1、rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,purrr,TwoSampleMR)

# 如果没有 devtools 包,则安装 devtools 包

if (!require("devtools")) {

install.packages("devtools")

} else {}

# 如果没有 data.table 包,则安装 data.table 包

if (!require("data.table")) {

install.packages("data.table")

} else {}

# 如果没有 TwoSampleMR 包,则从 GitHub 安装 TwoSampleMR 包

if (!require("TwoSampleMR")) {

devtools::install\_github("MRCIEU/TwoSampleMR")

} else {}

# 如果没有 TwoSampleMR 包,则从 GitHub 安装 TwoSampleMR 包

if (!require("MRInstruments")) {

devtools::install\_github("MRCIEU/MRInstruments")

} else {}

# 设置仪器变量文件

expofile = "immID.txt"

# 设置结果变量文件

outcomefile = "finngen\_R11\_M13\_SHOULDERBURSITIS"

# 设置显著性阈值

or\_pfilter = 1

# 读取工具变量数据

immucell\_data = fread(expofile,header = T,sep = "\t")

# 提取id为向量

immucell\_ID = as.vector(immucell\_data$id)

immucell\_ID

# 取id子集

#immucell\_ID = immucell\_ID[1:3]

immucell\_ID

# 使用 map 循环处理每个id

Biofsci <- map\_df(immucell\_ID, function(i) {

tryCatch({

# 提取仪器变量

expo\_data <- extract\_instruments(outcome = i,

p1 = 1e-5,

clump = TRUE,

p2 = 1e-5,

r2 = 0.001,

kb = 10000)

# 提取结果变量

mydata = fread("finngen\_R11\_M13\_SHOULDERBURSITIS.txt",header = T,sep = "\t")

fwrite(mydata,"finngen\_R11\_M13\_SHOULDERBURSITIS.txt")

colnames(mydata)

outc\_data <- read\_outcome\_data(

snps = expo\_data$SNP,

filename = "finngen\_R11\_M13\_SHOULDERBURSITIS.txt",

sep = ",", # "/t"

snp\_col = "rsids",

beta\_col = "beta",

se\_col = "sebeta",

effect\_allele\_col = "ref",

other\_allele\_col = "alt",

#eaf\_col = "A1FREQ",

chr\_col = "#chrom",

pval\_col = "pval")

# 匹配变量

harm\_data <- harmonise\_data(exposure\_dat = expo\_data,

outcome\_dat = outc\_data, action = 2)

# MR分析

mr\_result <- mr(harm\_data)

# 生成OR值

MRresult\_or = generate\_odds\_ratios(mr\_result)

# 筛选显著结果

if (MRresult\_or$pval[3] < or\_pfilter) {

# 返回一个数据框，包含id和pvalue

return(data.frame(id = i, pvalue = MRresult\_or$pval[3]))

} else {

# 如果不显著，返回空数据框

return(data.frame())

}

}, error = function(e) {

# 处理可能的错误

return(data.frame())

})

})

# 输出结果表

write.table(Biofsci, "MRresult3.txt", sep = "\t", quote = FALSE, row.names = FALSE)

2、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,dplyr)

# 创建一个空列表来存储所有数据框

df\_list <- list()

# 遍历所有TXT文件

files <- list.files(pattern="MRresult\*")

for(f in files){

# 读取每个TXT文件为数据框

df <- read.table(f, header=TRUE)

# 添加到列表中

df\_list <- c(df\_list, list(df))

}

# 通过列名合并多个数据框

combined\_df <- Reduce(function(x,y) rbind(x,y), df\_list)

# 去除重复的id

distinct\_df <- combined\_df %>% distinct(id, .keep\_all=TRUE)

# 查看去重后的数据

head(distinct\_df)

write.table(distinct\_df,"allMRresult.txt",sep = "\t",quote = F,row.names = F)

3、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,ggplot2,TwoSampleMR)

#install.packages("devtools")

#devtools::install\_github("MRCIEU/TwoSampleMR")

outcomefile = "ebi-a-GCST90016564"

or\_pfilter = 0.05

MRresult\_data=fread("allMRresult.txt",header = T,sep = "\t")

immucell\_ID=as.vector(MRresult\_data$id)

immucell\_ID=immucell\_ID[14:14]

# 载入数据

# immucell\_ID为免疫细胞ID

# 提取曝露因素SNP

# p1为LD阈值,clump为True表示进行LD泯灭,p2为p值阈值,r2为相关系数阈值,kb为窗口大小

for (i in immucell\_ID) {

expo\_data <- extract\_instruments(outcome = i, p1 = 1e-5,

clump = T, p2 = 1e-5,

r2 = 0.001, kb = 10000)

# 提取结果因变量数据

outc\_data <- extract\_outcome\_data(snps = expo\_data$SNP,

outcomes = outcomefile)

# 调和曝露因素和结果数据

harm\_data <- harmonise\_data(exposure\_dat = expo\_data,

outcome\_dat = outc\_data,

action = 2)

# 进行MR分析

mr\_result <- mr(harm\_data)

# 生成优比值

result\_or <- generate\_odds\_ratios(mr\_result)

# 创建输出目录

dir.create(i)

filename = i

# 输出结果

write.table(harm\_data, file = paste0(filename,"/harmonise.txt"),

row.names = F, sep = "\t", quote = F)

write.table(result\_or[,5:ncol(result\_or)],

file = paste0(filename,"/OR.txt"),

row.names = F, sep = "\t", quote = F)

# 保存 odds ratio 数据，并使用 filename 作为前缀

write.table(result\_or[, 5:ncol(result\_or)],

file = paste0(filename, "\_OR.txt"),

row.names = FALSE, sep = "\t", quote = FALSE)

# 绘制散点图

p1 <- mr\_scatter\_plot(mr\_result, harm\_data)

ggsave(p1[[1]], file=paste0(filename,"/scatter.pdf"),

width=8, height=8)

# 进行多重比较校正

pleiotropy <- mr\_pleiotropy\_test(harm\_data)

write.table(pleiotropy, file = paste0(filename,"/pleiotropy.txt"),

sep = "\t", quote = F)

# 进行异质性检验

heterogeneity <- mr\_heterogeneity(harm\_data)

write.table(heterogeneity, file = paste0(filename,"/heterogeneity.txt"),

sep = "\t", quote = F)

# MR-PRESSO

presso <- run\_mr\_presso(harm\_data, NbDistribution = 1000)

capture.output(presso, file = paste0(filename,"/presso.txt"))

# 单个SNP分析

singlesnp\_res <- mr\_singlesnp(harm\_data)

singlesnpOR <- generate\_odds\_ratios(singlesnp\_res)

write.table(singlesnpOR, file=paste0(filename,"/singlesnpOR.txt"),

row.names = F, sep = "\t", quote = F)

# 森林图

p2 <- mr\_forest\_plot(singlesnp\_res)

ggsave(p2[[1]], file=paste0(filename,"/forest.pdf"), width=8, height=8)

# Leave-one-out分析

sen\_res <- mr\_leaveoneout(harm\_data)

p3 <- mr\_leaveoneout\_plot(sen\_res)

ggsave(p3[[1]], file=paste0(filename,"/sensitivity-analysis.pdf"),

width=8, height=8)

# Funnel plot检验，如果不存在publication bias,那么数据点应该对称分布在两侧,呈漏斗形分布。

res\_single <- mr\_singlesnp(harm\_data)

p4 <- mr\_funnel\_plot(singlesnp\_res)

ggsave(p4[[1]], file=paste0(filename,"/funnelplot.pdf"), width=8, height=8)

}

4、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,ggplot2,TwoSampleMR,purrr)

#install packages

#install.packages("devtools")

#devtools::install\_github("MRCIEU/TwoSampleMR")

# 定义文件名和参数

MRresultfile = "allMRresult.txt" # MR结果文件

expofile = "ebi-a-GCST90013980" # 外部暴露数据文件

or\_pfilter = 1 # 滤除 odds ratio 的阈值

# 读取 MR 结果数据

MRresult\_data = read.table(MRresultfile, header = TRUE, sep = "\t")

immucell\_ID = as.vector(MRresult\_data$id) # 提取id列的数据

immucell\_ID

immucell\_ID=immucell\_ID[1:54]

# 使用 map 循环遍历 immucell\_ID 中的每个元素

results <- map(immucell\_ID, function(i) {

# 提取外部暴露数据

expo\_data <- extract\_instruments(outcome = expofile,

p1 = 5e-6,

clump = TRUE,

p2 = 5e-6,

r2 = 0.001,

kb = 10000)

# 提取结果数据

outc\_data <- extract\_outcome\_data(snps = expo\_data$SNP, outcomes = i)

# 数据协调处理

harm\_data <- harmonise\_data(exposure\_dat = expo\_data, outcome\_dat = outc\_data, action = 2)

# 进行 Mendelian Randomization 分析

mr\_result <- mr(harm\_data)

result\_or = generate\_odds\_ratios(mr\_result)

# 如果结果的第三个p值小于设定的阈值，执行以下操作

if (result\_or$pval[3] < or\_pfilter) {

dir.create(i) # 创建一个目录

filename = i

# 保存协调处理数据

write.table(harm\_data, file = paste0(filename, "/harmonise.txt"), row.names = FALSE, sep = "\t", quote = FALSE)

# 保存 odds ratio 数据

write.table(result\_or[, 5:ncol(result\_or)],

file = paste0(filename, "/ORReverse.txt"),

row.names = FALSE, sep = "\t", quote = FALSE)

# 保存 odds ratio 数据，并使用 filename 作为前缀

write.table(result\_or[, 5:ncol(result\_or)],

file = paste0(filename, "\_ORReverse.txt"),

row.names = FALSE, sep = "\t", quote = FALSE)

# 进行 MR 的共线性检验

pleiotropy = mr\_pleiotropy\_test(harm\_data)

write.table(pleiotropy, file = paste0(filename, "/pleiotropy.txt"), sep = "\t", quote = FALSE)

# 进行 MR 的异质性检验

heterogeneity = mr\_heterogeneity(harm\_data)

write.table(heterogeneity, file = paste0(filename, "/heterogeneity.txt"), sep = "\t", quote = FALSE)

# 运行 MR PRESSO 分析

presso = run\_mr\_presso(harm\_data, NbDistribution = 1000)

capture.output(presso, file = paste0(filename, "/presso.txt"))

# 生成 MR scatter plot 并保存

p1 = mr\_scatter\_plot(mr\_result, harm\_data)

ggsave(p1[[1]], file = paste0(filename, "/scatter.pdf"), width = 8, height = 8)

# 进行 MR 单一SNP分析

singlesnp\_res = mr\_singlesnp(harm\_data)

singlesnpOR = generate\_odds\_ratios(singlesnp\_res)

# 保存单一SNP的 odds ratio 数据

write.table(singlesnpOR, file = paste0(filename, "/singlesnpOR.txt"), row.names = FALSE, sep = "\t", quote = FALSE)

# 生成 MR forest plot 并保存

p2 = mr\_forest\_plot(singlesnp\_res)

ggsave(p2[[1]], file = paste0(filename, "/forest.pdf"), width = 8, height = 8)

# 进行 MR leave-one-out 分析

sen\_res = mr\_leaveoneout(harm\_data)

# 生成 MR sensitivity-analysis 图并保存

p3 = mr\_leaveoneout\_plot(sen\_res)

ggsave(p3[[1]], file = paste0(filename, "/sensitivity-analysis.pdf"), width = 8, height = 8)

# 重新进行 MR 单一SNP分析

res\_single = mr\_singlesnp(harm\_data)

# 生成 MR funnel plot 并保存

p4 = mr\_funnel\_plot(singlesnp\_res)

ggsave(p4[[1]], file = paste0(filename, "/funnelplot.pdf"), width = 8, height = 8)

}

})

5、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,dplyr)

# 创建一个空列表来存储所有数据框

df\_list <- list()

# 遍历所有TXT文件

files <- list.files(pattern="\*\_OR.txt")

for(f in files){

# 读取每个TXT文件为数据框

df <- fread(f, header=TRUE,sep = "\t",quote = F,)

df$id <- f

# 使用 gsub 去掉 "\_OR.txt"

df$id <- gsub("\_OR.txt", "", df$id)

# 添加到列表中

df\_list <- c(df\_list, list(df))

}

# 通过列名合并多个数据框

combined\_df <- Reduce(function(x,y) rbind(x,y), df\_list)

write.table(combined\_df,"allOR.txt",sep = "\t",quote = F,row.names = F)

ivw\_data <- combined\_df %>%

filter(method=="Inverse variance weighted")

write.table(combined\_df,"ivw\_allOR.txt",sep = "\t",quote = F,row.names = F)

6、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,dplyr)

# 创建一个空列表来存储所有数据框

df\_list <- list()

# 遍历所有TXT文件

files <- list.files(pattern="\*\_ORReverse.txt")

files

for(f in files){

# 读取每个TXT文件为数据框

df <- fread(f, header=TRUE,sep = "\t",quote = F,)

df$id <- f

# 使用 gsub 去掉 "\_OR.txt"

df$id <- gsub("\_ORReverse.txt", "", df$id)

# 添加到列表中

df\_list <- c(df\_list, list(df))

}

# 通过列名合并多个数据框

combined\_df <- Reduce(function(x,y) rbind(x,y), df\_list)

write.table(combined\_df,"allORReverse.txt",sep = "\t",quote = F,row.names = F)

ivw\_data <- combined\_df %>%

filter(method=="Inverse variance weighted")

write.table(ivw\_data,"ivw\_allORReverse.txt",sep = "\t",quote = F,row.names = F)

7、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,dplyr)

filename <- "allOR.txt"

Immcellsfile <- "Immcells.txt"

biofsci <- fread(filename, header=TRUE,sep = "\t",quote = F,)

Immdata <- fread(Immcellsfile, header=TRUE,sep = "\t",quote = F,)

biof\_imm <- merge(biofsci,Immdata,by="id")

write.table(biof\_imm,"all\_immOR.txt",sep = "\t",quote = F,row.names = F)

ivw\_data <- biof\_imm %>%

filter(method=="Inverse variance weighted")

write.table(ivw\_data,"ivw\_immOR.txt",sep = "\t",quote = F,row.names = F)

8、

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/") #https://mirrors.pku.edu.cn/CRAN/

# 加载“pacman”包，用于方便加载其他的R包

# 安装并加载pacman包,用于批量安装加载包

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置镜像加速bioconductor包安装

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载pacman包

library("pacman")

# 批量安装加载需要的包

p\_load(data.table,dplyr)

filename <- "allORReverse.txt"

Immcellsfile <- "Immcells.txt"

biofsci <- fread(filename, header=TRUE,sep = "\t",quote = F,)

Immdata <- fread(Immcellsfile, header=TRUE,sep = "\t",quote = F,)

biof\_imm <- merge(biofsci,Immdata,by="id")

write.table(biof\_imm,"all\_immORReverse.txt",sep = "\t",quote = F,row.names = F)

ivw\_data <- biof\_imm %>%

filter(method=="Inverse variance weighted")

write.table(ivw\_data,"ivw\_immORReverse.txt",sep = "\t",quote = F,row.names = F)

9、

#分析前准备

# 使用“ls()”函数获取当前环境中的所有变量、函数和对象的名称，然后传递给“list”参数

# 将“list”参数传递给“rm()”函数，表示删除这些变量、函数和对象

# 因此，“rm(list=ls())”表示删除当前环境中的所有变量、函数和对象，相当于清空当前环境

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载“pacman”包，用于方便加载其他的R包

library("pacman")

# 使用“p\_load”函数加载所需的R包

p\_load(forestploter,grid,ggplot2,data.table)

mrresultsfile <- "ivw\_immOR.txt"

# 定义变量mrresultsfile,用于存储MR结果文件的文件名

# 使用read.table()函数读取MR结果文件

# 文件名为mrresultsfile变量的值"MRresults.txt"

# header = T表示文件第一行作为变量名

# sep = "\t" 表示列之间使用制表符分隔

#biofsci = read.table(mrresultsfile, header = T, sep = "\t")

biofsci = fread(mrresultsfile, header = T)

colnames(biofsci)

# 处理P值的显示格式

# 对biofsci数据集的pval字段进行处理

# 使用ifelse()进行条件判断

# 如果pval小于0.001显示"<0.001"

# 否则使用sprintf()格式化显示P值到小数点后4位

biofsci$pval = as.numeric(biofsci$pval)

biofsci$pval <- ifelse(biofsci$pval<0.001, "<0.001", sprintf("%.4f", biofsci$pval))

# 格式化估计值列

biofsci$estimate <- paste0(format(round(biofsci$or, 2), nsmall = 2), " (",

format(round(biofsci$or\_lci95, 2), nsmall = 2), "-",

format(round(biofsci$or\_uci95, 2), nsmall = 2), ")")

biofsci$Trails <- biofsci$Trait

# 处理NA值,使用ifelse()对多个字段进行处理

# 如果值为NA,则替换为""

biofsci$Trails = ifelse(is.na(biofsci$Trails), "", biofsci$Trails)

biofsci$method = ifelse(is.na(biofsci$method), "", biofsci$method)

biofsci$nsnp = ifelse(is.na(biofsci$nsnp), "", biofsci$nsnp)

biofsci$pval = ifelse(is.na(biofsci$pval), "", biofsci$pval)

# 添加空格用于格式调整

# 使用rep()函数重复生成20个空格字符

# 使用paste()用空格拼接成一个字符串

# 赋值给新生成的变量 biofsci` `

colnames(biofsci)

biofsci$` ` <- paste(rep(" ", 15), collapse = " ")

biofsci$`OR(95%CI)` <- biofsci$estimate

# 提取class. 和 .id 之间的内容

# 仅提取class.和.id之间的内容

colnames(biofsci)

biofsci$Trails[duplicated(biofsci$Trails)] <- " "

biofsci1=biofsci[,c("Trails","method", "nsnp","pval","OR(95%CI)"," ")]

# 处理OR值的显示格式

# 使用ifelse()对OR字段进行处理

# 如果OR是NA,显示"",否则显示格式化的OR值和95%可信区间

colnames(biofsci1)

# 设置森林图的主题,定义图形元素的显示格式

# 使用forest\_theme()函数

# 定义基础字体大小,参考线颜色,脚注格式等参数

tm <- forest\_theme(base\_size = 10,

refline\_col = "red",

footnote\_col = "#636363",

footnote\_fontface = "italic")

# 绘制森林图

# 使用forest()函数,传递数据及参数

# biofsci[,c(1:5,9:10)]:选择需要的列数据

# est/lower/upper: OR值和置信区间

# arrow\_lab: 曲线两个方向的标签文本

# sizes: 点的大小

# ci\_column: 置信区间所在列

# ref\_line: 添加参考线

# xlim: x轴范围

# footnote: 脚注

# theme: 使用设置的主题对象

colnames(biofsci1)

pdf("forest1.pdf", width=8, height=8)

p=forest(biofsci1,

est = biofsci$or,

lower = biofsci$or\_lci95,

upper = biofsci$or\_uci95,

#arrow\_lab = c("Placebo Better", "Treatment Better"),

sizes = 0.4,

ci\_column = 6,

ref\_line = 1,

xlim = c(0.05, 3),

footnote = "",

theme = tm)

p

dev.off()

10、

#分析前准备

# 使用“ls()”函数获取当前环境中的所有变量、函数和对象的名称，然后传递给“list”参数

# 将“list”参数传递给“rm()”函数，表示删除这些变量、函数和对象

# 因此，“rm(list=ls())”表示删除当前环境中的所有变量、函数和对象，相当于清空当前环境

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载“pacman”包，用于方便加载其他的R包

library("pacman")

# 使用“p\_load”函数加载所需的R包

p\_load(forestploter,grid,ggplot2,data.table)

# 定义变量mrresultsfile,用于存储MR结果文件的文件名

mrresultsfile = "all\_immORReverse.txt"

# 使用read.table()函数读取MR结果文件

# 文件名为mrresultsfile变量的值"MRresults.txt"

# header = T表示文件第一行作为变量名

# sep = "\t" 表示列之间使用制表符分隔

#biofsci = read.table(mrresultsfile, header = T, sep = "\t")

biofsci = fread(mrresultsfile, header = T)

colnames(biofsci)

# 处理P值的显示格式

# 对biofsci数据集的pval字段进行处理

# 使用ifelse()进行条件判断

# 如果pval小于0.001显示"<0.001"

# 否则使用sprintf()格式化显示P值到小数点后4位

biofsci$pval = as.numeric(biofsci$pval)

biofsci$pval <- ifelse(biofsci$pval<0.001, "<0.001", sprintf("%.4f", biofsci$pval))

# 格式化估计值列

biofsci$estimate <- paste0(format(round(biofsci$or, 2), nsmall = 2), " (",

format(round(biofsci$or\_lci95, 2), nsmall = 2), "-",

format(round(biofsci$or\_uci95, 2), nsmall = 2), ")")

biofsci$outcome <- biofsci$Trait

# 处理NA值,使用ifelse()对多个字段进行处理

# 如果值为NA,则替换为""

biofsci$outcome = ifelse(is.na(biofsci$outcome), "", biofsci$outcome)

biofsci$method = ifelse(is.na(biofsci$method), "", biofsci$method)

biofsci$nsnp = ifelse(is.na(biofsci$nsnp), "", biofsci$nsnp)

biofsci$pval = ifelse(is.na(biofsci$pval), "", biofsci$pval)

# 添加空格用于格式调整

# 使用rep()函数重复生成20个空格字符

# 使用paste()用空格拼接成一个字符串

# 赋值给新生成的变量 biofsci` `

colnames(biofsci)

biofsci$` ` <- paste(rep(" ", 15), collapse = " ")

biofsci$`OR(95%CI)` <- biofsci$estimate

# 提取class. 和 .id 之间的内容

# 仅提取class.和.id之间的内容

colnames(biofsci)

biofsci$outcome[duplicated(biofsci$outcome)] <- " "

biofsci1=biofsci[,c("outcome","method", "nsnp","pval","OR(95%CI)"," ")]

# 处理OR值的显示格式

# 使用ifelse()对OR字段进行处理

# 如果OR是NA,显示"",否则显示格式化的OR值和95%可信区间

colnames(biofsci1)

# 设置森林图的主题,定义图形元素的显示格式

# 使用forest\_theme()函数

# 定义基础字体大小,参考线颜色,脚注格式等参数

tm <- forest\_theme(base\_size = 10,

refline\_col = "red",

footnote\_col = "#636363",

footnote\_fontface = "italic")

# 绘制森林图

# 使用forest()函数,传递数据及参数

# biofsci[,c(1:5,9:10)]:选择需要的列数据

# est/lower/upper: OR值和置信区间

# arrow\_lab: 曲线两个方向的标签文本

# sizes: 点的大小

# ci\_column: 置信区间所在列

# ref\_line: 添加参考线

# xlim: x轴范围

# footnote: 脚注

# theme: 使用设置的主题对象

colnames(biofsci1)

pdf("forest.pdf", width=15, height=50)

p=forest(biofsci1,

est = biofsci$or,

lower = biofsci$or\_lci95,

upper = biofsci$or\_uci95,

#arrow\_lab = c("Placebo Better", "Treatment Better"),

sizes = 0.4,

ci\_column = 6,

ref\_line = 1,

xlim = c(0.05, 3),

footnote = "",

theme = tm)

p

dev.off()

11、

#分析前准备

# 使用“ls()”函数获取当前环境中的所有变量、函数和对象的名称，然后传递给“list”参数

# 将“list”参数传递给“rm()”函数，表示删除这些变量、函数和对象

# 因此，“rm(list=ls())”表示删除当前环境中的所有变量、函数和对象，相当于清空当前环境

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载“pacman”包，用于方便加载其他的R包

library("pacman")

# 使用“p\_load”函数加载所需的R包

p\_load(forestploter,grid,ggplot2,data.table)

mrresultsfile <- "all\_immOR.txt"

# 定义变量mrresultsfile,用于存储MR结果文件的文件名

# 使用read.table()函数读取MR结果文件

# 文件名为mrresultsfile变量的值"MRresults.txt"

# header = T表示文件第一行作为变量名

# sep = "\t" 表示列之间使用制表符分隔

#biofsci = read.table(mrresultsfile, header = T, sep = "\t")

biofsci = fread(mrresultsfile, header = T)

biofsci$FDR <- p.adjust(biofsci$pval, method = "fdr")

write.table(biofsci,"all\_immOR\_FDR.txt",sep = "\t",quote = F,row.names = F)

12、

#分析前准备

# 使用“ls()”函数获取当前环境中的所有变量、函数和对象的名称，然后传递给“list”参数

# 将“list”参数传递给“rm()”函数，表示删除这些变量、函数和对象

# 因此，“rm(list=ls())”表示删除当前环境中的所有变量、函数和对象，相当于清空当前环境

rm(list=ls())

# 判断是否已经安装了“pacman”包，如果没有就安装它

if(!require("pacman")) install.packages("pacman",update = F,ask = F)

# 设置Bioconductor镜像地址为中国科技大学的镜像

options(BioC\_mirror="https://mirrors.ustc.edu.cn/bioc/")

# 加载“pacman”包，用于方便加载其他的R包

library("pacman")

# 使用“p\_load”函数加载所需的R包

p\_load(forestploter,grid,ggplot2,data.table)

mrresultsfile <- "all\_immOR\_FDR.txt"

# 定义变量mrresultsfile,用于存储MR结果文件的文件名

# 使用read.table()函数读取MR结果文件

# 文件名为mrresultsfile变量的值"MRresults.txt"

# header = T表示文件第一行作为变量名

# sep = "\t" 表示列之间使用制表符分隔

#biofsci = read.table(mrresultsfile, header = T, sep = "\t")

biofsci = fread(mrresultsfile, header = T)

colnames(biofsci)

# 处理P值的显示格式

# 对biofsci数据集的pval字段进行处理

# 使用ifelse()进行条件判断

# 如果pval小于0.001显示"<0.001"

# 否则使用sprintf()格式化显示P值到小数点后4位

biofsci$pval = as.numeric(biofsci$pval)

biofsci$pval <- ifelse(biofsci$pval<0.001, "<0.001", sprintf("%.4f", biofsci$pval))

# 格式化估计值列

biofsci$estimate <- paste0(format(round(biofsci$or, 2), nsmall = 2), " (",

format(round(biofsci$or\_lci95, 2), nsmall = 2), "-",

format(round(biofsci$or\_uci95, 2), nsmall = 2), ")")

biofsci$Trails <- biofsci$Trait

# 处理NA值,使用ifelse()对多个字段进行处理

# 如果值为NA,则替换为""

biofsci$Trails = ifelse(is.na(biofsci$Trails), "", biofsci$Trails)

biofsci$method = ifelse(is.na(biofsci$method), "", biofsci$method)

biofsci$nsnp = ifelse(is.na(biofsci$nsnp), "", biofsci$nsnp)

biofsci$pval = ifelse(is.na(biofsci$pval), "", biofsci$pval)

biofsci$FDR = ifelse(is.na(biofsci$FDR), "", biofsci$FDR)

# 添加空格用于格式调整

# 使用rep()函数重复生成20个空格字符

# 使用paste()用空格拼接成一个字符串

# 赋值给新生成的变量 biofsci` `

colnames(biofsci)

biofsci$` ` <- paste(rep(" ", 15), collapse = " ")

biofsci$`OR(95%CI)` <- biofsci$estimate

# 提取class. 和 .id 之间的内容

# 仅提取class.和.id之间的内容

colnames(biofsci)

biofsci$Trails[duplicated(biofsci$Trails)] <- " "

biofsci1=biofsci[,c("Trails","method", "nsnp","pval","OR(95%CI)"," ","FDR"," ")]

# 处理OR值的显示格式

# 使用ifelse()对OR字段进行处理

# 如果OR是NA,显示"",否则显示格式化的OR值和95%可信区间

colnames(biofsci1)

# 设置森林图的主题,定义图形元素的显示格式

# 使用forest\_theme()函数

# 定义基础字体大小,参考线颜色,脚注格式等参数

tm <- forest\_theme(base\_size = 10,

refline\_col = "red",

footnote\_col = "#636363",

footnote\_fontface = "italic")

# 绘制森林图

# 使用forest()函数,传递数据及参数

# biofsci[,c(1:5,9:10)]:选择需要的列数据

# est/lower/upper: OR值和置信区间

# arrow\_lab: 曲线两个方向的标签文本

# sizes: 点的大小

# ci\_column: 置信区间所在列

# ref\_line: 添加参考线

# xlim: x轴范围

# footnote: 脚注

# theme: 使用设置的主题对象

colnames(biofsci1)

pdf("forest.pdf", width=15, height=50)

p=forest(biofsci1,

est = biofsci$or,

lower = biofsci$or\_lci95,

upper = biofsci$or\_uci95,

#arrow\_lab = c("Placebo Better", "Treatment Better"),

sizes = 0.4,

ci\_column = 6,

ref\_line = 1,

xlim = c(0.05, 3),

footnote = "",

theme = tm)

p

dev.off()