Cell Reports Medicine, Volume *3*

Supplemental information

Radiogenomic analysis reveals tumor heterogeneity of triple-negative breast cancer

Lin Jiang, Chao You, Yi Xiao, He Wang, Guan-Hua Su, Bing-Qing Xia, Ren-Cheng Zheng, Dan-Dan Zhang, Yi-Zhou Jiang, Ya-Jia Gu, and Zhi-Ming Shao

Figure S1

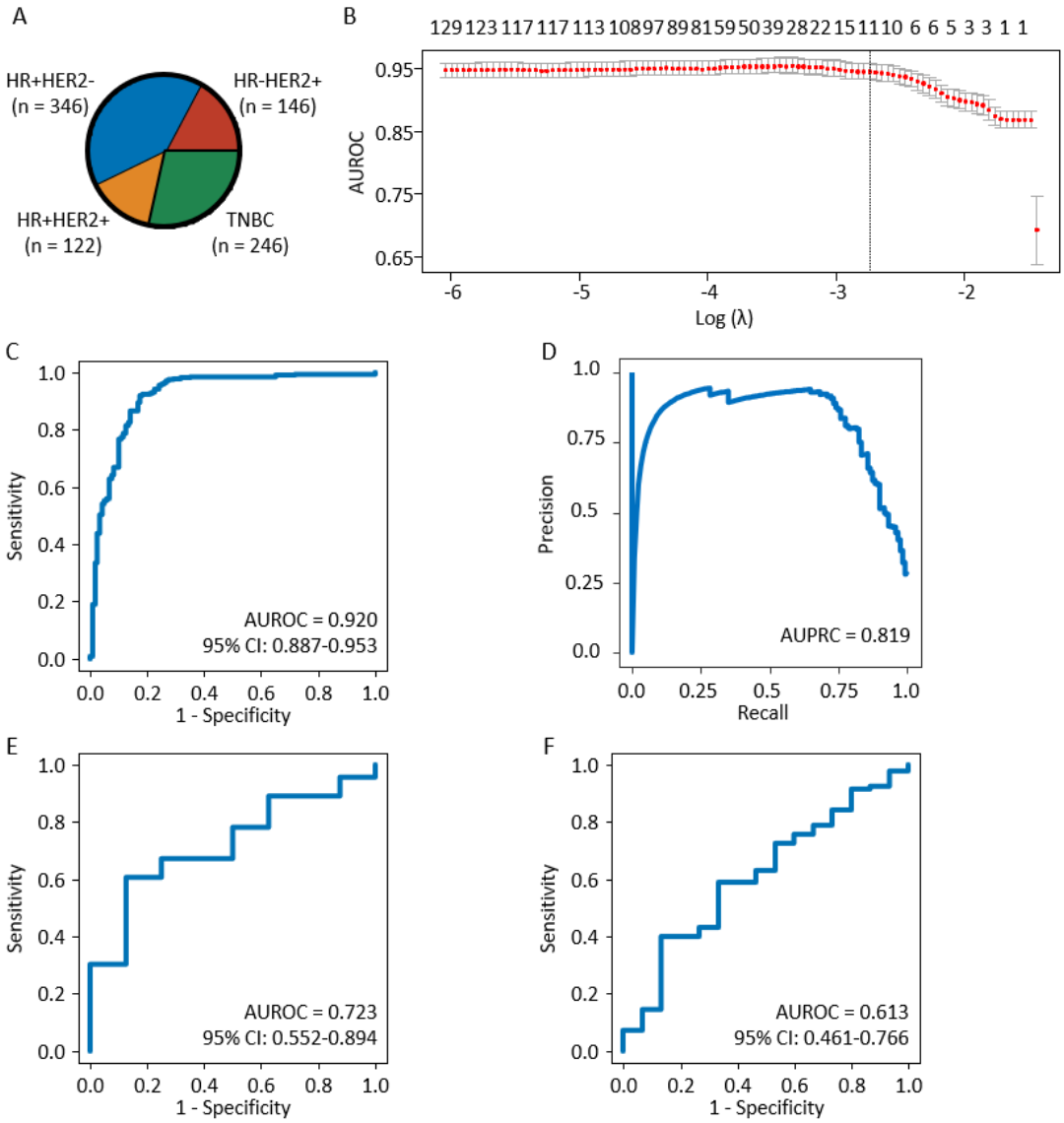

Figure S1. Patient composition, feature selection and model building based on FUSCC breast cancer radiomic cohort. Related to Figure 1, Table S2 and STAR METHODS.

(A) Composition of patients enrolled in FUSCC breast cancer radiomic cohort according to receptor status. (B) Radiomic feature selection using LASSO method. (C) AUROC of TNBC-predicting radiomic model using linear regression algorithm. (D) AUPRC of TNBC-predicting radiomic model using linear regression algorithm. (E) AUROC of the TNBC-predicting radiomic model in IPMCH dataset (n = 54). (F) AUROC of the TNBC-predicting radiomic model in RENJI dataset (n = 110). Abbreviations: AUPRC, Area under precision-recall curve; AUROC, Area under receiver operating characteristic curve; FUSCC, Fudan University Shanghai Cancer Center; IPMCH, Shanghai International Peace Maternal and Children Hospital; LASSO, least absolute shrinkage and selection operator; RENJI, Shanghai Jiaotong University Renji Hospital; TNBC, triple-negative breast cancer.

Figure S2

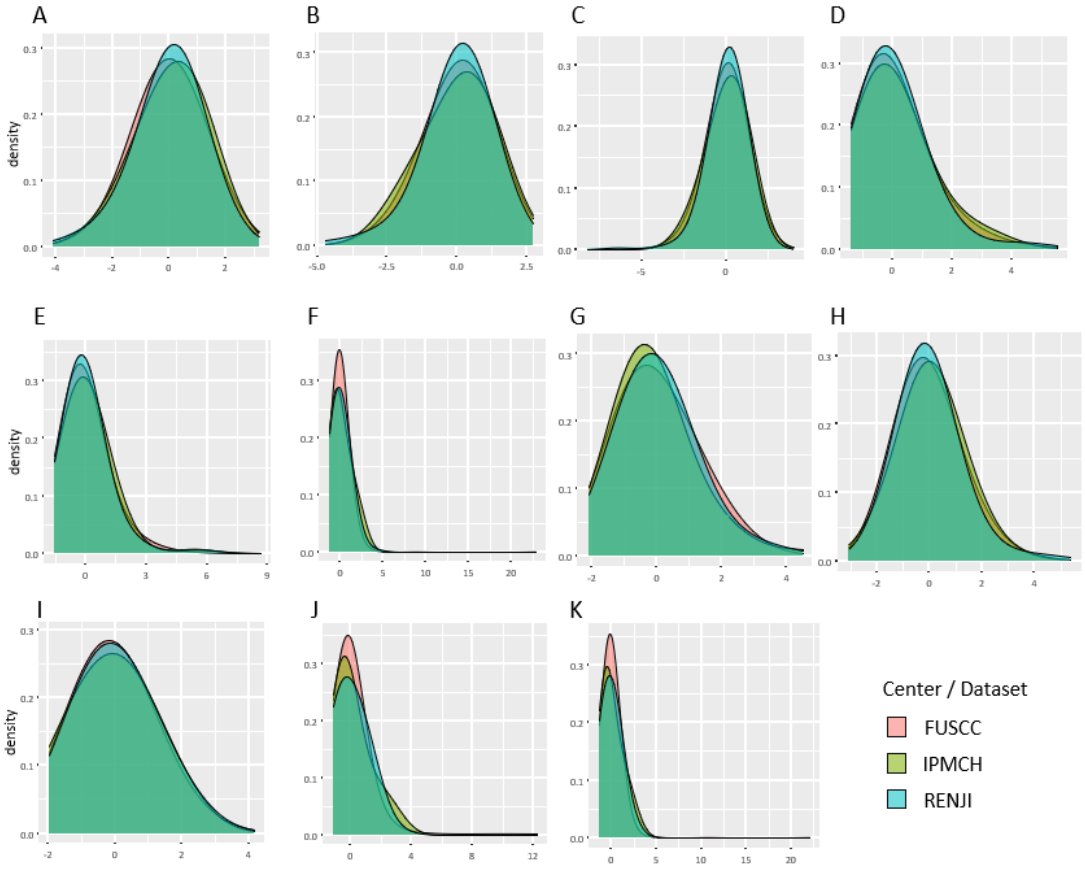

Figure S2. The density distribution of the eleven features used to predict TNBC between different centers and datasets. A to K represent these eleven features. Related to Figure 1 and Table S2.

Abbreviations: FUSCC, Fudan University Shanghai Cancer Center; IPMCH, Shanghai International Peace Maternal and Children Hospital; RENJI, Shanghai Jiaotong University Renji Hospital; TNBC, triple-negative breast cancer.

Figure S3

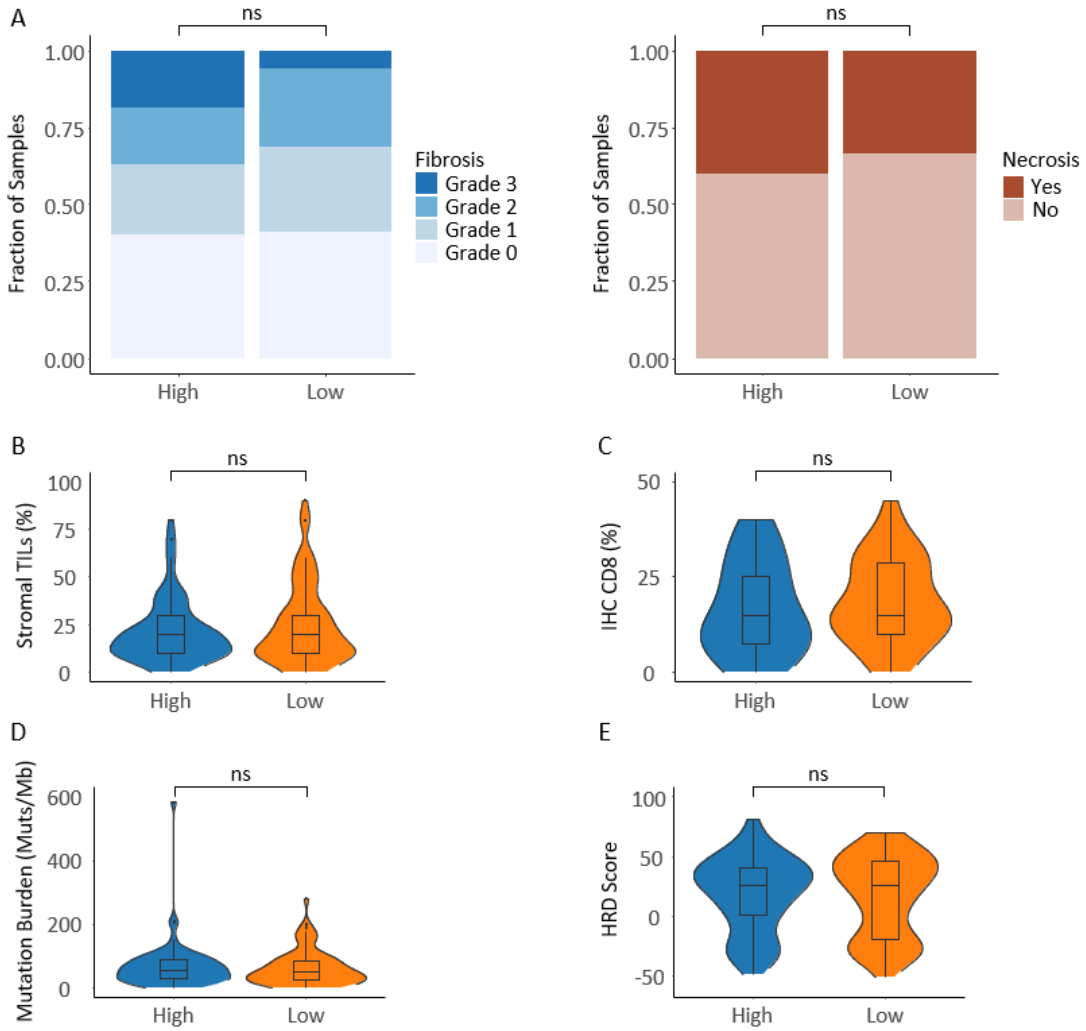

Figure S3. The associations between Peri_V_DN feature and clinicopathological factors. Related to Figure 3.

(A) Distribution of the grade of fibrosis and necrosis between high and low Peri_V_DN groups. (B) Distribution of the amount of stromal tumor infiltrating lymphocytes (TILs) between high and low Peri_V_DN groups. (C) Distribution of the immunohistochemistry (IHC) CD8 staining between high and low Peri_V_DN groups. (D) Distribution of the tumor mutation burden between high and low Peri_V_DN groups. (E) Distribution of the homologous recombination deficiency (HRD) score between high and low Peri_V_DN groups. ns $p > 0.05$.

Abbreviations: HRD, homologous recombination deficiency; IHC, immunohistochemistry; TIL, tumor infiltrating lymphocytes.

Figure S4

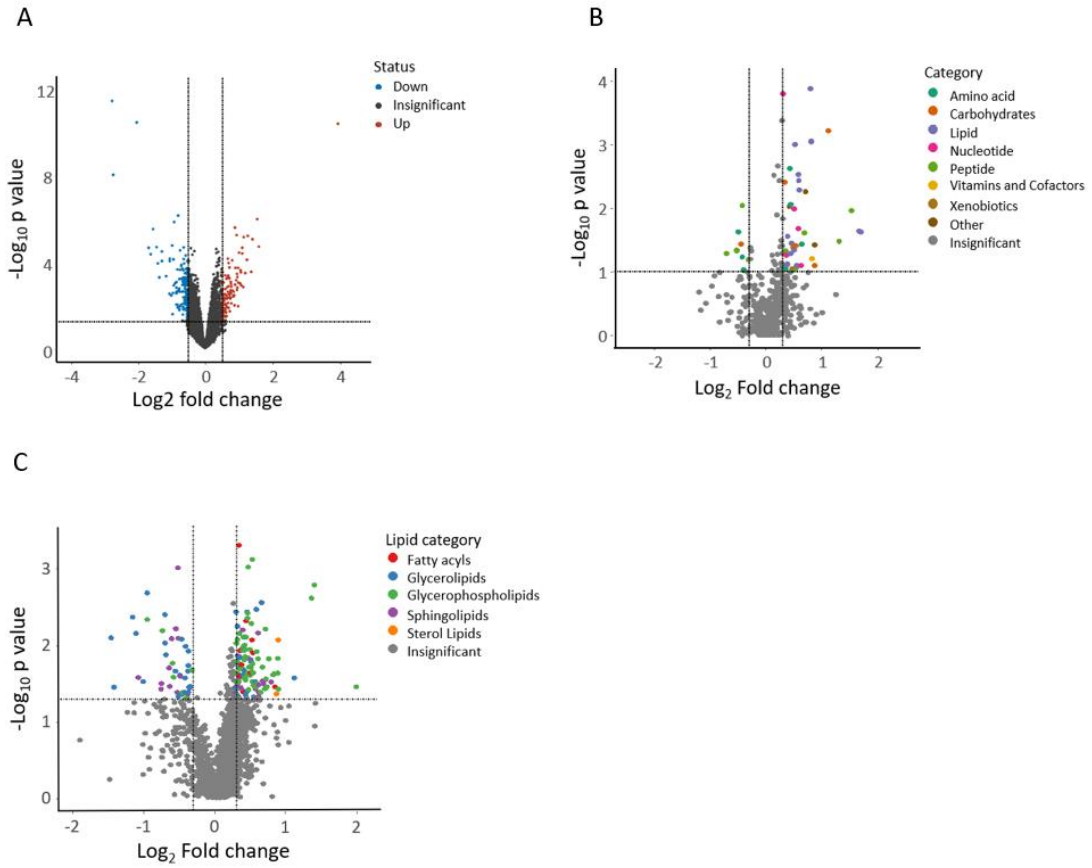

Figure S4. Differentially expressed genes and differentially abundant metabolites between high and low Peri_V_DN groups. Related to Figure 4, Table S5 and S6.

(A) Differentially expressed genes between high and low Peri_V_DN groups. (B) Differentially abundant polar metabolites between high and low Peri_V_DN groups. (C) Differentially abundant lipids between high and low Peri_V_DN groups.

Abbreviations: Peri_V_DN, peritumoral variance in dependence nonuniformity of peritumoral regions.

Table S1. The imaging parameters of different CE-MRI machines. Related to Figure 1 and STAR METHODS.

MRI machine	Magnetic intensity (T)	Dosage of Gd-DTPA (mmol/kg)	Number of series	Time of repetition (s)	Time of echo (s)
Aurora	1.5	0.1	6	5.1	1.7
GE	1.5	0.2	4	5	29
Siemens	3	0.1	4	6.5	3.5
Siemens	1.5	0.1	7	4.2	1.53
GE	3	0.1	6	4	2.1
Philips	3	0.1	6	4.2	2.1
Siemens	3	0.1	5	4.4	1.6

MRI machine	Magnetic intensity (T)	Field of view (mm*mm)	Slice thickness (mm)	Gap (mm)	Flip angle
Aurora	1.5	260*260	3	0	15°
GE	1.5	360*360	1.1	0	15°
Siemens	3	300*300	3	0	10°
Siemens	1.5	360*360	1.5	0	15°
GE	3	320*320	2.4	0	10°
Philips	3	358*280	2	0	12°
Siemens	3	320*320	1.5	0	10°

MRI machine	Magnetic intensity (T)	Cohort
Aurora	1.5	FUSCC breast cancer radiomic cohort and TNBC radiogenomic cohort
GE	1.5	FUSCC breast cancer radiomic cohort and TNBC radiogenomic cohort
Siemens	3	FUSCC breast cancer radiomic cohort
Siemens	1.5	IPMCH cohort
GE	3	RENJI cohort
Philips	3	RENJI cohort
Siemens	3	RENJI cohort

Abbreviations: CE-MRI, contrast-enhanced magnetic resonance imaging; FUSCC, Fudan University Shanghai Cancer Center; IPMCH, Shanghai International Peace Maternal and Children Hospital; RENJI, Shanghai Jiaotong University Renji Hospital.

Table S2. Retained radiomic features for identification of TNBC. Related to Figure 1.

Subtype	Selected feature	ROI
TNBC	mean of Zone Entropy	tumor
	mean of HHL Mean	peritumor
	mean of HHL Skewness	peritumor
	variance of LHL Mean	peritumor
	variance of HHL Mean	peritumor
	variance of HHH Contrast	peritumor
	skewness of Small Area High Gray Level Emphasis	intratumor
	mean of Mean	intratumor
	mean of Minimum	intratumor
	variance of Large Dependence Low Gray Level Emphasis	tumor-peritumor
	variance of Contrast	tumor-peritumor

Abbreviations: TNBC, triple-negative breast cancer.

Table S3. Demographic characteristics of TNBC radiogenomic cohort. Related to Figure 1 and Table 1.

TNBC radiogenomic cohort						
		Training cohort (n = 101)		Validation cohort (n = 101)		
Characteristics		No.	%	No.	%	P value
Age (years)						0.69
	Median	53		55		
	IQR	47-61		47-60		
	≤50	38	37.60%	41	40.60%	
	>50	63	62.40%	60	59.40%	
T						0.54
	1	48	47.50%	43	42.60%	
	2	48	47.50%	55	54.50%	
	3	5	5.00%	3	3.00%	
N						0.33
	0	60	59.40%	62	62.60%	
	1	30	29.70%	21	21.20%	
	2	8	7.90%	8	8.10%	
	3	3	3.00%	6	6.10%	
	Unknown	0	0%	2	2.00%	
Subtype						<0.001
	BLIS	35	34.70%	25	24.80%	
	IM	26	25.70%	17	16.80%	
	MES	16	15.80%	9	8.90%	
	LAR	20	19.80%	19	18.80%	
	Unknown	4	4.00%	31	30.70%	
Ki-67 (%)						0.52
	≤20	12	11.90%	16	15.80%	
	>20	89	88.10%	84	83.20%	
	Unknown	0	0%	1	1.00%	

Abbreviations: BLIS, basal-like immune-suppressed; IM, immunomodulatory; IQR, interquartile range; MES, mesenchymal-like; LAR, luminal androgen receptor; TNBC, triple-negative breast cancer.

Table S4. Retained radiomic features for distinguishing TNBC molecular subtypes. Related to Figure 2.

Subtype	Selected feature	ROI
BLIS	enh41 MaximumProbability	tumor
	enh41 RunLengthNonUniformityNormalized	tumor
	enh43 ClusterShade	peritumor
	enh41_Autocorrelation	peritumor
IM	enh43 SmallAreaEmphasis	tumor-peritumor
	enh43 LargeDependenceEmphasis	tumor-peritumor
	enh43 Busyness	tumor-peritumor
	enh43 Strength	tumor-peritumor
	enh32 Contrast	tumor-peritumor
	enh32 SmallDependenceLowGrayLevelEmphasis	tumor-peritumor
	enh32 LowGrayLevelEmphasis	tumor-peritumor
	enh32 LowGrayLevelRunEmphasis	tumor-peritumor
	e1 MaximumProbability	tumor-peritumor
	e3 Minimum	tumor-peritumor
	kurtosis of Correlation	tumor-peritumor
MES	Elongation	tumor
	enh32 Coarseness	peritumor
LAR	kurtosis of Skewness	tumor
	variance of MaximumProbability	tumor
	enhancement of ClusterShade	tumor
	enh32 Idmn	tumor
	kurtosis of Kurtosis	peritumor
	enh21 90Percentile	peritumor
	enh32 Idmn	peritumor

Abbreviations: BLIS, basal-like immune-suppressed; IM, immunomodulatory; MES, mesenchymal-like; LAR, luminal androgen receptor; ROI, region of interest.

Custom codes

```
######################################################################
########## Figure 3A-C Feature Selection and Prognosis #########

radiomic_data <- read.csv("Data Matrix.csv",header = T,sep = ",",stringsAsFactors = F,row.names = 1)

clinical_data <- read.csv("FUSCCTNBC_ClinicalData.csv",header = T,sep = ",",stringsAsFactors =
F,row.names = 1)

sample_names <- intersect(rownames(radiomic_data),rownames(clinical_data))

radiomic_data <- radiomic_data[rownames(radiomic_data)%in%sample_names,]
clinical_data <- clinical_data[match(rownames(radiomic_data),rownames(clinical_data)),]

radiomic_data <- as.data.frame(scale(radiomic_data))

feature <- colnames(radiomic_data)

id_list <- read.csv("group.csv",header = T,sep = ",",stringsAsFactors = F)

rownames(id_list) <- id_list[,1]

train_id <- rownames(id_list)[id_list$Cohort=="Training"]
val_id <- rownames(id_list)[id_list$Cohort=="Validation"]

radiomic_data_train <- radiomic_data[train_id,]
radiomic_data_val <- radiomic_data[val_id,]

clinical_data_train <- clinical_data[train_id,]
clinical_data_val <- clinical_data[val_id,]

RFS_HR_matrix <- as.data.frame(rfs_cox_results[,c(1,4)])
RFS_HR_matrix$HR <- log2(RFS_HR_matrix$HR)
colnames(RFS_HR_matrix)[1] <- "log2HR"

RFS_HR_cutoff <- 1.8

RFS_HR_matrix$change <- factor(ifelse(abs(RFS_HR_matrix$log2HR)>RFS_HR_cutoff,
                                      ifelse(RFS_HR_matrix$log2HR>RFS_HR_cutoff,'Incresing
risk','Decreasing risk'),'NOT'),
                                      levels = c('Decreasing risk','NOT','Incresing risk'))

table(RFS_HR_matrix$change)
```

```
rownames(RFS_HR_matrix)[RFS_HR_matrix$change == "Incresing risk"]
rownames(RFS_HR_matrix)[RFS_HR_matrix$change == "Decreasing risk"]

RFS_HR_matrix$label <- ifelse(RFS_HR_matrix$change == 'Decreasing risk' |
RFS_HR_matrix$change == 'Incresing risk',
                                        rownames(RFS_HR_matrix),"")

OS_HR_matrix <- as.data.frame(os_cox_results[,c(1,4)])
OS_HR_matrix$HR <- log2(OS_HR_matrix$HR)
colnames(OS_HR_matrix)[1] <- "log2HR"

OS_HR_cutoff <- 10

OS_HR_matrix$change <- factor(ifelse(abs(OS_HR_matrix$log2HR)>OS_HR_cutoff,
                                        ifelse(OS_HR_matrix$log2HR>OS_HR_cutoff,'Incresing
risk','Decreasing risk'),'NOT'),
                                        levels = c('Decreasing risk','NOT','Incresing risk'))

table(OS_HR_matrix$change)

rownames(OS_HR_matrix)[OS_HR_matrix$change == "Incresing risk"]
rownames(OS_HR_matrix)[OS_HR_matrix$change == "Decreasing risk"]

OS_HR_matrix$label <- ifelse(OS_HR_matrix$change == 'Decreasing risk' | OS_HR_matrix$change
== 'Incresing risk',
                                       rownames(OS_HR_matrix),"")

intersect(rownames(RFS_HR_matrix)[RFS_HR_matrix$change!="NOT"],
          rownames(OS_HR_matrix)[OS_HR_matrix$change!="NOT"])



HR_matrix <- as.data.frame(cbind(RFS_HR_matrix$log2HR,OS_HR_matrix$log2HR))

colnames(HR_matrix) <- c("RFS_HR","OS_HR")
rownames(HR_matrix) <- rownames(RFS_HR_matrix)
HR_matrix$Status <-
factor(ifelse(abs(HR_matrix$RFS_HR)<=RFS_HR_cutoff|abs(HR_matrix$OS_HR)<=OS_HR_cutoff,
"Insignificant",

ifelse(HR_matrix$RFS_HR>RFS_HR_cutoff&HR_matrix$OS_HR>OS_HR_cutoff,"Increasing
Risk","Decreasing Risk")),
                                    levels = c("Increasing Risk","Insignificant","Decreasing Risk"))
```

```
ggplot(data=HR_matrix,aes(x=RFS_HR, y=OS_HR,color=Status))+
   geom_point(size=8)+geom_jitter()+
   scale_color_manual(name="",values = c("#BC3C29FF","#767676FF","#0072B5FF"))+
   geom_hline(yintercept = OS_HR_cutoff,lty=4,lwd=0.6)+
   geom_hline(yintercept = -OS_HR_cutoff,lty=4,lwd=0.6)+
   geom_vline(xintercept = RFS_HR_cutoff,lty=4,lwd=0.6)+
   geom_vline(xintercept = -RFS_HR_cutoff,lty=4,lwd=0.6)+
   scale_x_continuous(limits = c(-3,3),breaks = seq(-3,3,1.5))+
   scale_y_continuous(limits = c(-30,30),breaks = seq(-30,30,15))+
   labs(x="RFS Hazard Ratio",y="OS Hazrad Ratio")+
   theme_classic()+
   theme(axis.title = element_text(size = 16),axis.text=element_text(size = 16),
         legend.title = element_text(size = 16),legend.text = element_text(size = 16))


var(radiomic_data_train[,"p_va42"])
var(radiomic_data_train[,"t_ske79"])
var(radiomic_data_train[,"t_enh7"])

prognostic_feature <- c("p_va42","t_ske79","t_enh7")

filtered_radiomic_train <- radiomic_data_train[,prognostic_feature]
filtered_radiomic_val <- radiomic_data_val[,prognostic_feature]


km_result_val <- as.data.frame(matrix(nrow = length(prognostic_feature),ncol = 1))
rownames(km_result_val) <- prognostic_feature
colnames(km_result_val) <- "p.value"

for(i in rownames(km_result_val)){
   group <-
factor(ifelse(filtered_radiomic_val[,i]>=median(filtered_radiomic_train[,i]),"High","Low"),levels =
c("Low","High"))
   surv_data <- data.frame(group=group,RFS_time=clinical_data_val$RFS_time_Days,
                           RFS_status=clinical_data_val$RFS_Status)
   m <- survdiff(Surv(RFS_time,RFS_status==1)~group,rho = 0,data = surv_data)
   p <- 1-pchisq(m$chisq,length(m$n)-1)
   km_result_val[i,1] <- p
}

km_result_val
```

```
surv_color <- pal_lancet()(2)

group1 <-
factor(ifelse(filtered_radiomic_train$p_va42>=median(filtered_radiomic_train$p_va42),"High","Low"
),
                        levels = c("Low","High"))

group2 <-
factor(ifelse(filtered_radiomic_val$p_va42>=median(filtered_radiomic_train$p_va42),"High","Low"),
                        levels = c("Low","High"))

surv_data2 <- data.frame(group=group2,
                                    RFS_time=clinical_data_val$RFS_time_Months/30,
                                    RFS_status=clinical_data_val$RFS_Status,
                                    OS_time=clinical_data_val$OS_time_Months/30,
                                    OS_status=clinical_data_val$OS)

survfit2_RFS <- survfit(Surv(RFS_time,RFS_status==1)~group,data = surv_data2)
survfit2_OS <- survfit(Surv(OS_time,OS_status==1)~group,data = surv_data2)

survdiff(Surv(RFS_time,RFS_status==1)~group,data = surv_data2,rho = 0)
survdiff(Surv(OS_time,OS_status==1)~group,data = surv_data2,rho = 0)

plot(survfit2_RFS,lwd=4,col = c(surv_color[1],surv_color[2]),
       mark.time = T,xlab = "Time",ylab = "Survival Probability")
legend("bottomleft", c("p-va42 low", "p-va42 high"),
          lwd = 4,col = c(surv_color[1],surv_color[2]))
text(3,0.4,"p = 0.01",cex = 1.5)

plot(survfit2_OS,lwd=4,col = c(surv_color[1],surv_color[2]),
       mark.time = T,xlab = "Time",ylab = "Survival Probability")
legend("bottomleft", c("p-va42 low", "p-va42 high"),
          lwd = 4,col = c(surv_color[1],surv_color[2]))
text(3,0.4,"p = 0.004",cex = 1.5)


#--- MultiCox


filtered_radiomic <- as.data.frame(cbind(clinical_data$RFS_time_Days,clinical_data$RFS_Status,
                                                    clinical_data$OS_time_Days,clinical_data$OS,
                                                    clinical_data$Size_cm,clinical_data$LN_positive,
                                                    radiomic_data$p_va42))
```

```
colnames(filtered_radiomic)[1:7] <-
c("RFS_time","RFS_status","OS_time","OS_status","Tumor_size","Lymph_nodes","p_va42")

filtered_radiomic$p_va42_group <-
factor(ifelse(filtered_radiomic$p_va42>=median(filtered_radiomic$p_va42),"High","Low"),
                                                        levels = c("High","Low"))

filtered_radiomic$Tumor_size_group <-
ifelse(filtered_radiomic$Tumor_size<=2,"T1",ifelse(filtered_radiomic$Tumor_size>5,"T3
above","T2"))

filtered_radiomic$Lymph_nodes_group <-
ifelse(filtered_radiomic$Lymph_nodes==0,"N0",ifelse(filtered_radiomic$Lymph_nodes<4,"N1",ifelse(
filtered_radiomic$Lymph_nodes>=10,"N3","N2")))

filtered_radiomic$subtype <- clinical_data$mRNA_Subtype

rfs_cox_model <-
coxph(Surv(RFS_time,RFS_status==1)~p_va42_group+Tumor_size_group+Lymph_nodes_group+sub
type,data = filtered_radiomic)
rfs_cox_model
summary(rfs_cox_model)

os_cox_model <-
coxph(Surv(OS_time,OS_status==1)~p_va42_group+Tumor_size_group+Lymph_nodes_group+subty
pe,data = filtered_radiomic)
os_cox_model
summary(os_cox_model)

rfs_cox_model_summary <- do.call("rbind",regressionTable(rfs_cox_model))[,4:7]
os_cox_model_summary    <- do.call("rbind",regressionTable(os_cox_model))[,4:7]
```

```
################################################################################
############## Figure 3D-E Clinicopathological Index Comparison ###############

clinical_data <- read.csv("FUSCCTNBC_ClinicalData.csv",header = T,sep = ",",row.names = 1)
list <- read.csv("High_Low_peritumorITH.csv",header = T,sep = ",")

rownames(list) <- list$Hospital_ID

clinical_data <- clinical_data[match(list$Hospital_ID,rownames(clinical_data)),]

data_total <- mutate(clinical_data, Cohort = list$Cohort, Group = list$Group)



ggplot(data = data_total,aes(x=Group,y=LN_positive,fill=Group))+
  geom_violin()+geom_boxplot(width=0.03)+
  theme_classic()+scale_fill_d3()+
  scale_y_continuous(name = "Positive lymph nodes",limits = c(0,10),breaks = c(0,5,10))+
  theme(axis.title = element_text(size = 14),axis.text=element_text(size = 14),
        legend.title = element_text(size = 14),legend.text = element_text(size = 14))

wilcox.test(LN_positive~Group,data = data_total)



mrna_color <- c("#EE4923","#7CC243","#9180BA","#3EAADF")

data_total1 <- filter(data_total,mRNA_Subtype!="Unknown")%>%
  group_by(Group,mRNA_Subtype)%>%
  summarise(n=n())%>%
  mutate(prop=n/sum(n))

ggplot(data = data_total1,aes(x=Group,y=prop,fill=mRNA_Subtype))+
  geom_bar(stat = "identity",position = "stack")+
  theme_classic()+coord_flip()+
  scale_fill_manual(name = "FUSCC TNBC Subtype",values = mrna_color,labels =
c("BLIS","IM","LAR","MES"))+
  theme(axis.title = element_text(size = 14),axis.text=element_text(size = 14),
        legend.title = element_text(size = 14),legend.text = element_text(size = 14))

chisq.test(matrix(c(data_total1$n),ncol = 2))
```

```
#####################################################################
############## Figure 4 Biological Characters #################

FUSCCTNBC_RNAseqShi.Tumor_log2 <- read.csv("RNAseq.csv",header = T,sep = ",",row.names =
1)

radiomic_data <- read.csv("TNBC 多组学队列影像组学数据.csv",header = T,sep =
",",stringsAsFactors = F,row.names = 1)

id_list <- read.csv("影像组学 ITH 高低分组.csv",header = T,sep = ",",stringsAsFactors = F)

clinical_data <- read.csv("FUSCCTNBC_ClinicalData.csv",header = T,sep = ",",stringsAsFactors = F)
clinical_data <- clinical_data[match(rownames(radiomic_data),clinical_data$Hospital_ID),]

rownames(clinical_data) <- clinical_data$Project_ID
rownames(radiomic_data) <- rownames(clinical_data)


sample_names <- rownames(radiomic_data)
train_id <- id_list$Project_ID[id_list$Cohort=="training"]
val_id <- id_list$Project_ID[id_list$Cohort=="validation"]

radiomic_data_train <- radiomic_data[train_id,]
radiomic_data_val <- radiomic_data[val_id,]

clinical_data_train <- clinical_data[train_id,]
clinical_data_val <- clinical_data[val_id,]

cutoff <- median(radiomic_data_train$p_va42)


need_p <- intersect(rownames(radiomic_data),colnames(FUSCCTNBC_RNAseqShi.Tumor_log2))

radiomic_data <- radiomic_data[need_p,]
clinical_data <- clinical_data[need_p,]


grouplist <- factor(ifelse(radiomic_data$p_va42>=cutoff,"High","Low"),
                    levels = c("Low","High"))

names(grouplist) <- rownames(radiomic_data)

genes <- rownames(FUSCCTNBC_RNAseqShi.Tumor_log2)
```

```
exprset <- 2^FUSCCTNBC_RNAseqShi.Tumor_log2-1

tmp <- apply(exprset,1,function(x){
  sum(x==0) < 10
})

exprset <- exprset[tmp,]

exprset <- exprset[,need_p]


if(T){
  library(edgeR)

  d <- DGEList(counts=exprset,group=factor(grouplist))

  keep <- rowSums(cpm(d)>1) >= 2
  table(keep)
  d <- d[keep, , keep.lib.sizes=FALSE]

  d$samples$lib.size<-colSums(d$counts)

  d <- calcNormFactors(d)
  d$samples

  plotMDS(d)

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

  rownames(design)<-colnames(d)
  colnames(design)<-levels(factor(grouplist))

  deg <- d
  deg <- estimateGLMCommonDisp(deg,design)
  deg <- estimateGLMTrendedDisp(deg, design)
  deg <- estimateGLMTagwiseDisp(deg, design)

  fit <- glmFit(deg,design)

  lrt <- glmLRT(fit,contrast=c(-1,1))

  edge_DEG <- topTags(lrt, n=nrow(deg))
  edge_DEG
```

```
    summary(de<-decideTestsDGE(lrt))

    detags<-rownames(d)[as.logical(de)];
    plotSmear(lrt,de.tags=detags);
    abline(h=c(-1,1),col="blue")

    edge_DEG <- as.data.frame(edge_DEG)

    nrDEG_edge <- edge_DEG[,c(1,5)]
    colnames(nrDEG_edge) <- c('log2FoldChange','FDR')

}

colnames(edge_DEG)[1] <- 'logFC'
colnames(edge_DEG)[4] <- 'P.Value'
colnames(edge_DEG)[5] <- 'FDR'
nrDEG <- edge_DEG

nrDEG$SYMBOL <- rownames(nrDEG)

df <- bitr(rownames(nrDEG),fromType = "SYMBOL",toType = "ENTREZID",OrgDb = org.Hs.eg.db )

nrDEG <- merge(nrDEG,df,by='SYMBOL')
head(nrDEG)

geneList <- nrDEG$logFC
names(geneList) <- nrDEG$ENTREZID
geneList <- sort(geneList,decreasing = T)

KEGG_gseresult <- gseKEGG(geneList,nPerm = 1000,minGSSize = 10,maxGSSize =
1000,pvalueCutoff = 0.2)
KEGG_gseresult

KEGG_gseresult_1 <- arrange(KEGG_gseresult,desc(abs(NES)))%>%
    group_by(sign(NES))%>%
    as.data.frame()

KEGG_gseresult_1$NES <- as.numeric(KEGG_gseresult_1$NES)
KEGG_gseresult_1$Group <- factor(KEGG_gseresult_1$`sign(NES)`,levels = c(1,-1))

KEGG_gseresult_2 <- KEGG_gseresult_1[KEGG_gseresult_1$Description%in%c("Oxidative
phosphorylation",
                                                                        "Biosynthesis
of unsaturated fatty acids","Glycolysis / Gluconeogenesis",
```

```
                                                                    "Fatty acid
elongation","Biosynthesis of amino acids","Arginine biosynthesis",
                                                                    "Cysteine and
methionine metabolism","Alanine, aspartate and glutamate metabolism",
                                                                    "Carbon
metabolism","Nitrogen metabolism",
                                                                    "Th17 cell
differentiation","Th1 and Th2 cell differentiation","Primary immunodeficiency",
                                                                    "Antigen
processing and presentation","Natural killer cell mediated cytotoxicity",
                                                                    "T cell
receptor signaling pathway","B cell receptor signaling pathway"),]

KEGG_gseresult_2 <- KEGG_gseresult_2[order(KEGG_gseresult_2$NES,decreasing = F),]

ggplot(KEGG_gseresult_2,aes(x=reorder(Description,order(NES,decreasing=F)),y=NES,fill=Group))
+
  geom_bar(stat = "identity") +
  scale_fill_npg() +
  scale_x_discrete(name = "Pathway names") +
  scale_y_continuous(name = "NES") +
  coord_flip() + theme_classic()  +
  theme(axis.title = element_text(size = 14),axis.text=element_text(size = 14),
        legend.title = element_text(size = 14),legend.text = element_text(size = 14))



ReactomePA_result <- gsePathway(geneList,nPerm = 1000,minGSSize = 10,maxGSSize =
1000,pvalueCutoff = 0.2)
ReactomePA_result

reactome_1 <- arrange(ReactomePA_result,desc(abs(NES)))%>%
  group_by(sign(NES))%>%
  as.data.frame()

reactome_1$NES <- as.numeric(reactome_1$NES)
reactome_1$Group <- factor(reactome_1$`sign(NES)`,levels = c(1,-1))

reactome_2 <- reactome_1[reactome_1$Description%in%c("Cholesterol
biosynthesis","Gluconeogenesis","Glycolysis",
                                                    "Metabolism of polyamines","Glucose metabolism","Nucleobase
biosynthesis",
                                                    "Fatty acyl-CoA biosynthesis","The citric acid (TCA) cycle and
respiratory electron transport",
                                                    "Metabolism of amino acids and derivatives",
```

```
                                "Interferon alpha/beta signaling","PD-1 signaling","Interferon gamma
signaling",
                                "Adaptive Immune System","Costimulation by the CD28 family",
                                "TCR signaling","Innate Immune System"),]

reactome_2 <- reactome_2[order(reactome_2$NES,decreasing = F),]

ggplot(reactome_2,aes(x=reorder(Description,order(NES,decreasing=F)),y=NES,fill=Group)) +
  geom_bar(stat = "identity") +
  scale_fill_npg() +
  scale_x_discrete(name = "Pathway names") +
  scale_y_continuous(name = "NES") +
  coord_flip() + theme_classic()  +
  theme(axis.title = element_text(size = 14),axis.text=element_text(size = 14),
        legend.title = element_text(size = 14),legend.text = element_text(size = 14))
```

```
################################################################################################
############### Figure 5 Immune Cells and Immune-Related Molecules #################

exprset <- as.matrix(FUSCCTNBC_RNAseqShi.Tumor_log2)

signature_file <- read.csv("Cibersort+MCPCounter_signature.csv",header = T,sep =
",",stringsAsFactors = F)

signature_file <- signature_file[order(signature_file$CellClass),]
signature <- split(as.matrix(signature_file)[,1], signature_file[,3])

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

list <- list[order(list$Cohort,list$Group),]

need_p <- intersect(list$Project_ID,colnames(exprset))

exprset <- exprset[,need_p]
list <- list[list$Project_ID%in%need_p,]

list_train <- list[list$Cohort=="training",]
list_val <- list[list$Cohort=="validation",]

exprset_train <- exprset[,list_train$Project_ID]
dim(exprset_train)
exprset_val <- exprset[,list_val$Project_ID]
dim(exprset_val)



a <- gsva(exprset,signature,method='ssgsea',kcdf='Gaussian',abs.ranking=T)

cellorder <- unique(signature_file$CellType)

a1 <- t(scale(t(a)))
a1 <- a1[cellorder,]

whole_df <- as.data.frame(cbind(list$Group,t(a1)))
colnames(whole_df)[1] <- "Group"

for (i in 2:ncol(whole_df)) {
   whole_df[,i] <- as.numeric(as.character(whole_df[,i]))
}
```

```
compare <- as.data.frame(matrix(nrow = ncol(whole_df)-1,ncol = 5))
colnames(compare) <- c("Cell Type","High ITH","Low ITH","pvalue","Status")
rownames(compare) <- colnames(whole_df)[2:ncol(whole_df)]

compare[,1] <- cellorder

whole_compare <- compare

for (i in rownames(whole_compare)) {
  whole_compare[i,2] <- mean(whole_df[whole_df$Group=="High",i])
  whole_compare[i,3] <- mean(whole_df[whole_df$Group=="Low",i])
  t <- wilcox.test(whole_df[whole_df$Group=="High",i],whole_df[whole_df$Group=="Low",i])
  whole_compare[i,4] <- t$p.value
}

whole_compare$FDR <- p.adjust(whole_compare$pvalue,method = "fdr")

whole_compare$Status <- ifelse(whole_compare$FDR>=0.1,"NOT",
                               ifelse(whole_compare$`High ITH`>whole_compare$`Low
ITH`,"UP","DOWN"))

whole_df_long <- gather(whole_df,key = CellType,value = Score,"B.cells.naive":"Fibroblasts")
whole_df_long$Score <- as.numeric(whole_df_long$Score)
whole_df_long$CellType <- factor(whole_df_long$CellType,levels = cellorder)

ggplot(whole_df_long,aes(x=CellType,y=Score,fill=Group))+
  geom_boxplot()+scale_fill_nejm()+coord_flip()+
  scale_y_continuous(limits = c(-5,5),breaks = seq(-5,5,2.5))+
  theme_classic()+labs(x="Cell Type",y="ssGSEA Score")+
  theme(axis.title = element_text(size = 14),axis.text=element_text(size = 14),
        legend.title = element_text(size = 14),legend.text = element_text(size = 14),
        axis.text.x = element_text(angle = 90, hjust = 1))



co_mol <-
as.data.frame(t(exprset[c("CTLA4","TIGIT","BTLA","CD48","PDCD1","LAG3","CD274","HAVCR2
",

"BTN2A2","LAIR1","BTN3A1","PDCD1LG2","BTN1A1","VTCN1","BTNL2",

"ICOS","TNFRSF9","CD70","CD80","TNFRSF13C","TMIGD2","TNFRSF13B",

"CD27","SLAMF1","TNFSF13B","CD86","TNFSF4","TNFRSF18","CD28","CD226",
```

```
"TNFSF9","TNFSF8","TNFSF18","HAVCR1","TNFRSF4","TNFSF15","TNFSF13",

"CD58","ICOSLG","TNFRSF8","TNFRSF14","BTNL8"),]))

co_mol_scale <- as.data.frame(scale(co_mol))

co_mol_scale$Group <- list$Group



compare2 <- as.data.frame(matrix(nrow = ncol(co_mol_scale)-1,ncol = 4))
colnames(compare2) <- c("Co-molecules","High ITH","Low ITH","pvalue")
rownames(compare2) <- colnames(co_mol_scale)[1:42]

compare2[,1] <- rownames(compare2)


for (i in rownames(compare2)) {
   compare2[i,2] <- mean(co_mol_scale[co_mol_scale$Group=="High",i])
   compare2[i,3] <- mean(co_mol_scale[co_mol_scale$Group=="Low",i])
   t <-
wilcox.test(co_mol_scale[co_mol_scale$Group=="High",i],co_mol_scale[co_mol_scale$Group=="Lo
w",i])
   compare2[i,4] <- t$p.value
}

compare2$FDR <- p.adjust(compare2$pvalue,method = "fdr")

order2 <- unique(compare2$`Co-molecules`)

colorcount <- 10
getpalette <- colorRampPalette(c("blue","white","red"))

pheatmap(compare2[,c("High ITH","Low ITH")],cluster_cols = F,cluster_rows = F,
            border_color = "grey",gaps_row = c(15),
            cellwidth = 10,cellheight = 10,fontsize = 10,
            color = getpalette(colorcount),breaks = seq(-0.5,0.5,by=0.1))
```