# Set E -------------------------------------------------------------------

install.packages("showtext")

install.packages("networktools")

library(showtext)

font\_add("SimHei", "SimHei.ttf")

library(networktools)

showtext\_auto()

install.packages(c("psych", "qgraph", "bootnet"))

install.packages("rgl", type = "source")

library(rgl)

library(psych)

library(qgraph)

library(bootnet)

library(ggplot2)

library(dplyr)

setwd("C:/Users/lenovo/Desktop/PBICR/数据分析") # 换自己的wd

#导入数据

PSSS <- read.csv("PSSS.csv")

BFI <- read.csv("BFI.csv")

GAD <- read.csv("GAD-7.csv")

#导出列名

a <- colnames(PSSS)

a <- a[2:13]

a[[1]] <- "X44. 在我遇到问题时.有些人.亲戚.邻居.同事.会出现在我身旁"

b <- colnames(BFI)

b <- b[2:11]

b[1] <- "X43. 我认为我话不多"

c <- colnames(GAD)

c <- c[2:8]

c[1] <- "X75. 感觉紧张.焦虑或急切"

#列名合集items

items <- c(a,b,c)

#合并数据

combined\_df <- merge(PSSS, BFI, by = "序号")

combined\_df <- merge(combined\_df, GAD, by = "序号")

# 处理列名

n\_X44 <- 12 # Adjust based on your actual number of X44 columns

n\_X43 <- 10 # Adjust based on your actual number of X43 columns

n\_X75 <- 7 # Adjust based on your actual number of X75 columns

# Rename columns for X44, X43, and X75

colnames(combined\_df) <- c("序号", paste0("X44.", 1:n\_X44),

 paste0("X43.", 1:n\_X43),

 paste0("X75.", 1:n\_X75))

# Change to factor --------------------------------------------------------

# Mapping for GAD-7

map\_to\_factor\_GAD <- function(x) {

 levels <- c("完全没有", "有几天", "超过一半", "几乎每天")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Mapping for BFI

map\_to\_factor\_BFI <- function(x) {

 levels <- c("非常不同意", "有点不同意", "既不同意也不反对", "有点同意", "非常同意")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Mapping for PSSS

map\_to\_factor\_PSSS <- function(x) {

 levels <- c("极不同意", "很不同意", "稍不同意", "中立", "稍同意", "很同意", "极同意")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Apply the mappings

combined\_df <- combined\_df %>%

 mutate(across(starts\_with("X75"), map\_to\_factor\_GAD)) %>%

 mutate(across(starts\_with("X43"), map\_to\_factor\_BFI)) %>%

 mutate(across(starts\_with("X44"), map\_to\_factor\_PSSS))

# View the structure of the combined data frame

combined\_df <- combined\_df %>%

 mutate(across(starts\_with("X44"), as.numeric),

 across(starts\_with("X43"), as.numeric),

 across(starts\_with("X75"), as.numeric))

str(combined\_df)

#反向记分

combined\_df <- combined\_df %>%

 mutate(

 X43.1 = 6 - X43.1,

 X43.3 = 6 - X43.3,

 X43.4 = 6 - X43.4,

 X43.5 = 6 - X43.5,

 X43.7 = 6 - X43.7

 )

combined\_df <- combined\_df %>% select(c(2:30))

# 标准化数据

combined\_df\_scaled <- combined\_df %>%

 mutate(across(starts\_with("X44"), scale),

 across(starts\_with("X43"), scale),

 across(starts\_with("X75"), scale))

# 查看标准化后的数据结构

str(combined\_df\_scaled)

#分类

traits = c("OS", "OS", "FS", "FS", "OS","OS", "OS", "FS", "OS", "OS","FS", "OS", "Extraversion", "Agreeableness", "Conscientiousness", "Neuroticism", "Openness", "Extraversion", "Agreeableness", "Conscientiousness", "Neuroticism", "Openness", "Anxiety", "Anxiety","Anxiety","Anxiety","Anxiety","Anxiety","Anxiety")

install.packages("corrplot")

library(corrplot)

M<-cor(combined\_df)

View(M)

correlation\_df <- as.data.frame(M)

judge\_correlation <- function(r) {

 if (abs(r) >= 0.6) {

 return("强相关")

 } else if (abs(r) >= 0.4 & abs(r) < 0.6) {

 return("中相关")

 } else {

 return("弱相关")

 }

}

M<-cor(combined\_df)

correlation\_matrix<-M

# 创建一个矩阵来存储相关性强度的判断结果

correlation\_strengths\_matrix <- matrix(NA, nrow(correlation\_matrix), ncol(correlation\_matrix))

diag(correlation\_strengths\_matrix) <- NA # 对角线元素设置为NA（因为自身与自身的相关性总是1）

# 遍历非对角线元素，判断相关性强度

for (i in 1:(nrow(correlation\_matrix) - 1)) {

 for (j in (i + 1):ncol(correlation\_matrix)) {

 r <- correlation\_matrix[i, j]

 correlation\_strengths\_matrix[i, j] <- judge\_correlation(r)

 correlation\_strengths\_matrix[j, i] <- correlation\_strengths\_matrix[i, j] # 因为相关矩阵是对称的

 }

}

# 将相关性强度信息转换为向量（可选，用于统计）

correlation\_strengths <- as.vector(correlation\_strengths\_matrix)

correlation\_strengths <- correlation\_strengths[!is.na(correlation\_strengths)]

# 统计每种强度出现的次数和比例

strength\_table <- table(correlation\_strengths)

strength\_proportion <- prop.table(strength\_table)

# 打印结果

print("相关性强度统计：")

print(strength\_table)

print("相关性强度比例：")

print(strength\_proportion)

corrplot.mixed(M,tl.pos=c("lt"),

 number.cex=0.2)

# Network analysis --------------------------------------------------------

network <- estimateNetwork(combined\_df, default = "EBICglasso", corMethod = "spearman")

custom\_colors <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00","#CC79A7","#C6B9D3")

View(network)

getWmat(network)

adj.ggm<-getWmat(network)

sum(adj.ggm[upper.tri(adj.ggm,diag=F)]!=0)

total\_weight <- sum(adj.ggm[adj.ggm != 0])

num\_edges <- sum(adj.ggm[upper.tri(adj.ggm, diag = FALSE)] != 0)

average\_weight <- total\_weight / num\_edges

num\_edges <- network::nedges(network)

print(paste("Number of edges (using attribute):", num\_edges))

View(num\_edges)

#圆圈图

plot(network,

 layout = "circle",

 groups = traits,

 negDashed = T,

 color = custom\_colors

)

#经典图

plot(network,

 layout = "spring",

 groups = traits,

 negDashed = T,

 color = custom\_colors,

 #legend.mode = "style2",

 legend.cex = 0.5,

 #nodeNames = items,

 font = 2

)

#中心指标

centralityPlot(network, include = "all",scale="z-scores")

# 中心性指标稳定性

central\_stability <- bootnet(network, nBoots = 100, type = "case", statistics = c("edge","strength", "closeness", "betweenness"))

corStability(central\_stability)

#Drop 图

plot(central\_stability, statistics = c("strength", "closeness", "betweenness"))

# 边权重准确性，不要type = case

central\_stability1 <- bootnet(network, nBoots = 10, statistics = c("edge","strength", "closeness", "betweenness"))

central\_stability1

plot(central\_stability1, "strength", order = "sample")

plot(central\_stability1, "closeness", order = "sample")

plot(central\_stability1, "betweenness", order = "sample")

plot(central\_stability1, "edge", order = "sample")

# 分组Network analysis ------------------------------------------------------

# 创建新列并根据分组规则求和

combined\_df$Extraversion <- rowSums(combined\_df[, c("X43.1", "X43.6")])

combined\_df$Agreeableness <- rowSums(combined\_df[, c("X43.2", "X43.7")])

combined\_df$Conscientiousness <- rowSums(combined\_df[, c("X43.3", "X43.8")])

combined\_df$Neuroticism <- rowSums(combined\_df[, c("X43.4", "X43.9")])

combined\_df$Openness <- rowSums(combined\_df[, c("X43.5", "X43.10")])

combined\_df$FS <- rowSums(combined\_df[, c("X44.3", "X44.4", "X44.8", "X44.11")])

combined\_df$FOS <- rowSums(combined\_df[, c("X44.1", "X44.2", "X44.5", "X44.6", "X44.7", "X44.9", "X44.10", "X44.12")])

combined\_df$Anxiety <- rowSums(combined\_df[, paste0("X75.", 1:7)])

# 查看新的数据框架

head(combined\_df)

combined\_df <- combined\_df %>% select(c(30:37))

install.packages("corrplot")

library(corrplot)

M<-cor(combined\_df)

View(M)

correlation\_df <- as.data.frame(M)

judge\_correlation <- function(r) {

 if (abs(r) >= 0.6) {

 return("强相关")

 } else if (abs(r) >= 0.4 & abs(r) < 0.6) {

 return("中相关")

 } else {

 return("弱相关")

 }

}

M<-cor(combined\_df)

correlation\_matrix<-M

# 创建一个矩阵来存储相关性强度的判断结果

correlation\_strengths\_matrix <- matrix(NA, nrow(correlation\_matrix), ncol(correlation\_matrix))

diag(correlation\_strengths\_matrix) <- NA # 对角线元素设置为NA（因为自身与自身的相关性总是1）

# 遍历非对角线元素，判断相关性强度

for (i in 1:(nrow(correlation\_matrix) - 1)) {

 for (j in (i + 1):ncol(correlation\_matrix)) {

 r <- correlation\_matrix[i, j]

 correlation\_strengths\_matrix[i, j] <- judge\_correlation(r)

 correlation\_strengths\_matrix[j, i] <- correlation\_strengths\_matrix[i, j] # 因为相关矩阵是对称的

 }

}

# 将相关性强度信息转换为向量（可选，用于统计）

correlation\_strengths <- as.vector(correlation\_strengths\_matrix)

correlation\_strengths <- correlation\_strengths[!is.na(correlation\_strengths)]

# 统计每种强度出现的次数和比例

strength\_table <- table(correlation\_strengths)

strength\_proportion <- prop.table(strength\_table)

# 打印结果

print("相关性强度统计：")

print(strength\_table)

print("相关性强度比例：")

print(strength\_proportion)

corrplot.mixed(M,tl.pos=c("lt"),

 number.cex=0.4)

network <- estimateNetwork(combined\_df, default = "EBICglasso", corMethod = "spearman")

custom\_colors <- c("#CC79A7", "#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#8E44AD")

traits = colnames(combined\_df)

View(network)

str(combined\_df)

getWmat(network)

adj.ggm<-getWmat(network)

sum(adj.ggm[upper.tri(adj.ggm,diag=F)]!=0)

total\_weight <- sum(adj.ggm[adj.ggm != 0])

num\_edges <- sum(adj.ggm[upper.tri(adj.ggm, diag = FALSE)] != 0)

average\_weight <- total\_weight / num\_edges

#圆圈图

plot(network,

 layout = "circle", # 圆形布局

 groups = traits, # 节点的分组方式

 label.cex = 1.0, # 节点标签的大小

 label.color = 'black', # 节点标签的颜色

 negDashed = TRUE, # 负边使用虚线表示

 legend = TRUE, # 显示图例

 color = custom\_colors, # 自定义颜色

 legend.cex = 0.5, # 图例的大小

 legend.mode = 'style3', # 图例的样式

 nodeNames = colnames(data),# 节点名称

 font = 2, # 标签字体样式

 palette = "ggplot2") # 使用ggplot2调色板

plot(network,

 layout = "circle",

 groups = traits,

 negDashed = T,

 color = custom\_colors,

 legend.cex = 0.5,

 #nodeNames = items,

 font = 2

)

#经典图

plot(network,

 layout = "spring",

 groups = traits,

 negDashed = T,

 color = custom\_colors,

 legend.cex = 0.5,

 #nodeNames = items,

 font = 2

)

# 画中心性图

centralityPlot(network, orderBy = "Betweenness", include = "all", scale = "z-scores")

centralityPlot(network, orderBy = "Betweenness", include = "all")

# bridge

bridge\_sym <- bridge(network$graph, communities = c("1","1","1","1","1","2","2","3"))

bridge\_sym

#画bridge strength

plot(bridge\_sym, order = "value", include = c('Bridge Strength'))

# 进行Bootstrap抽样以评估中心性指标的稳定性

# 中心性指标稳定性

central\_stability <- bootnet(network, nBoots = 100, type = "case", statistics = c("edge","strength", "closeness", "betweenness"))

corStability(central\_stability)

# 计算中心稳定性系数

cs\_results <- corStability(central\_stability)

# 打印中心稳定性系数结果

print(cs\_results)

# 绘制中心性指标的稳定性图

plot(central\_stability, statistics = c("strength", "closeness", "betweenness"))

# 边权重准确性，不要type = case

central\_stability1 <- bootnet(network, nBoots = 500, statistics = c("edge","strength", "closeness", "betweenness", "bridgeStrength"),communities = c("1","1","1","1","1","2","2","3"))

central\_stability1

plot(central\_stability1, "strength", order = "sample")

plot(central\_stability1, "closeness", order = "sample")

plot(central\_stability1, "betweenness", order = "sample")

plot(central\_stability1, "edge", order = "sample")

# 提取桥接强度结果

#创建社区

communities = c("1","1","1","1","1","2","2","3")

#计算桥接中心性

bridge\_sym <- bridge(network$graph, communities = c("1","1","1","1","1","2","2","3"))

plot(bridge\_sym, order = "value", include = c('Bridge Strength'))

#节点Bridge strength

bridge\_strength <- bridge\_sym$`Bridge Strength`

bridge\_strength #节点最大的是Anxiety，其实和上面一样

# 创建一个数据框来存储结果

bridge\_df <- data.frame(

 Variable = colnames(combined\_df),

 Community = communities,

 BridgeStrength = bridge\_strength

)

# 按社区分组并计算平均桥接强度

avg\_bridge\_strength <- aggregate(BridgeStrength ~ Community, data = bridge\_df, FUN = mean)

# 打印平均桥接强度

print(avg\_bridge\_strength)

# 可视化桥接强度的分布

ggplot(bridge\_df, aes(x = as.factor(Community), y = BridgeStrength, fill = as.factor(Community))) +

 geom\_boxplot() +

 labs(x = "Community", y = "Bridge Strength", title = "Bridge Strength Distribution by Community") +

 theme\_minimal()

#性别网络比较

install.packages("bruceR")

install.packages("NetworkComparisonTest")

install.packages(c("psych", "qgraph", "bootnet"))

install.packages("Networktools")

install.packages("showtext")

library(showtext)

library(qgraph)

library(ggplot2)

library("bruceR")

library("NetworkComparisonTest")

library("networktools")

setwd("C:/Users/lenovo/Desktop/August - 副本")

Male<-import("Male.xlsx")

Female<-import("Female.xlsx")

map\_to\_factor\_Male <- function(x) {

 levels <- c("完全没有", "有几天", "超过一半", "几乎每天")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Mapping for Male

map\_to\_factor\_Male <- function(x) {

 levels <- c("非常不同意", "有点不同意", "既不同意也不反对", "有点同意", "非常同意")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Mapping for Male

map\_to\_factor\_Male <- function(x) {

 levels <- c("极不同意", "很不同意", "稍不同意", "中立", "稍同意", "很同意", "极同意")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Apply the mappings

Male <- mutate(Male, across(starts\_with("X75"), map\_to\_factor\_GAD))

Male <- mutate(Male, across(starts\_with("X43"), map\_to\_factor\_BFI))

Male <- mutate(Male, across(starts\_with("X44"), map\_to\_factor\_PSSS))

Male <- Male %>%

 mutate(across(starts\_with("X75"), map\_to\_factor\_GAD)) %>%

 mutate(across(starts\_with("X43"), map\_to\_factor\_BFI)) %>%

 mutate(across(starts\_with("X44"), map\_to\_factor\_PSSS))

# View the structure of the combined data frame

Male <- Male %>%

 mutate(across(starts\_with("X44"), as.numeric),

 across(starts\_with("X43"), as.numeric),

 across(starts\_with("X75"), as.numeric))

str(Male)

Male <- Male %>%

 mutate(

 X43.1 = 6 - X43.1,

 X43.3 = 6 - X43.3,

 X43.4 = 6 - X43.4,

 X43.5 = 6 - X43.5,

 X43.7 = 6 - X43.7

 )

Male\_scaled <- Male %>%

 mutate(across(starts\_with("X44"), scale),

 across(starts\_with("X43"), scale),

 across(starts\_with("X75"), scale))

map\_to\_factor\_Female <- function(x) {

 levels <- c("完全没有", "有几天", "超过一半", "几乎每天")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Mapping for Female

map\_to\_factor\_Female <- function(x) {

 levels <- c("非常不同意", "有点不同意", "既不同意也不反对", "有点同意", "非常同意")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Mapping for Female

map\_to\_factor\_Female <- function(x) {

 levels <- c("极不同意", "很不同意", "稍不同意", "中立", "稍同意", "很同意", "极同意")

 factor(x, levels = levels, labels = 1:length(levels))

}

# Apply the mappings

Female <- Female %>%

 mutate(across(starts\_with("X75"), map\_to\_factor\_GAD)) %>%

 mutate(across(starts\_with("X43"), map\_to\_factor\_BFI)) %>%

 mutate(across(starts\_with("X44"), map\_to\_factor\_PSSS))

# View the structure of the combined data frame

Female <- Female %>%

 mutate(across(starts\_with("X44"), as.numeric),

 across(starts\_with("X43"), as.numeric),

 across(starts\_with("X75"), as.numeric))

str(Female)

Female <- Female %>%

 mutate(

 X43.1 = 6 - X43.1,

 X43.3 = 6 - X43.3,

 X43.4 = 6 - X43.4,

 X43.5 = 6 - X43.5,

 X43.7 = 6 - X43.7

 )

Female\_scaled <- Female %>%

 mutate(across(starts\_with("X44"), scale),

 across(starts\_with("X43"), scale),

 across(starts\_with("X75"), scale))

gender\_compare<-NCT(Male,Female,gamma=0.5,it=1000,

 binary.data=FALSE,

 test.edges = TRUE, edges="all",

 test.centrality = TRUE,centrality = c("strength"),

 p.adjust.methods = "BH",

 nodes="all")

save(gender\_compare,file="gender\_compare.Rdata")

summary(gender\_compare)

#global strength results全局强度结果

gender\_compare$glstrinv.pval

gender\_compare$glstrinv.real

plot(gender\_compare,what="strength",labels = sprintf("%.2f", df$pvalue))

#network test results网络结构比较结果

gender\_compare$nwinv.pval

gender\_compare$nwinv.real

plot(gender\_compare,what="network",labels = sprintf("%.2f", df$pvalue))

#edge results边线比较结果

gender\_compare$einv.pval

sum(gender\_compare$einv)