

```
#医学研究のための因果推論レクチャー
#付録Rコマンド
#更新日 2024年2月26日
```

```
#◆第7章Rコマンドは以下
```

```
install.packages("sandwich")
install.packages("lmtest")
install.packages("tidyverse")
library("sandwich")
library("lmtest")
library("tidyverse")
#データセット作成
id <- 1:3000
aspirin <- c(rep(1, 500), rep(0, 2500))
CHDhistory <- c(rep(1, 300), rep(0, 200), rep(1, 200), rep(0, 2300))
CHDevent <-
c(rep(1, 30), rep(0, 270), rep(1, 2), rep(0, 198), rep(1, 60), rep(0, 140), rep(1, 69), rep(0, 2231))
df <- data.frame(id, aspirin, CHDhistory, CHDevent)
df$aspirin_lab <- factor(df$aspirin, levels = c(0, 1), labels = c("No
Aspirin", "Aspirin"))
df$CHDhistory_lab <- factor(df$CHDhistory, levels = c(0, 1), labels = c("No
CHDhistory", "CHDhistory"))
df$CHDevent_lab <- factor(df$CHDevent, levels = c(0, 1), labels = c("No
CHDevent", "CHDevent"))
#第6回 表6-1
df%>%
  filter (CHDhistory_lab == "No CHDhistory") -> df0
df%>%
  filter (CHDhistory_lab == "CHDhistory") -> df1
addmargins(table(df$aspirin_lab, df$CHDevent_lab))
32/500-129/5000
(32/500)/(129/2500)
addmargins(table(df1$aspirin_lab, df1$CHDevent_lab))
30/300-60/200
(30/300)/(60/200)
addmargins(table(df0$aspirin_lab, df0$CHDevent_lab))
2/200-69/2300
(2/200)/(69/2300)
#第7章 表7-2
#ロジスティック回帰モデル
model1 <- glm(CHDevent ~ aspirin + CHDhistory, family = binomial(link = "logit"), data
= df)
summary(model1)
estimate<-(summary(model1)$coefficients[2, "Estimate"]) %>% exp() %>% round(3)
ci. low<-(summary(model1)$coefficients[2, "Estimate"] +
qnorm(0.05/2)*summary(model1)$coefficients[2, "Std. Error"]) %>% exp() %>% round(3)
ci. up<-(summary(model1)$coefficients[2, "Estimate"] +
qnorm(1-0.05/2)*summary(model1)$coefficients[2, "Std. Error"]) %>% exp() %>% round(3)
paste("アスピリンのオッズ比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=", estimate, "
(", ci. low, ", ", ci. up, ")", sep="")
estimate<-(summary(model1)$coefficients[3, "Estimate"]) %>% exp() %>% round(3)
ci. low<-(summary(model1)$coefficients[3, "Estimate"] +
qnorm(0.05/2)*summary(model1)$coefficients[3, "Std. Error"]) %>% exp() %>% round(3)
ci. up<-(summary(model1)$coefficients[3, "Estimate"] +
qnorm(1-0.05/2)*summary(model1)$coefficients[3, "Std. Error"]) %>% exp() %>% round(3)
paste("冠動脈疾患既往のオッズ比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=",
estimate, "(", ci. low, ", ", ci. up, ")", sep="")
#修正ポアソンモデル
model2 <- glm(CHDevent ~ aspirin + CHDhistory, family = poisson(link = "log"), data =
df)
summary(model2)
#ロバスト分散に基づいた信頼区間を求める
estimate<-coefest(model2, vcov = sandwich) [2, 1]%>% exp() %>% round(3)
```

```

ci.low<-(coefest(model2, vcov = sandwich)[2,1] + qnorm(0.05/2)*coefest(model2, vcov =
sandwich)[2,2]) %>% exp() %>% round(3)
ci.up<-(coefest(model2, vcov = sandwich)[2,1]+ qnorm(1-0.05/2)*coefest(model2, vcov =
sandwich)[2,2]) %>% exp() %>% round(3)
paste("アスピリンのリスク比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=", estimate, "
(", ci.low, ",", ci.up, ")", sep="")
estimate<-coefest(model2, vcov = sandwich)[3,1]%>% exp()%>% round(3)
ci.low<-(coefest(model2, vcov = sandwich)[3,1] + qnorm(0.05/2)*coefest(model2, vcov =
sandwich)[3,2]) %>% exp() %>% round(3)
ci.up<-(coefest(model2, vcov = sandwich)[3,1]+ qnorm(1-0.05/2)*coefest(model2, vcov =
sandwich)[3,2]) %>% exp() %>% round(3)
paste("冠動脈疾患既往のリスク比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=",
estimate, "(", ci.low, ",", ci.up, ")", sep="")

```

#◆第7章Rコマンドここまで

#-----
-

#◆第8章Rコマンドは以下

```

install.packages("sandwich")
install.packages("lmtest")
install.packages("tidyverse")
install.packages("MatchIt")
install.packages("Rcpp")
install.packages("geepack")
install.packages("cobalt")
library("sandwich")
library("lmtest")
library("tidyverse")
library("MatchIt")
library("Rcpp")
library("geepack")
library("cobalt")

#データを作成する
set.seed(777)
N <- 10000
id<- (1:10000)
age <- floor(rnorm(N, 60, 10))
sex <- rbinom(N, 1, 0.5)
dl<-rbinom(N, 1, 1/(1+exp(-(-7+0.1*age-0.5*sex))))
CHDhistory<-rbinom(N, 1, 1/(1+exp(-(-4.81+0.05*age-0.5*sex+1*dl))))
aspirin<-rbinom(N, 1, 1/(1+exp(-(-5.6+0.05*age-0.5*sex+0.5*dl+2.87*CHDhistory))))
CHDevent<-rbinom(N, 1,
1/(1+exp(-(-10-1.4675*aspirin+0.1*age-0.5*sex+1*dl+2*CHDhistory))))
data<-data.frame(id, age, sex, dl, CHDhistory, aspirin, CHDevent)
summary(data)

#傾向スコアを求める
model_ps <- glm(aspirin ~ CHDhistory+age+sex+dl, family = binomial(link = "logit"),
data = data)
data$ps <- predict(model_ps, data)

#1. 回帰モデル（修正ポアソンモデル）
model1<- glm(CHDevent ~ aspirin + CHDhistory+age+sex+dl, family = poisson(link =
"log"), data = data)
summary(model1)
estimate<-coefest(model1, vcov = sandwich)[2,1]%>% exp()%>% round(3)
ci.low<-(coefest(model1, vcov = sandwich)[2,1] + qnorm(0.05/2)*coefest(model1, vcov =
sandwich)[2,2]) %>% exp() %>% round(3)
ci.up<-(coefest(model1, vcov = sandwich)[2,1]+ qnorm(1-0.05/2)*coefest(model1, vcov =
sandwich)[2,2]) %>% exp() %>% round(3)

```

```
paste("アスピリンのリスク比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=", estimate, "  
(", ci.low, ", ", ci.up, ") ", sep="")
```

#2. PSを回帰モデルの共変量として用いる

```
model2 <- glm(CHDevent ~ aspirin + ps, family = poisson(link = "log"), data = data)  
summary(model2)  
estimate<-coefest(model2, vcov = sandwich)[2,1]%>% exp()%>% round(3)  
ci.low<-(coefest(model2, vcov = sandwich)[2,1] + qnorm(0.05/2)*coefest(model2, vcov =  
sandwich)[2,2]) %>% exp() %>% round(3)  
ci.up<-(coefest(model2, vcov = sandwich)[2,1]+ qnorm(1-0.05/2)*coefest(model2, vcov =  
sandwich)[2,2]) %>% exp() %>% round(3)  
paste("アスピリンのリスク比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=", estimate, "  
(", ci.low, ", ", ci.up, ") ", sep="")
```

#3. PSマッチング

```
fit.psm <- matchit(aspirin ~ age+sex+dl+CHDhistory, method = "nearest", caliper =0.005,  
                  distance = "linear.logit", data=data)  
data.psm <- match.data(fit.psm)
```

#3-0. バランスチェック

```
summary(fit.psm)  
bal.tab(fit.psm, thresholds = c(m = 0.1), un = TRUE)  
bal.plot(fit.psm, var.name = "age")  
bal.plot(fit.psm, var.name = "age", mirror = TRUE, type = "histogram")  
love.plot(fit.psm, stats = c("mean.diffs"),  
          thresholds = c(m = 0.1), abs = TRUE, binary = "std", var.order =  
"unadjusted")
```

#3-1. マッチングされた集団での因果リスク比 (マッチング考慮なし)

```
model3 <- glm(CHDevent ~ aspirin, family = poisson(link = "log"), data = data.psm)  
summary(model3)  
estimate<-coefest(model3, vcov = sandwich)[2,1]%>% exp()%>% round(3)  
ci.low<-(coefest(model3, vcov = sandwich)[2,1] + qnorm(0.05/2)*coefest(model3, vcov =  
sandwich)[2,2]) %>% exp() %>% round(3)  
ci.up<-(coefest(model3, vcov = sandwich)[2,1]+ qnorm(1-0.05/2)*coefest(model3, vcov =  
sandwich)[2,2]) %>% exp() %>% round(3)  
paste("アスピリンのリスク比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=", estimate, "  
(", ci.low, ", ", ci.up, ") ", sep="")
```

#3-2. マッチングされた集団での因果リスク比 (※推奨：マッチング考慮あり)

```
model4_1_results <-coefest(model3, vcov. = vcovCL, cluster = ~subclass)  
estimate<-model4_1_results[2,1]%>% exp()%>% round(3)  
ci.low<-(model4_1_results[2,1] + qnorm(0.05/2)*model4_1_results[2,2]) %>% exp() %>%  
round(3)  
ci.up<-(model4_1_results[2,1]+ qnorm(1-0.05/2)*model4_1_results[2,2]) %>% exp() %>%  
round(3)  
paste("アスピリンのリスク比：点推定値 (95%信頼区間下限, 95%信頼区間上限)=", estimate, "  
(", ci.low, ", ", ci.up, ") ", sep="")
```

#◆第8章Rコマンドここまで

```
#-----  
-
```

#◆第9章Rコマンドは以下

```
rm(list = ls(all.names = TRUE))  
install.packages("lmtest")  
install.packages("sandwich")  
install.packages("tidyverse")  
install.packages("boot")  
library("lmtest")  
library("sandwich")
```

```

library("tidyverse")
library(boot)

#データを作成する
set.seed(777)
N <- 100000
id<- (1:100000)
age <- floor(rnorm(N, 60, 10))
sex <- rbinom(N, 1, 0.5)
ldl_base<-rnorm(N, 120+0.1*age-1.5*sex, 15)
CHDhistory<-rbinom(N, 1, 1/(1+exp(-(-6+0.05*age-0.5*sex+0.01*ldl_base))))
statin_base<-rbinom(N, 1,
1/(1+exp(-(-5.6+0.05*age-0.5*sex+0.01*ldl_base+1*CHDhistory))))
ldl_follow<-rnorm(N, 120+0.1*age-30*sex+0.05*ldl_base+10*CHDhistory-20*statin_base, 15)
statin_follow<-rbinom(N, 1,
1/(1+exp(-(-5+0.05*age-0.5*sex+2*statin_base+0.01*(ldl_base-100)+0.03*(ldl_follow-100)+
0.2*CHDhistory))))
CHDevent<-rbinom(N, 1,
1/(1+exp(-(-12-0.2*statin_base-0.2*statin_follow-0.05*statin_base*statin_follow+0.1*age
-0.5*sex+0.01*ldl_base+0.03*ldl_follow+1*CHDhistory))))
data<-data.frame(id, age, sex, ldl_base, ldl_follow, CHDhistory, statin_base,
statin_follow, CHDevent)
summary(data)

#IPTW
iptw<-function(d0, indices) {
  d <- d0[indices,]
  #T1に対する重みを求める
  psmodel1<-glm(statin_base ~ age+sex+ldl_base+CHDhistory,
                data=d, family=binomial(link="logit"))
  d$ps1<-predict(psmodel1, type = "response")
  psmodel2<-glm(statin_base ~ 1, data=d, family=binomial(link="logit"))
  d$ps2<-predict(psmodel2, type = "response")
  d$weight1 <- ifelse(d$statin_base==1, d$ps2/d$ps1, (1-d$ps2)/(1-d$ps1))
  #d$weight1[d$weight1>quantile(d$weight1, c(.99))]<- quantile(d$weight1, c(.99)) #必
  要に応じてtruncateする
  #d$weight1[d$weight1<quantile(d$weight1, c(.01))]<- quantile(d$weight1, c(.01)) #必
  要に応じてtruncateする

  #T2に対する重みを求める
  psmodel3<-glm(statin_follow ~ age+sex+ldl_base+CHDhistory+ldl_follow+statin_base,
                data=d, family=binomial(link="logit"))
  d$ps3<-predict(psmodel3, type = "response")
  psmodel4<-glm(statin_follow ~ 1, data=d, family=binomial(link="logit"))
  d$ps4<-predict(psmodel4, type = "response")
  d$weight2 <- ifelse(d$statin_follow==1, d$ps4/d$ps3, (1-d$ps4)/(1-d$ps3))
  #d$weight2[d$weight2>quantile(d$weight2, c(.99))]<- quantile(d$weight2, c(.99)) #必
  要に応じてtruncateする
  #d$weight2[d$weight2<quantile(d$weight2, c(.01))]<- quantile(d$weight2, c(.01)) #必
  要に応じてtruncateする

  #T1, T2に対する重みを求める
  weight<-d$weight1*d$weight2
  #hist(weight)
  #summary(weight)
  # (重みをApplyしたときに曝露群・非曝露群で共変量のバランスが取れているかを確認する)
  model<-glm(CHDevent ~ statin_base*statin_follow, data=d, family = poisson(link =
"log"),
            weights=weight)

  #Return value
  estimate_bt2 = exp(sum(coef(model)[c(' statin_follow')]))

```

```

    estimate_bt1 = exp(sum(coef(model)[c('statin_base')]))
    estimate_bt12 = exp(sum(coef(model)[c('statin_base', 'statin_follow',
'statin_base:statin_follow')]))
    return(c(bt2=estimate_bt2, bt1=estimate_bt1, bt12=estimate_bt12))
}

```

#ブートストラップ法で95% CIを計算する。

```

results_iptw <- boot(data=data, statistic=iptw, R=200)
mean_iptw <- results_iptw$t0
se_iptw <- c(sd(results_iptw$t[,1]), sd(results_iptw$t[,2]), sd(results_iptw$t[,3]))
ll_iptw <- mean_iptw - qnorm(0.975)*se_iptw
ul_iptw <- mean_iptw + qnorm(0.975)*se_iptw
bootstrap_iptw <- data.frame(cbind(c("Statin-/+","Statin+/-",
"Statin+/+"), mean_iptw, ll_iptw, ul_iptw))
bootstrap_iptw

```

#G-computations

```

gcomp =function(d0, indices) {
  d <- d0[indices,]
  ##Step 1. M (LDL follow) と Y (CHDevent) の予測モデルを構築する
  d$statin_temp1<-d$statin_base
  d$statin_temp2<-d$statin_follow
  reg_M<-glm(ldl_follow ~ age+sex+ldl_base+CHDhistory+statin_temp1, data=d,
family=gaussian)
  d$ldl_temp<-d$ldl_follow
  reg_Y<-glm(CHDevent ~ age+sex+ldl_base+CHDhistory+ldl_temp+statin_temp1*statin_temp2,
data=d, family=binomial(link="logit"))

```

#Step2. データをコピーして、スタチン内服の状態を新しく割り当てる

```

data1<-data2<-data3<-data4<-d
#全員スタチン内服（研究開始時）している/全員スタチン内服（1年後フォロー）している場
合
data1$statin_temp1=1
data1$statin_temp2=1
#全員スタチン内服（研究開始時）している/全員スタチン内服（1年後フォロー）していない
場合
data2$statin_temp1=1
data2$statin_temp2=0
#全員スタチン内服（研究開始時）していない/全員スタチン内服（1年後フォロー）している
場合
data3$statin_temp1=0
data3$statin_temp2=1
#全員スタチン内服（研究開始時）していない/全員スタチン内服（1年後フォロー）していな
い場合
data4$statin_temp1=0
data4$statin_temp2=0
#4つのコピーデータを結合する。
newdata<-rbind(data1, data2, data3, data4)

```

#Step3: 結合したデータで、新しく割り当てられたスタチン内服の状態のもと、MとYを改めて予測する

```

N<-nrow(d)
newdata$ldl_temp<-predict(reg_M, newdata=newdata)
newdata$pY<-rbinom(4*N, size = 1, prob=1/(1+exp(-(predict(reg_Y, newdata=newdata)))))

```

#Step4: ステップ3で算出されたアウトカムの予測値に対して新しく割り当てられたスタチン内服の状態を回帰モデルに当てはめて因果効果を推定する

```

reg_po<-glm(pY ~ statin_temp1*statin_temp2, data=newdata, family = poisson(link =
"log"))

```

#Step5: Return value

```

reg_po_results <-coefctest(reg_po, vcov. = vcovCL, cluster = ~id)
estimate_bt2 = exp(reg_po_results['statin_temp2', 1])

```

```

estimate_bt1 = exp(reg_po_results['statin_temp1', 1])
estimate_bt12 = exp(reg_po_results['statin_temp1', 1]+reg_po_results['statin_temp2',
1]+ reg_po_results['statin_temp1:statin_temp2', 1])
return(c(bt2=estimate_bt2, bt1=estimate_bt1, bt12=estimate_bt12))
}

```

#ブートストラップ法で95% CIを計算する。

```

results_gcomp <- boot(data=data, statistic=gcomp, R=200)
mean_gcomp <- results_gcomp$t0
se_gcomp <- c(sd(results_gcomp$t[, 1]), sd(results_gcomp$t[, 2]),
sd(results_gcomp$t[, 3]))
ll_gcomp <- mean_gcomp - qnorm(0.975)*se_gcomp
ul_gcomp <- mean_gcomp + qnorm(0.975)*se_gcomp
bootstrap_gcomp <- data.frame(cbind(c("Statin-/+","Statin+/-",
"Statin+/+"), mean_gcomp, ll_gcomp, ul_gcomp))

bootstrap_gcomp

```

#◆第9章Rコマンドここまで

#-----
-

#◆第11章Rコマンドは以下

```

rm(list = ls(all.names = TRUE))
install.packages("geepack")
install.packages("broom")
install.packages("devtools")
devtools::install_github("BS1125/CMAverse")
#install.packages("remotes") #if the above code doesn't work to install CMAverse
#remotes::install_github("LindaValeri/CMAverse", force = TRUE)

library("geepack")
library("broom")
library("CMAverse")

#データを作成する
set.seed(777)
N <- 100000
ID<- (1:N)
C1 <-rbinom(N, 1, 0.3) #E-M confounder
C2 <-rbinom(N, 1, 0.3) #M-Y confounder
C3 <-rbinom(N, 1, 0.3) #E-Y confounder
E <-rbinom(N, 1, 1/(1+exp(-(log(0.30/0.70) + log(2.0)*C1+ log(2.0)*C3))))
M <-rbinom(N, 1, 1/(1+exp(-(log(0.20/0.80) + log(3.0)*E + log(2.0)*C1 + log(2.0)*C2))))
Y <-rbinom(N, 1, 1/(1+exp(-(log(0.03/0.97) + log(1.5)*E + log(1.5)*M
+ log(1.2)*M*E + log(3.0)*C2+ log(3.0)*C3))))
d<-as.data.frame(cbind(C1, C2, C3, E, M, Y, ID))
summary(d)

## 回帰モデル
# Total effect
te_out<-glm(data=d, Y ~ E + C1+C3, family = binomial)
summary(te_out)
exp(cbind(OR=coef(te_out), confint(te_out)))

# Controlled direct effect
cde_out<-glm(data=d, Y ~ E + C1 + C2 +C3 + M + E*M, family = binomial)
summary(cde_out)
exp(cbind(OR=coef(cde_out), confint(cde_out)))

#Total effect of E: OR = 1.83872059
#Controlled direct effect of E: OR =1.47904848

```

```

## CMAverseパッケージ
#
https://bs1125.github.io/CMAverse/articles/overview.html#the-marginal-structural-model-1
#
https://bs1125.github.io/CMAverse/articles/single_mediator.html#the-marginal-structural-model-1
set.seed(123)
fit_msms <- cmest(data = d, model = "msm", outcome = "Y", exposure = "E", #Marginal
Structure approach
  mediator = "M", basec = c("C1", "C2", "C3"), EMint = TRUE,
  ereg = glm(E ~ C1+C3, family = binomial(link="logit"), data = d),
  yreg = glm(Y ~ E*M, family = binomial(link="logit"), data = d),
  mreg = list(glm(M ~ E, family = binomial(link="logit"), data =
d)),
  wmnomreg = list(glm(M ~ E, family = binomial(link="logit"), data =
data = d)),
  wmdenomreg = list(glm(M ~ E+C1+C2, family = binomial(link="logit"),
  data = d)),
  astar = 0, a = 1, mval = list(0),
  estimation = "imputation", inference = "bootstrap", nboot = 2)
summary(fit_msms)
#Total effect of E: RR = 1.8233943
#Controlled direct effect of E: RR = 1.4421112
#See the summary output for other causal effects of interest

## IPWアプローチ
#IPW for exposure
fitipwe<-glm(data=d,E ~ C1+C3,family = binomial)
d$p_e <- predict(fitipwe, d, type="response")
fitipwen<-glm(data=d,E ~ 1,family = binomial)
d$p_en <- predict(fitipwen, d, type="response")
d$sw_e<-ifelse(E==1,yes= d$p_en / d$p_e,
no= ifelse(E==0, yes= (1-d$p_en) / (1-d$p_e), no=NA))

#IPW for mediator
fitipwm<-glm(data=d,M ~ E+C1+C2+C3,family = binomial)
d$p_m <- predict(fitipwm, d, type="response")
fitipwmn<-glm(data=d,M ~ 1,family = binomial)
d$p_mn <- predict(fitipwmn, d, type="response")
d$sw_m<-ifelse(M==1,yes=d$p_mn / d$p_m,
no= ifelse(M==0, yes= (1-d$p_mn) / (1-d$p_m), no=NA))

# Combine IPWs
d$sw_e_m<- d$sw_e*d$sw_m

# Total effect of E
msm.te <- geeglm(
  Y ~ E,
  data = d,
  weights = sw_e,
  id = ID,
  family=binomial(link=logit),
  corstr = "independence"
)
summary(msm.te)
msm.te.out<-broom::tidy(x = msm.te, conf.int = TRUE)
exp(cbind(RR=msm.te.out[2,2],msm.te.out[2,6:7]))

# Controlled direct effect
msm.cde <- geeglm(
  Y ~ E*M,
  data = d,
  weights = sw_e_m,

```

```

    id = ID,
    family=binomial(link=logit),
    corstr = "independence"
)
summary(msm.cde)
msm.cde.out<-broom::tidy(x = msm.cde, conf.int = TRUE)
exp(cbind(RR=msm.cde.out[2, 2],msm.cde.out[2, 6:7]))
#Total effect of E: RR = 1.81
#Controlled direct effect of E: RR = 1.44

```

#◆第11章Rコマンドここまで

```

#-----
-

```

#◆第12章Rコマンドは以下

```

##使用するパッケージのインストールとロード
install.packages("tableone")
install.packages("tidyverse")
library("tableone")
library("tidyverse")

```

##データの作成

```

df1 <- data.frame(E = 1, D=1, Z=1, ID = 1:60)
df2 <- data.frame(E = 1, D=0, Z=1, ID = 61:100)
df3 <- data.frame(E = 0, D=1, Z=1, ID = 101:700)
df4 <- data.frame(E = 0, D=0, Z=1, ID = 701:1100)
df5 <- data.frame(E = 1, D=1, Z=0, ID = 1101:1140)
df6 <- data.frame(E = 1, D=0, Z=0, ID = 1141:2000)
df7 <- data.frame(E = 0, D=1, Z=0, ID = 2001:6400)
df8 <- data.frame(E = 0, D=0, Z=0, ID = 6401:101000)
df <- rbind(df1, df2, df3, df4, df5, df6, df7, df8)

```

```

#作成したデータの確認
summary(df)

```

#本文中の表と値が一致することを確認

```

CreateTableOne(vars = c("Z", "D"), strata = c("E"), data = df, test=FALSE)
count <-addmargins(table(df$D, df$E))
print(count)

```

#バイアス調整前のRR

```

RR<-(count[2, 2]/count[3, 2])/(count[2, 1]/count[3, 1]) %>% round(3)
print(paste("バイアス調整前RR=", RR))
#exp(glm(D~E, family=binomial(link = "log"), data=df)$coef) #glmを用いた分析

```

#穴埋めしたシナリオにおけるバイアス調整後のリスク比(RR)

```

AdjustedRR <-exp(glm(D~E+Z, family=binomial(link = "log"), data=df)$coef[2]) %>%
round(3)# 共変量調整した結果
print(paste("バイアス調整後RR(共変量調整)=", AdjustedRR))
AdjustedRR1 <- exp(glm(D~E, family=binomial(link =
"log"), data=df, subset=df$Z==1)$coef[2]) %>% round(3) #Z=1に限定
print(paste("バイアス調整後RR(Z=1に限定)=", AdjustedRR1))
AdjustedRR0 <- exp(glm(D~E, family=binomial(link =
"log"), data=df, subset=df$Z==0)$coef[2]) %>% round(3) #Z=0に限定
print(paste("バイアス調整後RR(Z=0に限定)=", AdjustedRR0))

```

#Bias formulaを用いてシナリオを与えたときのバイアス分析

```

rrdz <- 13.5
pz0 <- 0.01
pz1 <-0.1
bias <- ((rrdz*pz1 + 1 - pz1) / (rrdz*pz0 + 1 - pz0)) # Bias formula

```



```
Adjusted2RR <- RR/bias %>% round(3)
print(paste("バイアス調整後RR=", Adjusted2RR))
```

```
#様々なシナリオを与えた場合のバイアス分析
```

```
rrdz <- 1:20
```

```
pz0 <- 0.01
```

```
pz1 <- 1:30*0.01
```

```
bias <- ((rrdz*pz1 + 1 - pz1) / (rrdz*pz0 + 1 - pz0)) # Bias formula
```

```
Adjusted3RR <- (RR/bias) %>% round(3)
```

```
print(paste("RRdz=", rrdz, "pz0=", pz0, "pz1=", pz1, ":", Adjusted3RR))
```

```
#◆第12章Rコマンドここまで
```