library("RColorBrewer")

library("ggplot2")

library(tidyverse)

library(dplyr)

library(ggalt)

library(scales)

library(gridExtra)

library(ggpubr)

library(xCell)

library(GSEABase)

'%!in%' <- function(x,y)!('%in%'(x,y))

###########################

### MAKE DEA DOT PLOT ###

###########################

mycounts <- read.csv("DEA\_LA\_deadvsalive.csv")

tmp <- mycounts %>% filter(Name %in% c("FOLR2","GAS6","CLEC10A","STAB1","F13A1","TMEM176B"))

pdf("week1and2\_DEA\_LA\_adjpvalue.pdf",width=8, height=4)

ggplot(tmp, aes(x=Name, y=logFC))+

geom\_point(aes(size=adj.P.Val))+

facet\_grid(cols=vars(visit\_categ))+

coord\_flip()+theme\_pubr()+

labs(y="z score")+theme(axis.title.y=element\_blank())

dev.off()

###########################

### MAKE xCell results LA ###

###########################

metadatafile\_filename="sample\_metadata.csv"

metadata = read.csv(metadatafile\_filename) %>% filter(sample\_type %in% c('TA','BAL'))

mycounts <- read.table("LA\_hosttranscriptome\_readcounts.txt", header=TRUE, row.names=1, as.is=TRUE, sep='\t')

namlikecell <- GeneSet(setName="namlikecell",geneIds=c("FOLR2","F13A1","TMEM176B","GAS6","STAB1","CLEC10A"))

geneset <- GeneSetCollection(namlikecell)

#rawEnrichmentAnalysis: reads a gene expression matrix and returns an enrichment score for each of the 64 cell types across the input gene expression samples. For its calculations, xCell considers 10,808 common genes.

scores <- rawEnrichmentAnalysis(mycounts,signatures=geneset,genes=xCell.data$genes)

#transformScores: transform scores from raw enrichment scores to a linear scale that resembles percentages

tmp\_fitvals <- read.csv("spill.fv\_newcalibrationparams.csv")

tscores = transformScores(scores,tmp\_fitvals)

tscores\_t <- as.data.frame(t(tscores))

tscores\_t$yl\_sample\_id <- rownames(tscores\_t)

merged\_tscores\_t <- merge(tscores\_t,metadata,on="yl\_sample\_id")

pal\_deadoralive <- c("Alive"="#776bcd","Dead"="firebrick1")

txt\_size<-12

ggsave(filename=paste0("transformScores\_justnamlike\_boxplotDA.pdf"),

ggplot(merged\_tscores\_t, aes(x=dead\_or\_alive, y=namlikecell\*100,color=factor(dead\_or\_alive)))+

geom\_boxplot(aes(fill=dead\_or\_alive,color=dead\_or\_alive), width=0.5,position=position\_dodge(0.8),outlier.shape = NA,alpha=0.3)+

geom\_point(position=position\_jitterdodge(dodge.width=0.5),size=0.8,alpha=0.8)+

scale\_color\_manual(values=pal\_deadoralive,name="")+scale\_fill\_manual(values=pal\_deadoralive,name="")+

labs(y='Log Nam-like cell')+

theme\_pubr()+theme(legend.position="none",plot.title=element\_text(size=txt\_size,hjust=0.5,vjust=2.5),axis.ticks=element\_line(size=0.3),plot.margin=margin(10,1,5.5,5.5,"pt"),axis.title.x=element\_blank(),axis.line=element\_line(size=0.2),axis.text.y=element\_text(size=txt\_size,,color="black"),axis.title.y=element\_text(size=txt\_size,margin=margin(r=10),vjust=1.5),axis.ticks.x=element\_blank(),strip.background.x=element\_blank(), strip.placement = "outside",strip.text.x=element\_blank(),panel.spacing=unit(0.5,"lines"))+

stat\_compare\_means(aes(label =ifelse(after\_stat(p.format)>0.05,"",..p.signif..)), method="wilcox.test",label.x=1.5,vjust=0.6,size=10)+

stat\_compare\_means(aes(label =ifelse(after\_stat(p.format)>0.05,..p.signif..,"")), method="wilcox.test",label.x=1.5,vjust=0.04,size=7)+

scale\_y\_log10(labels=trans\_format('log10', math\_format(10^.x))),

height=4,width=4)

############################

### MAKE pathway heatmap ###

############################

# filename\_v1 <- "IPA\_LA\_week1.csv"

# filename\_v2 <- "IPA\_LA\_week2.csv"

filename\_v1 <- "IPA\_Blood\_week1.csv"

filename\_v2 <- "IPA\_Blood\_week2.csv"

#IPAq0.2,byzscore\_0.05

# outputplot\_filename <-paste0("heatmap\_LA\_IPA.pdf")

outputplot\_filename <-paste0("heatmap\_Blood\_IPA.pdf")

labels\_for\_columns <-c("Week 1","Week 2")

file\_v1 <- read.csv(filename\_v1, sep=,) %>% select("Pathways","z.score","NegLog.p.value")

file\_v2 <- read.csv(filename\_v2, sep=,) %>% select("Pathways","z.score","NegLog.p.value")

file\_v1$z.score <- as.numeric(file\_v1$z.score)

file\_v2$z.score <- as.numeric(file\_v2$z.score)

colnames(file\_v1)[which(colnames(file\_v1) == "z.score")]<-"week1.zscore"

colnames(file\_v2)[which(colnames(file\_v2) == "z.score")]<-"week2.zscore"

colnames(file\_v1)[which(colnames(file\_v1) == "NegLog.p.value")]<-"week1.neg.log.p.value"

colnames(file\_v2)[which(colnames(file\_v2) == "NegLog.p.value")]<-"week2.neg.log.p.value"

merged\_df <- merge(file\_v1, file\_v2, by = "Pathways", all = TRUE)

dat <- merged\_df

dat <- dat %>% mutate(Pathways = Pathways)

# #-log(0.2)= 0.69 so maybe i wont keep any rows where p value less than that

dat$week1.zscore <- as.numeric(dat$week1.zscore)

dat$week2.zscore <- as.numeric(dat$week2.zscore)

dat$week1.neg.log.p.value <- as.numeric(dat$week1.neg.log.p.value)

dat$week2.neg.log.p.value <- as.numeric(dat$week2.neg.log.p.value)

dat <- dat %>% filter(Pathways %in% c('Neutrophil degranulation','IL-6 Signaling','Coronavirus Pathogenesis Pathway','Interferon Signaling'))

dat$is\_significant\_w1 <- ifelse(dat$week1.neg.log.p.value > 1.3,TRUE,FALSE)

dat$is\_significant\_w2 <- ifelse(dat$week2.neg.log.p.value > 1.3,TRUE,FALSE)

dat <- dat %>% arrange(desc(week1.zscore), desc(week2.zscore))

#Make sure at least one row has a z score and significant p value

dat <- dat %>% filter(!is.na(week1.zscore) | !is.na(week2.zscore))

dat\_m\_long <- pivot\_longer(data = dat%>% select(Pathways,week1.zscore,week2.zscore,is\_significant\_w1,is\_significant\_w2),

cols = starts\_with("week"),

names\_to = "week",

values\_to = "zscore")

dat\_m\_long <- dat\_m\_long %>% mutate(is\_significant=case\_when(

week == "week1.zscore" ~ is\_significant\_w1,

week == "week2.zscore" ~ is\_significant\_w2

))

dat\_m\_long <- dat\_m\_long %>% select(Pathways,week,is\_significant,zscore)

dat\_m\_long$is\_significant <- ifelse(dat\_m\_long$is\_significant,"Y","N")

dat\_m\_long$is\_significant <- ifelse(is.na(dat\_m\_long$is\_significant),"N",dat\_m\_long$is\_significant)

dat\_m\_long$week = factor(dat\_m\_long$week,levels=c("week1.zscore","week2.zscore"),labels=labels\_for\_columns)

pathway\_list <- unique(dat\_m\_long$Pathways)

dat\_m\_long$Pathways <- factor(dat\_m\_long$Pathways, levels=rev(pathway\_list))

#Use this when want z score to be NA when p value not significant

ggsave(filename=paste0(outputplot\_filename),

ggplot(data = dat\_m\_long, mapping = aes(x = week,y = Pathways,fill = zscore)) +

geom\_tile(color="black",lwd=1) +

geom\_point(aes(x=week,y=Pathways,shape=is\_significant),color="black")+

scale\_shape\_manual(labels=c("Y","N"), limits=c("Y","N"), values=c(Y=42,N=32))+

xlab(label = "Sample")+

scale\_fill\_gradient2(name="Z-score", low = "#1432F6",high = "#FB9937",na.value = "white")+

scale\_x\_discrete(position = "top")+

theme\_pubr()+

theme(axis.title.x=element\_blank(),axis.title.y=element\_blank(),axis.line=element\_blank(),axis.text.y=element\_text(color="black"),axis.text.x=element\_text(angle = 45, vjust=1,hjust=-0.1,color="black"),axis.ticks.x=element\_line(size=0.6),axis.ticks.length=unit(0.2,"cm"))+

theme(plot.margin = unit(c(0.2, 0.4, 0.2, 0.2),"inches"))+

theme(legend.position="none"),

height=2,,width=5)