1.matrix download----

```
rm(list = ls())
library(GEOquery)
library(GEOmirror)
gse_number_21610 = "GSE21610"
gse_number_1145 = "GSE1145"
gse_number_29819 = "GSE29819"
eSet_21610 <- geoChina(gse_number_21610)
eSet_1145 <- geoChina(gse_number_1145)
eSet_29819 <- geoChina(gse_number_29819)
class(eSet_1145)
eSet_1145_1 = eSet_1145[[1]]
eSet_21610_1 = eSet_21610[[1]]
eSet_29819_1 = eSet_29819[[1]]
#(1)提取表达矩阵exp
exp_1145_all <- exprs(eSet_1145_1)
exp_1145_all[1:4,1:4]
exp_1145_all = log2(exp_1145_all+1)
```

boxplot(exp_1145_all)

```
exp_21610_all <- exprs(eSet_21610_1)
exp_21610_all[1:4,1:4]
exp_21610_all = log2(exp_21610_all+1)
```

boxplot(exp_21610_all)

```
exp_29819_all <- exprs(eSet_29819_1)
exp_29819_all[1:4,1:4]
exp_29819_all = log2(exp_29819_all+1)
```

boxplot(exp_29819_all)

```
library(tidyverse)
pd_21610 <- pData(eSet_21610_1) %>%
  filter(`disease status:ch1`!='ischemic cardiomyopathy',`vad status:ch1`!='after VAD support') %>%
  select(c(2,39,41))  %>%
  mutate(disease=c(rep('none',8),rep('dilated cardiomyopathy',21)),dataset='GSE21610') %>%
  select(1,4,5)
colnames(pd_21610)<-c("accession","disease","dataset")

pd_29819 <- pData(eSet_29819_1) %>%
  filter(`ventricle:ch1`=='left',`indication:ch1`!="Arrhythmogenic right ventricular cardiomyopathy") %>%
  select(c(2,35,36)) %>%
  mutate(disease=c(rep('dilated cardiomyopathy',7),rep('none',6)),dataset='GSE29819')%>%
```

```
  select(1,4,5)
colnames(pd_29819)<-c("accession","disease","dataset")

pd_1145_ctrl<-pData(eSet_1145_1) %>%
  filter(str_detect(`title`,'PA')) %>%
  filter(str_detect(`title`,'N'))
pd_1145_dcm<-pData(eSet_1145_1) %>%
  filter(str_detect(`title`,'Hs')) %>%
  filter(str_detect(`title`,'D'))
pd_1145<- rbind(pd_1145_ctrl,pd_1145_dcm) %>%
  select(2) %>%
  mutate(disease=c(rep('none',11),rep('dilated cardiomyopathy',12)),dataset='GSE1145')
colnames(pd_1145)<-c("accession","disease","dataset")
pd<-rbind(pd_1145,pd_21610,pd_29819)

exp_1145<- exp_1145_all[,colnames(exp_1145_all)%in% rownames(pd_1145)]
exp_21610<- exp_21610_all[,colnames(exp_21610_all)%in% rownames(pd_21610)]
exp_29819<- exp_29819_all[,colnames(exp_29819_all)%in% rownames(pd_29819)]
```

2.remove batch effect----

```
exp<-cbind(exp_1145,exp_21610,exp_29819)
pdf(file = 'figure/step1_exp_boxplot_raw.pdf',width = 8,height = 5)
par(oma=c(1,0,0,0))
p1<-boxplot(exp,las=2)
dev.off()

library(sva)
library(limma)
library(tidyverse)
mod=model.matrix(~as.factor(disease),data=pd)
batch<-c(rep('A',23),rep('B',29),rep('C',13))
combat_exp<-ComBat(exp,batch = batch,mod=mod) %>%
  normalizeBetweenArrays()

pdf(file = 'figure/step1_exp_boxplot_combat.pdf',width = 8,height = 5)
par(oma=c(1,0,0,0))
p2<-boxplot(combat_exp,las=2)
dev.off()
```

3.PCA plot----

```
library(FactoMineR)
library(factoextra)
dat=as.data.frame(t(combat_exp))
dat.pca <- PCA(dat, graph = FALSE)
pd$disease<-ifelse(pd[,2]=='none','Ctrl','DCM')
pca_plot <- fviz_pca_ind(dat.pca,
```

```
                  geom.ind = "point", # show points only (nbut not "text")
                  col.ind = pd$dataset, # color by groups
                  palette = c("red","blue","purple"),
                  addEllipses = TRUE, # Concentration ellipses
                  legend.title = "Datasets"
)
pca_plot
ggsave(pca_plot, filename = 'figure/step1_pca_dataset_combat.pdf',width = 6,height = 5)

dat_raw=as.data.frame(t(exp))
dat_raw.pca <- PCA(dat_raw, graph = FALSE)
pca_plot_raw <- fviz_pca_ind(dat_raw.pca,
                     geom.ind = "point", # show points only (nbut not "text")
                     col.ind = pd$dataset, # color by groups
                     palette = c("red","blue","purple"),
                     addEllipses = TRUE, # Concentration ellipses
                     legend.title = "Datasets"
)
pca_plot_raw
ggsave(pca_plot_raw,filename = 'figure/step1_pca_dataset_raw.pdf',width = 6,height = 5)

library(patchwork)
pca<- pca_plot_raw + pca_plot +plot_layout(guides = 'collect')
ggsave(pca,filename = 'figure/step1_pca_dataset.pdf',width = 6,height = 5)

library(stringr)
Group = factor(pd$disease,
           levels = c("Ctrl","DCM"))
ids <- AnnoProbe::idmap('GPL570')  #download probe annotations
```

4.DEGs analysis----

```
library(limma)
library(dplyr)
exp<-combat_exp
design=model.matrix(~Group)
fit=lmFit(exp,design)
fit=eBayes(fit)
deg=topTable(fit,coef=2,number = Inf) %>%
  mutate(probe_id=rownames(.))
head(deg)
```

remove duplication of probes corresponding to the same gene

```
ids = ids[!duplicated(ids$symbol),]
deg <- inner_join(deg,ids,by="probe_id")
head(deg)
nrow(deg)

logFC_t=log2(1.5)
P.Value_t = 0.05
k1 = (deg$adj.P.Val < P.Value_t)&(deg$logFC < -logFC_t)
k2 = (deg$adj.P.Val < P.Value_t)&(deg$logFC > logFC_t)
deg_k <- mutate(deg,change = ifelse(k1,"down",ifelse(k2,"up","stable")))
table(deg_k$change)
```

5. RRA----

```
library(limma)
library(tidyverse)
exp_1145<-normalizeBetweenArrays(exp_1145)
exp_21610<-normalizeBetweenArrays(exp_21610)
exp_29819<-normalizeBetweenArrays(exp_29819)



group_1145<-pd_1145$disease %>%
  factor(., levels = c("none","dilated cardiomyopathy"))
group_21610<-pd_21610$disease %>%
  factor(., levels = c("none","dilated cardiomyopathy"))
group_29819<-pd_29819$disease %>%
  factor(., levels = c("none","dilated cardiomyopathy"))

library(dplyr)
library(clusterProfiler)
library(org.Hs.eg.db)
ids = ids[!duplicated(ids$symbol),]
logFC_t=log2(1.5)
P.Value_t = 0.05

dega <- function(exp,group){
  design=model.matrix(~group)
  fit=lmFit(exp,design)
  fit=eBayes(fit)
  deg=topTable(fit,coef=2,number = Inf) %>%
    mutate(probe_id=rownames(.)) %>%
    inner_join(.,ids,by='probe_id')
  head(deg)
  nrow(deg)

  k1 = (deg$P.Value < P.Value_t)&(deg$logFC < -logFC_t)
  k2 = (deg$P.Value < P.Value_t)&(deg$logFC > logFC_t)
  deg <- mutate(deg,change = ifelse(k1,"down",ifelse(k2,"up","stable")))
  table(deg$change)
```

```
  s2e <- bitr(deg$symbol,
              fromType = "SYMBOL",
              toType = "ENTREZID",
              OrgDb = org.Hs.eg.db)
  deg <- inner_join(deg,s2e,by=c("symbol"="SYMBOL"))
}

deg_1145<-dega(exp_1145,group_1145)
deg_21610<-dega(exp_21610,group_21610)
deg_29819<-dega(exp_29819,group_29819)
deg_5406<-dega(exp_5406,group_5406)

down_1145<-filter(deg_1145,deg_1145$change=="down")%>%
  arrange(logFC) %>%
  dplyr::select(8) %>%
  unique()

down_21610<-filter(deg_21610,deg_21610$change=="down")%>%
  arrange(logFC) %>%
  dplyr::select(8) %>%
  unique()

down_29819<-filter(deg_29819,deg_29819$change=="down")%>%
  arrange(logFC) %>%
  dplyr::select(8) %>%
  unique()

up_1145<-filter(deg_1145,deg_1145$change=="up")%>%
  arrange(desc(logFC)) %>%
  dplyr::select(8) %>%
  unique()

up_21610<-filter(deg_21610,deg_21610$change=="up")%>%
  arrange(desc(logFC)) %>%
  dplyr::select(8) %>%
  unique()

up_29819<-filter(deg_29819,deg_29819$change=="up")%>%
  arrange(desc(logFC)) %>%
  dplyr::select(8) %>%
  unique()

library(RobustRankAggreg)
glist_down<-list(down_1145$symbol,down_21610$symbol,down_29819$symbol)
glist_up<-list(up_1145$symbol,up_21610$symbol,up_29819$symbol)
downs<-aggregateRanks(glist = glist_down,N=length(unique(unlist(glist_down))))
ups<-aggregateRanks(glist = glist_up,N=length(unique(unlist(glist_up))))
tmp_down<-as.data.frame(table(unlist(glist_down)))
tmp_up<-as.data.frame(table(unlist(glist_up)))
downs$Freq<-tmp_down[match(downs$Name,tmp_down[,1]),2]
ups$Freq<-tmp_up[match(ups$Name,tmp_up[,1]),2]
```

```
ups<-filter(ups,ups$Score<=0.05)
downs<-filter(downs,downs$Score<=0.05)
```

6.Add ENTREZID----

```
library(clusterProfiler)
library(org.Hs.eg.db)
s2e <- bitr(deg_k$symbol,
        fromType = "SYMBOL",
        toType = "ENTREZID",
        OrgDb = org.Hs.eg.db)

dim(deg_k)
deg_k <- inner_join(deg_k,s2e,by=c("symbol"="SYMBOL"))
dim(deg_k)
length(unique(deg_k$symbol))
```

7.Functional enrichment analysis----

```
library(clusterProfiler)
library(ggthemes)
library(org.Hs.eg.db)
library(dplyr)
library(ggplot2)
library(stringr)
library(enrichplot)
```

1.GO

```
table(deg_k$change)
deg<-deg_k
gene_up = deg$ENTREZID[deg$change == 'up']
gene_down = deg$ENTREZID[deg$change == 'down']
gene_diff = c(gene_up,gene_down)

#GO enrichment
ego_BP <- enrichGO(gene = gene_diff,
                   OrgDb= org.Hs.eg.db,
                   pAdjustMethod = 'none',
                   ont = "BP",
                   readable = TRUE)
ego_MF <- enrichGO(gene = gene_diff,
                   OrgDb= org.Hs.eg.db,
                   pAdjustMethod = 'none',
                   ont = "MF",
                   readable = TRUE)
```

```
ego_CC <- enrichGO(gene = gene_diff,
                   OrgDb= org.Hs.eg.db,
                   pAdjustMethod = 'none',
                   ont = "CC",
                   readable = TRUE)
```

KEGG enrichment

```
kk.diff <- enrichKEGG(gene         = gene_diff,
                      pAdjustMethod = 'none',
                      organism     = 'hsa')

p_BP<-dotplot(ego_BP, color='pvalue',x='Count',font.size=16)
p_MF<-dotplot(ego_MF, color='pvalue',x='Count',font.size=16)
p_CC<-dotplot(ego_CC, color='pvalue',x='Count',font.size=16)
p_kk<-dotplot(kk.diff, color='pvalue',x='Count',font.size=16)
ggsave(p_BP,filename = 'enrichment_BP.pdf',width = 12,height =5 )
ggsave(p_CC,filename = 'enrichment_CC.pdf',width = 12,height =5 )
ggsave(p_MF,filename = 'enrichment_MF.pdf',width = 12,height =5 )
ggsave(p_kk,filename = 'enrichment_kk.pdf',width = 12,height =5 )
```

8.Immune cell infiltration analysis----

```
library(xCell)
deg<-deg_k
deg<-deg[,-ncol(deg)] %>% unique(.)
exp.file <- as.data.frame(combat_exp) %>%
  filter(.,rownames(.)%in% deg$probe_id)
rownames(exp.file)<-deg$symbol[match(rownames(exp.file),deg$probe_id)]
exp.file<-as.matrix(exp.file)
xCell <- xCellAnalysis(exp.file)

xCell<-t(xCell) %>%
  as.data.frame()
xCell_tpms<-xCell %>%t()
dim(xCell_tpms)
```

visulization

```
library(RColorBrewer)
library(tidyr)
library(ggpubr)
xCell_tpms1 <- xCell_tpms %>%
  as.data.frame() %>%
  rownames_to_column("cell_type") %>%
  gather(sample, fraction, - cell_type)
xCell_tpms2<-xCell_tpms1
xCell_tpms2$group<-rep(pd$disease,each=67)
```

```
p_value<-compare_means(fraction~group,xCell_tpms2,method = 't.test',group.by ="cell_type")

lymphoid_myeloid <- c("CD8+ T-cells","CD8+ Tcm","Th1 cells","Macrophages M1", "cDC")
stromal<-c("Adipocytes","Fibroblasts","mv Endothelial
cells","Neurons","StromaScore","MicroenvironmentScore",
          "Smooth muscle", "Chondrocytes","Skeletal muscle", "Myocytes")
stem<-c("HSC","CLP","CMP","GMP","Megakaryocytes")
p_value_1<-filter(p_value, p_value$p.signif!="ns")
xCell_tpms3<-filter(xCell_tpms2,xCell_tpms2$cell_type %in% p_value_1$cell_type)
xCell_tpms3_lm<-filter(xCell_tpms3,xCell_tpms3$cell_type %in% lymphoid_myeloid)
xCell_tpms3_str<-filter(xCell_tpms3,xCell_tpms3$cell_type %in% stromal)
xCell_tpms3_ste<-filter(xCell_tpms3,xCell_tpms3$cell_type %in% stem)

lm<- ggplot(xCell_tpms3_lm,
            aes(cell_type,
                fraction,
                fill=group)) +
  geom_boxplot(outlier.shape = 21,
               color = "black") +
  theme_bw() +
  labs(x=NULL,
       y = "xCell_Score") +
  theme(axis.text.x = element_text(angle = 45,hjust = 0.5,vjust = 0.5)) +
  theme(axis.ticks.x = element_blank())+
  stat_compare_means(label = 'p.signif',method = 't.test',hide.ns = T,size=8,label.y.npc = 0.95 )

str<- ggplot(xCell_tpms3_str,
             aes(cell_type,
                 fraction,
                 fill=group)) +
  geom_boxplot(outlier.shape = 21,
               color = "black") +
  theme_bw() +
  labs(x=NULL,
       y = "xCell_Score") +
  theme(axis.text.x = element_text(angle = 45,hjust = 0.5,vjust = 0.5)) +
  theme(axis.ticks.x = element_blank())+
  stat_compare_means(label = 'p.signif',method = 't.test',hide.ns = T,size=8, label.y.npc = 0.95)

ste<- ggplot(xCell_tpms3_ste,
             aes(cell_type,
                 fraction,
                 fill=group)) +
  geom_boxplot(outlier.shape = 21,
               color = "black") +
  theme_bw() +
  labs(x=NULL,
       y = "xCell_Score") +
  theme(axis.text.x = element_text(angle = 45,hjust = 0.5,vjust = 0.5)) +
```

```
  theme(axis.ticks.x = element_blank())+
  stat_compare_means(label = 'p.signif',method = 't.test',hide.ns = T,size=8,label.y.npc = 0.95)

library(patchwork)
p_all<-(lm + ste) / str + plot_annotation(tag_levels = 'A') + plot_layout(guides = 'collect')
ggsave(p_all,filename = 'xCell_cell_fraction_boxplot.pdf',width = 12,height = 10)
```

gene and immune cells correlation analysis

```
library(psych)
library(tidyverse)
library(xCell)
library(corrplot)
xCell <- xCellAnalysis(exp.file)

immune_cell <-c("iDC","Eosinophils","Macrophages M2" ,"Macrophages","Th1 cells","Plasma
cells","NKT","Monocytes",
                "Macrophages M1","aDC","DC", "cDC","Basophils","CD8+ T-cells","CD8+ Tcm","pDC","NK cells",
                "Th2 cells","B-cells","CD4+ T-cells","Mast cells","Tregs")

xCell_sub<-as.data.frame(xCell) %>%
  filter(rownames(.)%in% immune_cell) %>%
  t()

library(corrplot)
M<-cor(xCell_sub)

M_tmp<-M
M_tmp[M_tmp==1]<-NA

p.mat<-cor.mtest(xCell_sub)
p.mat<-p.mat$p
p.mat.tmp<-p.mat
p.mat.tmp[p.mat.tmp>=0.05]<-NA
p.mat.tmp<- -log10(p.mat.tmp)
```

Replace correlation coefficient indexes where p>=0.05 to NA

```
p.mat.tmp_2<-p.mat
p.mat.tmp_2[p.mat.tmp_2>=0.05]<-NA

library(paletteer)
my_color = rev(paletteer_d("RColorBrewer::RdYlBu"))
my_color = colorRampPalette(my_color)(10)

pdf(file="xCell_immunecell_relationships.pdf",width=15,height=16)
par(oma=c(0,0,0,0))
corrplot(M_tmp,order = "original",addCoef.col = "black",type="upper",method = 'square',na.label =
'square',na.label.col = 'white',
```

```
        tl.pos = 'full',tl.cex = 2,tl.col = 'black',col = my_color,
        p.mat = p.mat,insig = 'blank',pch.col='blank',diag = TRUE)
corrplot(p.mat.tmp,order = "original",type="lower",method = 'circle',add=TRUE,
        p.mat=p.mat.tmp_2,sig.level = c(0.001,0.01,0.05),insig = 'label_sig',na.label = 'square',na.label.col =
'white',col = my_color,
        pch.cex = 2,tl.pos = 'n',diag = FALSE,is.corr = FALSE)
dev.off()
```

9.ROC curve----

```
library(glmnet)
library(foreign)
library(tidyverse)
library(pROC)
pd$group<-ifelse(pd$disease=='Ctrl',0,1)
exp<-t(exp.file) %>%
  as.data.frame() %>%
  mutate(group=pd$group)

#CCL2
model_ccl2<-glm(group~CCL2,data=exp,family=binomial())
summary(model_ccl2)
exp$predvalue_ccl2<-predict(model_ccl2,type="response")
ROC_ccl2 <- roc(exp$group,exp$predvalue)
auc(ROC_ccl2)
ci(auc(ROC_ccl2))
tmp_predvalue<-data.frame(group=exp$group,CCL2=exp$predvalue_ccl2)

#CCL5
exp<-exp[,-ncol(exp)]
model_ccl5<-glm(group~CCL5,data=exp,family=binomial())
summary(model_ccl5)
exp$predvalue_ccl5<-predict(model_ccl5,type="response")
ROC_ccl5 <- roc(exp$group,exp$predvalue)
auc(ROC_ccl5)
ci(auc(ROC_ccl5))
tmp_predvalue$CCL5<-(exp$predvalue_ccl5)

#TLR2
exp<-exp[,-ncol(exp)]
model_TLR2<-glm(group~TLR2,data=exp,family=binomial())
summary(model_TLR2)
exp$predvalue_TLR2<-predict(model_TLR2,type="response")
ROC_TLR2 <- roc(exp$group,exp$predvalue)
auc(ROC_TLR2)
ci(auc(ROC_TLR2))
tmp_predvalue$TLR2<-(exp$predvalue_TLR2)
```

```
#CCL2+CCL5+TLR2
exp<-exp[,-ncol(exp)]
model_all<-glm(group~CCL2+CCL5+TLR2,data=exp,family=binomial())
summary(model_all)
exp$All<-predict(model_all,type="response")
ROC_all <- roc(exp$group,exp$All)
auc(ROC_all)
ci(auc(ROC_all))
tmp_predvalue$all<-(exp$predvalue_all)

Roc<-list(ROC_ccl2,ROC_ccl5,ROC_TLR2,ROC_all)
gene<-c('CCL2','CCL5','TLR2','CCL2+CCL5+TLR2')
Main<-c('CCL2','CCL5','TLR2','Diagnosis Model')
pdf(file = 'ROC.pdf',width = 10,height = 2.5)
par(mfrow=c(1,4))
for (i in 1:4) {
  plot(Roc[[i]],print.auc=TRUE,auc.polygon=TRUE,auc.polygon.col='skyblue',col='darkgrey',lwd=1,
print.thres=FALSE,
       identity.lwd=1,identity.lty=2,legacy.axes=TRUE,print.auc.cex=1,print.auc.col='black',
       main=Main[i])
  labels(x='x','Fontname', 'Times New Roman','FontSize',12)
}
dev.off()
```