uppressPackageStartupMessages(library("optparse"))

suppressPackageStartupMessages(library("stats"))

usage = "Rscript %prog -f -t -n -o

example: Rscript %prog

-f /users/ablife/ablife-R/Bar/Bar\_Rpkm/latest/Mapping\_distribution.txt

-t Mapping\_distribution

-n Mapping\_distribution

-o ./"

option\_list <- list(

make\_option(c("-f", "--file"),action = "store",type = "character",

help = "The Input file"),

make\_option("--log",action = "store\_true",default = FALSE,

help = "handel the data using log"),

make\_option(c("-t", "--title"),action = "store",type = "character",

help = "The title of outimage"),

make\_option(c("-n", "--filename"),action = "store",type = "character",

help = "The name of outimage"),

make\_option(c("-X","--Xaxis"),action = "store",type = "character",default = "Type",

help = "The label of X axis"),

make\_option(c("-Y","--Yaxis"),action = 'store',type = "character",default = "Percentage(%)",

help = "The label of Y axis"),

make\_option("--ymax",action = "store",type = "integer",default = "100",

help = "The max of y axis"),

make\_option(c("-o", "--outdir"),action = "store",type = "character",default = "./",

help = "The outdir")

)

opt <- parse\_args(OptionParser(option\_list = option\_list,usage=usage)) #####解释器

setwd(opt$outdir) ##Set the out path

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

#### （Load Package）

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

library(ggplot2)

library(reshape2)

library(plotrix)

library(methods)

library(RColorBrewer)

library("ggsci")

library("ggthemes")

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

#### （Load Color）

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

colour <- c('#85A2EF','#D285EF','#A2EF85','#4682B4','#A0522D','#87CEEB','#6B8E23','#6A5ACD','#E59B95','#EFD285','#B4B643','#2E9AFE','#A1DDBB','#FF8C00')

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

##（Deal with Data）

title <- gsub('\_',' ',opt$title)

Xaxis <- gsub('\_',' ',opt$Xaxis)

Yaxis <- gsub('\_',' ',opt$Yaxis)

data <- read.table(opt$file,header=T) #read.table must with header

colname <- colnames(data) #

dim\_data <- dim(data) #

ymax <- as.numeric(opt$ymax)

sample <-colname[2:dim\_data[2]] #

newdata1 <- data

for (i in 2:dim\_data[2])

{sum <- sum(data[,i])

for(j in 1:dim\_data[1])

{mean <- data[j,i]/sum

mean <- round(mean\*100,2)

percentage <- paste(mean,'%',sep='')

newdata1[j,i]<-percentage

# mean <- log10(mean)+4 #

data[j,i]<-mean}

}

newdata <- melt(data,id.vars=colname[1],measure.vars = c(colname[2:dim\_data[2]]))

ymax = max(newdata[,3]) + 10

if(ymax > 100){

ymax =100

}

Percent <- paste(newdata[,3],'%',sep='')

head(newdata)

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

###Plot Theme for ABLife

###theme(),Tha last term without comma

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

ablife\_theme\_bar <- function(base\_size = 12){

library(grid) ####for using unit function

theme(

plot.title = element\_text(size=12,lineheight = 10,colour="#000000"),

axis.title.x = element\_text(size=12,colour = "#000000"),

axis.title.y = element\_text(size=12,colour = "#000000"),

axis.text.x = element\_text(size = 12,colour = "#000000"),

axis.text.y = element\_text(size = 12 ,colour = "#000000"),

legend.title = element\_text(size = 12),

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

legend.key.size = unit(0.5,"cm"),

panel.background = element\_rect(fill = "white",colour = NA),

panel.border = element\_rect(size = 1,colour = "#000000",fill =NA)

# panel.border = element\_rect(size = 1,colour = "#8B8B8B",fill =NA)

# panel.grid.major = element\_line(size=0.5,colour = "#BFBFBF"),

# panel.grid.minor = element\_line(size=0.1,colour = "#7F7F7F")

)

}

theme\_paper <- theme(

# panel.border = element\_rect(fill = NA,colour = "black"),

panel.grid.major = element\_line(size = 0.2,linetype = "dashed"),

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

panel.grid.minor = element\_line(colour = "grey90",size = 0.2,linetype = "dashed"),

# axis.text.x= element\_text(vjust = 1,hjust = 1, angle = 45),

legend.position = "top",

legend.direction = "horizontal",

legend.key = element\_rect(fill = 'white', color = 'white'),

legend.key.size = unit(0.5, "cm"),

# legend.position = c(0.82, 0.68),

# legend.position = "top",

# legend.direction = "vertical",

legend.title = element\_blank(),

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

# plot.margin = margin(1,1,1,1,unit="cm"),

legend.text = element\_text(size = 11),

axis.title = element\_text(size = 12, face = "bold"),

plot.title = element\_text(size = 12, face = "bold"),

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

)

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

###Plot by ggplot2

ggplot(newdata,aes(x = newdata[,1],y=newdata[,3],fill = newdata[,2],stat = "identity"))+

geom\_bar(width = 0.9,stat = "identity",position = "dodge")+labs(x=Xaxis,y= Yaxis)+

# geom\_text(aes(label=Percent),hjust=0,angle = -3,position = position\_dodge(.9),size=5)+

scale\_y\_continuous(limits=c(0,ymax))+

coord\_flip()+

# scale\_fill\_manual(name="Name",values = colour[1:dim\_data[2]-1])+

scale\_fill\_nejm()+

# geom\_text(aes(label=newdata[,3]),vjust =1.5,position = position\_dodge(.9),size=2)+

theme\_bw() + theme\_paper

#scale\_fill\_discrete(name="Sample\_name") ##label，breaks

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

###Save Plot File

ggsave(file = paste(opt$filename,"\_Bar.pdf",sep=''), width = 180,height = 120,dpi = 450,units = "mm")

ggsave(file = paste(opt$filename,"\_Bar.png",sep=''), width = 180,height = 120,dpi = 450,units = "mm")