m = 100, B = 100)

plot(cal,lwd=2,lty=1,errbar.col=c(rgb(0,118,192,maxColorValue=255)),

xlim=c(0.6,1), ylim=c(0.6,1), xlab='Nomogram-Predicted Probability of 5-Year OS', ylab='Actual 5-Year OS(proportion)', col=c(rgb(192,98,83,maxColorValue=255)))

lines(cal[,c('mean.predicted','KM')],type='b',lwd=2, col=c(rgb(192,98,83,maxColorValue=255)), pch=16)

abline(0,1,lty=3,lwd=2,col=c(rgb(0,118,192,maxColorValue=255)))

#calibration curve 1 year

f12<-cph(Surv(OS.time,OS) ~ pred.multinom +Stage,

x = T,y = T,surv = T,data = nomoclinical,time.inc = 12)

cal12<-calibrate(f12, cmethod="KM", method="boot",

u=12, #u需要与前面模型中time.inc一致，生存数据是月份，评估一年所以是12

m=100, #每次抽样的样本量，需要根据样本量确定，分成几组图中就显示几个点

B=1000) #抽样次数

par(mar=c(8,5,3,2),cex=1.0)

plot(cal12,lwd=2,lty=1,errbar.col=c(rgb(0,118,192,maxColorValue=255)),

xlim=c(0,1.0),ylim=c(0,1.0),

xlab="Nomogram-Predicted Probability of 1-year overall survival",

ylab="Actual 1-year overall survival (proportion)",

col=c(rgb(192,98,83,maxColorValue=255)))

#calibration curve 3 year

f36<-cph(Surv(OS.time,OS) ~ pred.multinom +Stage,

x = T,y = T,surv = T,data = nomoclinical,time.inc = 36)

cal36<-calibrate(f36, cmethod="KM", method="boot", u=36, m=100, B=1000)

par(mar=c(8,5,3,2),cex=1.0)

plot(cal36,lwd=2,lty=1,errbar.col=c(rgb(0,118,192,maxColorValue=255)),

xlim=c(0,1.0),ylim=c(0,1.0),

xlab="Nomogram-Predicted Probability of 3-year overall survival",

ylab="Actual 3-year overall survival (proportion)",

col=c(rgb(192,98,83,maxColorValue=255)))

#calibration curve 5 year

f60 <- cph(Surv(OS.time,OS) ~ pred.multinom +Stage,

x = T,y = T,surv = T,data = nomoclinical,time.inc = 60)

cal60 <- calibrate(f60, cmethod="KM", method="boot", u=60, m=100, B=1000)

par(mar=c(8,5,3,2),cex=1.0)

plot(cal60,lwd=2,lty=1,errbar.col=c(rgb(0,118,192,maxColorValue=255)),

xlim=c(0,1.0),ylim=c(0,1.0),

xlab="Nomogram-Predicted Probability of 5-year overall survival",

ylab="Actual 5-year overall survival (proportion)",

col=c(rgb(192,98,83,maxColorValue=255)))

######GSEA#####

LUADFerro <- LUAD\_TPM[,colnames(LUAD\_TPM) %in% rownames(datExpr)]

LUADFerro <- cbind(LUADFerro[,colnames(LUADFerro) %in% GroupHigh],

LUADFerro[,colnames(LUADFerro) %in% GroupLow])

write.csv(LUADFerro,file = "Plot/GSEA/LUADFerro.csv")